PMTA: Perception-Aware Multi-Task Transformer Network for Personalized Multi-Domain Adaptation

Chenbin Zhang
Bytedance
Beijing, China
aleczhang13@gmail.com

Xiaoxie Zhu*
Bytedance
Shanghai, China
zhuxiaoxie.777@bytedance.com

Xingchao Cao
Bytedance
Beijing, China
caoxingchao@gmail.com

Qiwen Chen
Bytedance
Shanghai, China
chenqiwei5@qq.com

Feng Zhang
Bytedance
Shanghai, China
feng.zhang@bytedance.com

Yang Xiao
Bytedance
Beijing, China
wuqi.shaw@bytedance.com

Zuotao Liu
Bytedance
Singapore
michael.liu@bytedance.com

Abstract

The escalating complexity of industrial recommendation systems, characterized by diverse user behaviors and cross-domain application scenarios, necessitates advanced multi-task and multi-domain learning paradigms. Existing methods often struggle with efficient knowledge transfer across tasks and domains due to semantic gaps and distribution shifts. To address these challenges, we propose the ***P**erception-Aware **M**ulti-Task **T**ransformer Network for Personalized Multi-Domain **A**daptation* (**PMTA**), a unified framework that integrates three key innovations: First, **the Task Prompt Encoding** (**TPE**) module dynamically generates prompts by synthesizing personalized user data with task-specific information. Second, **the Transformer-based Multi-Task Perception** (**TMPN**) network enables adaptive cross-task knowledge transfer through attention mechanisms. Third, the **Multi-Domain Adaptation** (**MDAN**) component captures domain-specific behavior patterns via learnable prior information. Experimental results demonstrate PMTA's effectiveness, achieving **0.168%** increase in watch time and significant improvements in engagement metrics (AAD: **+0.0113%**, AAH: **+0.0608%**). Deployed on Douyin and Douyin Lite, it significantly improves recommendation quality and drives commercial success.

CCS Concepts

• **Information systems** → **Information retrieval**.

*Equal contribution.

CIKM '25, Seoul, Republic of Korea

ACM ISBN 979-8-4007-2040-6/2025/11
https://doi.org/10.1145/3746252.3760841

Keywords

Multi-Task Learning; Multi-Domain Adaptation; Transformer; Recommendation Systems

ACM Reference Format:
Chenbin Zhang, Xiaoxie Zhu, Xingchao Cao, Qiwen Chen, Feng Zhang, Yang Xiao, and Zuotao Liu. 2025. PMTA: Perception-Aware Multi-Task Transformer Network for Personalized Multi-Domain Adaptation. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management (CIKM '25), November 10–14, 2025, Seoul, Republic of Korea.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3746252.3760841

1 Introduction

Traditional recommendation systems focus on single-task modeling in isolated scenarios, but modern applications require multi-task learning across diverse domains [1, 12, 16]. E-commerce and short video platforms generate domain-specific data with varying user behaviors—e.g., price-driven purchases in e-commerce vs. trend-driven engagement in social media (**multi-domains problem**). Taobao illustrates sequential scenarios such as homepage "Guess What You Like," product page "Choose Again," and post-purchase recommendations. Douyin spans both its main and Lite apps. Both platforms require modeling cross-task interactions (e.g., clicks, purchases, likes, shares), highlighting the **multi-tasks problem**.The challenges of domain heterogeneity and multi-task objectives motivate PMTA, a framework that uses task prompts and cross-task attention to integrate cross-domain knowledge and capture task-specific preferences.

For industrial recommender systems, multi-domain learning aims to align sample spaces across scenarios (**domain seesaw**[11]) but neglects inter-task correlations, while multi-task learning focuses on fitting task-specific target distributions (**task seesaw**[14]) but overlooks cross-domain semantic inconsistencies. In real-world applications, these challenges manifest as the **imperfect double seesaw phenomenon**[2], with varying semantic features and target sparsity across domains and tasks. Joint multi-task and multi-domain optimization is thus critical, often requiring personalized user priors and inter-task dependency modeling. We propose PMTA,

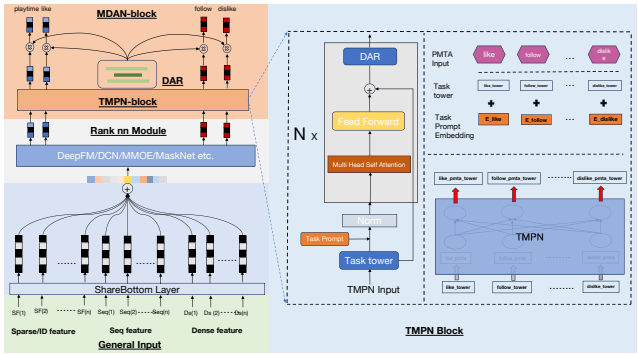

Figure 1: The architecture of PMTA, comprising three components: (1) TPE for task-aware prompt generation, (2) TMPN for cross-task knowledge interaction via self-attention, and (3) MDAN for domain-specific representation adaptation.

a plug-and-play Transformer framework that models task correlations, adapts via task prompts, and integrates user/domain priors through gating. Compared with SOTA methods in multi-task ([2, 8–10, 17]) and multi-domain ([2, 7, 11]) learning, PMTA features high efficiency and broad adaptability. The contributions of this work can be summarized as follows:

- PMTA is a lightweight model embeddable into any recommender system, mitigating the double seesaw effect in industrial settings.
- Deployed on Douyin and Douyin Lite, PMTA achieves (+0.168% /+0.231%) gains in Watch Time, with consistent improvements in AAD (Average Active Days) (+0.0113%/+0.0147%) and AAH (Average Active Hours) (+0.0608%/+0.0647%). These results fully validate the effectiveness and commercial value of the approach in large-scale industrial scenarios.

2 Method

2.1 Problem Formulation

We model multi-task multi-domain recommendation as a joint optimization problem across domains and tasks, aiming to learn a shared representation that balances cross-domain/task correlations and domain/task-specific preferences. The objective is to minimize prediction losses across all domain-task pairs with parameter regularization for efficient knowledge transfer.

Let $\mathcal{U}$ and $\mathcal{V}$ denote the user set and video set, respectively. The system contains D distinct domains ($d \in \{1, 2, \ldots, D\}$), each corresponding to a unique recommendation scenario. Users $u \in \mathcal{U}$ interact with videos across domains, with the core task of predicting user preferences for multiple task within a specific domain. For each domain d, the model learns **domain-specific embeddings** for users/videos while capturing **shared embeddings** across domains: **Shared Embeddings**: Users and videos are represented by global vectors $\mathbf{e}_u^g, \mathbf{e}_v^g$, learned across all domains. **Domain-Specific Embeddings**: In domain d, users and videos are represented by exclusive vectors $\mathbf{e}_u^d, \mathbf{e}_v^d$, learned only within domain d.

Given the neural network H, the classifier C^t for task t, the user u, the domain d, the multitask prediction for task $t \in \{1, 2, \ldots, T\}$ can be expressed as follows:

$$\hat{y}_{u,i}^{d,t} = C^t\left(H\left(\mathbf{e}_u^g, \mathbf{e}_u^d, \mathbf{e}_v^g, \mathbf{e}_v^d, \mathbf{e}_o\right)\right)$$

where $\mathbf{e}_o$ is the embedding of other features, including context features and combined features.

2.2 Overview of PMTA

We propose **PMTA**, a plug-and-play Transformer-based architecture for multi-task recommendation. As a generic component compatible with any feature interaction module H, PMTA operates at the classifier layer C with three key components (Fig. 1):

- **Task Prompt Embedding (TPE)**: Generates task-specific embeddings from user/video features and metadata to enrich Transformer inputs.
- **Transformer-based Multi-Task Perception Network (TMPN)**: Core module using self-attention to enable cross-task interaction among hidden outputs of classifiers C_t.
- **Multi-Domain Adaptation Network (MDAN)**: Generates domain-specific scaling weights for TMPN from domain embeddings.

2.3 Task Prompt Embedding

Inspired by HyperPrompt [5], PMTA enriches task representations via **Task Prompt Embedding (TPE)**, which extends HyperPrompt's learnable parameters by incorporating task-related feature embeddings (e.g., user action sequences and user/video statistics). This augmentation with personalized task signals enhances TPE quality. TPE employs a two-layer neural network H_{TPE}:

$$\mathbf{x}_{TPE}^t = ReLU\left(Concat(\mathbf{p}^t, \mathbf{e}_u^t, \mathbf{e}_v^t)\mathbf{W}_1^t + \mathbf{b}_1^t\right), \quad (1)$$

$$H_{TPE}^t\left(\mathbf{p}^t, \mathbf{e}_u^t, \mathbf{e}_v^t\right) = \mathbf{x}_{TPE}\mathbf{W}_2^t + \mathbf{b}_2^t, \quad (2)$$

where $\mathbf{p}^t$ is a learnable vector for task t, $\mathbf{e}_u^t$ is the embedding of user features related to task t, $\mathbf{e}_v^t$ is the embedding of video features related to task t, and $\mathbf{W}_1^t$, $\mathbf{b}_1^t$, $\mathbf{W}_2^t$, $\mathbf{b}_2^t$ are the learnable parameters of the neural network H_{TPE} for task t. We simply choose $ReLU$ as the non-linear activation function.

2.4 Transformer-based Multi-Task Perception Network

Transformers' attention mechanism aligns well with our needs, enabling effective cross-task interaction through embedding queries. Thus, we adopt it as the core of our PMTA framework.

Figure 1 illustrates the overall architecture of the Transformer-based Multi-Task Perception Network (TMPN). The network takes as input a task embedding matrix $\mathbf{M}$, composed of individual task embeddings $\mathbf{m}^t$. For each task t, the embedding is represented as:

$$\mathbf{m}^t = \text{Concat}\left(H_{\text{TPE}}^t\left(\mathbf{p}^t, \mathbf{e}_u^t, \mathbf{e}_v^t\right), C_1^t\left(H\left(\mathbf{e}_u^g, \mathbf{e}_u^d, \mathbf{e}_v^g, \mathbf{e}_v^d, \mathbf{e}_o\right)\right)\right) \quad (3)$$

where H_{TPE}^t generates task prompt embeddings, and C_1^t denotes the first-layer output of the task classifier. Considering the substantial size of backbone network H and PMTA's plug-in architecture design, we initialize the model through parameter warm-up, enabling C_1^t outputs to effectively encode task-specific information.

Given the task embedding matrix $\mathbf{M}$, we model cross-task interactions via a L-layer Transformer [15]. Each layer integrates multi-head self-attention (MHA) and a feed-forward network (FFN), defined as follows:

$$\text{MHA}(\mathbf{X}) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\mathbf{W}^O, \quad (4a)$$

$$\text{head}_i = \text{Attention}(\mathbf{X}\mathbf{W}_i^Q, \mathbf{X}\mathbf{W}_i^K, \mathbf{X}\mathbf{W}_i^V), \quad (4b)$$

where the scaled dot-product attention is:

$$\text{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}. \tag{5}$$

Here, $\mathbf{W}_i^Q \in \mathbb{R}^{d_x \times d_k}$, $\mathbf{W}_i^K \in \mathbb{R}^{d_x \times d_k}$, $\mathbf{W}_i^V \in \mathbb{R}^{d_x \times d_v}$, and $\mathbf{W}^O \in \mathbb{R}^{hd_v \times d_x}$ are projection matrices. The feed-forward network is:

$$\text{FFN}(\mathbf{X}) = \text{ReLU}(\mathbf{X}\mathbf{W}_1^{FFN} + \mathbf{b}_1^{FFN})\mathbf{W}_2^{FFN} + \mathbf{b}_2^{FFN}. \tag{6}$$

For cross-task transformer (CTT) with residual connections:

$$\mathbf{X}_i^{SA} = \text{LN}\left(\text{MHA}(\mathbf{X}_i^{CTT}) + \mathbf{X}_i^{CTT}\right), \tag{7a}$$

$$\mathbf{X}_{i+1}^{CTT} = \text{LN}\left(\text{FFN}(\mathbf{X}_i^{SA}) + \mathbf{X}_i^{SA}\right), \tag{7b}$$

with $\mathbf{X}_1^{CTT} = \text{LN}(\mathbf{M})$. We set $L = 2$ layers in this work.

2.5 Multi-Domain Adaptation Network

In Sections 2.3 and 2.4, we have addressed the multi-task learning problem via task-specific prompt embeddings and cross-task Transformer interactions. For the multi-domain challenge, we draw inspiration from conditional normalization techniques (e.g., LHUC [13], PEPNet [2]) to design a Multi-Domain Adaptation Network (**MDAN**). This gating mechanism takes domain embeddings as input and generates per-layer scaling factors to adaptively modulate the intermediate outputs of the Transformer, thereby enhancing domain-specific feature representation while preserving cross-domain shared knowledge.

MDAN employs Domain-Aware Rescaling (**DAR**), a two-layer bottleneck network that adapts layer outputs to domain-specific characteristics. DAR first projects concatenated user and item embeddings $\mathbf{e}_u^d, \mathbf{e}_v^d$ into a lower-dimensional space and then expands them to the target dimension:

$$\mathbf{X}^{DAR} = \text{ReLU}\left(\text{Concat}(\mathbf{e}_u^d, \mathbf{e}_v^d)\mathbf{W}_1^{DAR} + \mathbf{b}_1^{DAR}\right), \tag{8a}$$

$$\text{DAR}(\mathbf{e}_u^d, \mathbf{e}_v^d) = 2 \cdot \text{Sigmoid}\left(\mathbf{X}^{DAR}\mathbf{W}_2^{DAR} + \mathbf{b}_2^{DAR}\right), \tag{8b}$$

where $\mathbf{W}_1^{DAR} \in \mathbb{R}^{n \times k}$, $\mathbf{W}_2^{DAR} \in \mathbb{R}^{k \times m}$, and $\mathbf{b}_1^{DAR}, \mathbf{b}_2^{DAR}$ are learnable parameters. Here, n is the input dimension, m the candidate embedding dimension, and k the bottleneck dimension ($k < n, k < m$). This bottleneck structure reduces parameters while maintaining performance. MDAN integrates Domain-Aware Rescaling (DAR) across all layers of PMTA to enable domain-specific adaptation of intermediate and final outputs. For the i-th layer of the cross-task transformer, the rescaled output is computed as:

$$\text{CTT}_i^{\text{rescale}}(\mathbf{X}_i^{\text{CTT}}) = \text{CTT}_i(\mathbf{X}_i^{\text{CTT}}) \odot \text{DAR}(\mathbf{e}_u^d, \mathbf{e}_v^d), \tag{9}$$

where $\odot$ denotes element-wise multiplication, allowing adaptive modulation of representations based on domain characteristics.

3 Experiment

3.1 Experimental Settings

3.1.1 ***Datasets and Metrics.*** We evaluate PMTA on two industrial domains from Douyin[1]: the main feeds tab (Domain A) and the Lite feeds tab (Domain B). The dataset spans October 2-15, 2024, with users/items filtered to have ≥10 interactions. We predict six binary tasks: *Like, Dislike, Follow, Comment, Headsildeleft, Finish* (Finish=1 if watch time ≥100% percentile). Data is split into 10-day training, 2-day validation, and 2-day testing sets. Table 3 summarizes key statistics: **Task Sparsity** varies significantly, with Finish showing the highest positive rate (24.17–24.62%) and sparse tasks like Follow (0.09–0.11%) and Dislike (0.04–0.06%). **Cross-Domain Overlap** features high user overlap (67.8–87.2%) but low item overlap (23.2–30.1%), reflecting distinct user preferences across domains. We use AUC and GAUC [21] for evaluation.

3.1.2 ***Compared Models.*** To evaluate PMTA, we compare it with models representing three key paradigms in recommendation systems. For **single-domain single-task scenarios**, we include **DeepFM** [4], which fuses Factorization Machines and deep neural networks to model low- and high-order feature interactions. **DCN** [18] and **DCNv2** [19] extend this by incorporating Cross Networks for explicit feature crossing, with DCNv2 optimizing scalability via low-rank factorization. In the **single-domain multi-task setting**, we consider **SharedBottom**, a baseline that shares bottom-layer DNN parameters across tasks while using task-specific output layers. **MMoE** [9] introduces task-specific gating networks to select from shared expert modules, while **PLE** [14] enhances this by separating shared and task-specific experts for dynamic resource allocation. **DCNv2-MT** adapts DCNv2 for multi-task learning through task-specific parameter customization. For **multi-domain multi-task learning**, **PLE-MD** extends PLE by sharing input embedding layers across domains to facilitate cross-domain knowledge transfer. **SharedTop** and **SpecificTop** represent architectures that share top-layer DNN or embeddings, respectively, with the former keeping domain-specific bottom layers and the latter introducing domain-specific task towers. **SpecificAll** fully specializes both embeddings and task layers for each domain. Finally, **PEPNet** [2] captures personalized patterns by leveraging domain-specific embeddings and task-specific parameterization.

3.1.3 ***Implementation Details.*** All models are implemented in TensorFlow with Xavier initialization [6] and the Adam optimizer [3] (learning rate 0.001, batch size 1500). For PMTA, we set 4 heads and 256-dimensional embeddings in TMPN. To ensure fairness, prior information in PMTA is included as additional inputs to the embedding layers of all baselines.

3.2 Overall Performance

Table 1 compares model performance on six tasks, two domains. **Multi-Task Superiority**: PMTA outperforms PEPNet [2] (the prior SOTA) across all metrics, achieving AUC gains of **0.002–0.005** (e.g., Like task in Domain A: 0.9634 vs. PEPNet's 0.9589) and GAUC improvements up to 0.007. Notably, a **0.001 AUC** in sparse tasks is already statistically significant ($p < 0.05$) in recommendation systems, where even marginal gains reflect meaningful user behavior optimization. These results confirm PMTA's effectiveness in mitigating the task seesaw effect. **Cross-Domain Robustness**: Despite domain disparities (e.g., Like sparsity: 2.3% in A vs. 2.91% in B), PMTA maintains balanced performance: AUC of 0.9588 (Like) and 0.9867 (Comment) in Domain B. Its shared-bottom architecture with task-specific towers enables knowledge transfer even at low

[1]The above statistics are derived from sampled data.

Table 1: Model performance on Domain A (Douyin) and B (Douyin Lite) for six tasks: AUC and GAUC (full precision).

Method	Domain A												Domain B											
	Like		Follow		Comment		Dislike		Headslideleft		Finish		Like		Follow		Comment		Dislike		Headslideleft		Finish	
	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC	AUC	GAUC
DeepFM	0.9598	0.7713	0.9601	0.8343	0.9811	0.8885	0.9433	0.8103	0.9016	0.8211	0.7857	0.8308	0.9564	0.7653	0.9513	0.8014	0.9848	0.8872	0.9528	0.7988	0.9094	0.8195	0.8708	0.8439
DCN	0.9601	0.7701	0.9621	0.8348	0.9804	0.8872	0.9411	0.8101	0.9012	0.8208	0.7792	0.8304	0.9553	0.7668	0.9502	0.8098	0.9843	0.8881	0.9527	0.7978	0.9092	0.8187	0.8705	0.8440
xDeepFM	0.9602	0.7715	0.9611	0.8346	0.9813	0.8887	0.9431	0.8122	0.9018	0.8213	0.7862	0.8311	0.9562	0.7664	0.9514	0.8054	0.9849	0.8876	0.9531	0.7996	0.9101	0.8192	0.8711	0.8441
DCNv2	0.9604	0.7703	0.9625	0.8351	0.9802	0.8874	0.9413	0.8109	0.9014	0.8213	0.7796	0.8310	0.9558	0.7671	0.9511	0.8102	0.9848	0.8892	0.9532	0.7983	0.9098	0.8193	0.8709	0.8442
DCNv2-MT	0.9603	0.7705	0.9631	0.8353	0.9809	0.8892	0.9415	0.8113	0.9013	0.8215	0.7803	0.8313	0.9562	0.7674	0.9514	0.8121	0.9861	0.8903	0.9536	0.7992	0.9111	0.8203	0.8711	0.8445
SharedBottom	0.9585	0.7655	0.9595	0.8314	0.9801	0.8791	0.9411	0.7901	0.9010	0.8189	0.7756	0.8299	0.9556	0.7597	0.9464	0.8076	0.9843	0.8860	0.9368	0.7729	0.9092	0.8129	0.8695	0.8436
MMoE	0.9601	0.7753	0.9603	0.8347	0.9811	0.8821	0.9432	0.8213	0.9030	0.8218	0.7801	0.8325	0.9561	0.7612	0.9469	0.8103	0.9850	0.8911	0.9431	0.7988	0.9097	0.8233	0.8711	0.8453
PLE	0.9605	0.7762	0.9606	0.8352	0.9813	0.8832	0.9441	0.8232	0.9080	0.8232	0.7833	0.8338	0.9565	0.7632	0.9472	0.8133	0.9853	0.8923	0.9445	0.8001	0.9112	0.8239	0.8715	0.8455
PLE-MD	0.9608	0.7773	0.9611	0.8377	0.9815	0.8845	0.9449	0.8239	0.9110	0.8239	0.7855	0.8341	0.9571	0.7645	0.9476	0.8143	0.9859	0.8944	0.9448	0.8034	0.9117	0.8276	0.8718	0.8459
SharedTop	0.9579	0.7833	0.9621	0.8422	0.9819	0.8887	0.9501	0.8242	0.9118	0.8261	0.7863	0.8332	0.9576	0.7685	0.9469	0.8187	0.9853	0.8953	0.9442	0.8076	0.9134	0.8288	0.8726	0.8464
SpecificTop	0.9571	0.7822	0.9633	0.8368	0.9803	0.8862	0.9472	0.8232	0.9112	0.8238	0.7843	0.8325	0.9572	0.7648	0.9461	0.8176	0.9847	0.8945	0.9414	0.8064	0.9128	0.8281	0.8721	0.8469
SpecificAll	0.9575	0.7801	0.9619	0.8362	0.9809	0.8833	0.9487	0.8228	0.9012	0.8232	0.7783	0.8321	0.9533	0.7642	0.9433	0.8095	0.9842	0.8922	0.9409	0.8045	0.9119	0.8274	0.8718	0.8463
PEPNet	0.9611	0.7871	0.9643	0.8452	0.9821	0.8923	0.9521	0.8249	0.9123	0.8333	0.7877	0.8347	0.9575	0.7719	0.9538	0.8218	0.9865	0.8977	0.9535	0.8085	0.9167	0.8301	0.8736	0.8477
PMTA	**0.9634**	**0.7889**	**0.9660**	**0.8497**	**0.9834**	**0.8946**	**0.9571**	**0.8266**	**0.9144**	**0.8347**	**0.7894**	**0.8359**	**0.9588**	**0.7735**	**0.9556**	**0.8288**	**0.9867**	**0.8999**	**0.9542**	**0.8101**	**0.9189**	**0.8307**	**0.8744**	**0.8493**

Table 2: Ablation study of PMTA evaluates key components' impact. Results from Domain A experiments are averaged over five runs.

Variant	Finish AUC	Dislike AUC	Follow AUC	Like AUC	Comment AUC	Headslideleft AUC
PMTA w/o TPE	0.7104	0.8942	0.9060	0.9133	0.9210	0.8136
PMTA use task fusion mlp	0.7494	0.9271	0.9360	0.9484	0.9788	0.8623
PMTA w/o DAR	0.7794	0.9172	0.9452	0.9412	0.9669	0.8832
PMTA	0.7894	0.9571	0.9660	0.9634	0.9834	0.9144

Table 3: Task sparsity and cross-domain overlap.

	Domains	Douyin (A)	Douyin Lite (B)
Task Sparsity	Like	2.3%	2.91%
	Follow	0.09%	0.11%
	Dislike	0.04%	0.06%
	Comment	0.18%	0.15%
	HeadSlideleft	1.35%	1.44%
	Finish	24.62%	24.17%
User Overlap	A→B	-	67.8%
	B→A	87.2%	-
Item Overlap	A→B	-	23.2%
	B→A	30.1%	-

Table 4: Online gains in two representative domains.In Douyin's short-video scenario, a 0.01% Watch Time increase is significant.

	Douyin	Douyin Lite
Watch Time	+0.168%	+0.231%
Dislike	-1.092%	-2.794%
Like	+0.395%	+0.54%
Follow	+0.519%	+0.527%
Comment	+0.280%	+0.538%
Headslideleft	+0.389%	+0.786%
AAH	**+0.0608%**	**+0.0647%**
AAD	**+0.0113%**	**+0.0147%**

user/item overlap (23.2–30.1%), resolving the domain seesaw problem. **Perception-Aware Personalization**: PMTA's Transformer-based TMPN module achieves **1.2–2.1% AUC/GAUC improvements** over PEPNet on critical tasks (e.g., Headslideleft: 0.9144 AUC in Domain A). Notably, a 0.001 gain in Finish task GAUC (e.g., 0.8359 vs. 0.8348) highlights its ability to model subtle user engagement signals without increasing complexity, making it suitable for sparse cross-domain scenarios. Overall, PMTA sets new SOTA results by addressing both task and domain imbalance issues, validating its effectiveness for real-world recommendation systems. We systematically ablated PMTA's core components to quantify their contributions (Table 2):

- **w/o TPE** (Task Prompt Encoding): Removing TPE yields average AUC drops of 8.3% across tasks, with HeadSlideleft degrading from 0.9144 to 0.8136. This confirms that task-specific prompt embeddings are critical for disentangling shared representations and enhancing feature relevance.
- **TMPN→MLP** (Transformer → MLP Fusion): Replacing the Cross-Task Transformer with a 3-layer MLP reduces AUC by 1.2–3.5%, particularly in tasks with high cross-dependency (e.g., Follow and Comment). This indicates self-attention mechanisms are superior for modeling complex task relationships.
- **w/o DAR** (Domain-Aware Rescaling): Removing DAR causes significant performance degradation in domain-sensitive tasks (Dislike: 0.673 → 0.631; Comment: 0.715 → 0.683). These results underscore the importance of domain-specific scaling factors in mitigating cross-domain generalization gaps.

3.3 Online A/B Testing

We evaluated PMTA via A/B testing on Douyin/Douyin Lite, focusing on metrics: *Watch Time*, *Dislike*, *Like*, *Follow*, *Comment*, *Headslideleft*, *AAD*, and *AAH* (Table 4). Following prior work [20], we use **AAD** (active days) and **AAH** (active hours) to assess user experience. PMTA achieves significant improvements across all metrics:

- Watch Time Surges: PMTA boosts user engagement, with Watch Time rising by +0.168% on Douyin and +0.231% on Douyin Lite. This seemingly small increase leads to a significant rise in content consumption, proving its effectiveness in extending user sessions.
- User Activity Soars: AAD grows +0.0113% (Douyin) and +0.0147% (Douyin Lite), while AAH increases +0.0608% and +0.0647%, indicating improved user retention and daily engagement.
- Interaction Dynamics Transformed: Positive behaviors such as Likes, Follows, and Comments increase across platforms, with Likes up +0.395%/+0.54%, Follows +0.519%/+0.527%, and Comments +0.280%/+0.538%. Dislikes drop sharply by -1.092% (Douyin) and -2.794% (Douyin Lite), demonstrating PMTA's ability to align content with user preferences and enhance satisfaction.

4 Conclusion

This paper presents **PMTA**, a perception-aware multi-task Transformer for personalized recommendation. It integrates TPE, TMPN, and MDAN to address cross-task interference and domain shift. Experiments on Douyin/Douyin Lite show PMTA outperforms SOTA on key metrics. Deployed in production, its modular design enables scalable adaptation to diverse scenarios, validating its commercial impact on user retention and watch time.

5 GenAI Usage Disclosure

We hereby declare that all content within this paper was independently authored by us, and all data presented are the genuine results of our experiments. The only usage of GenAI tools in the preparation of this paper was limited to DeepSeek and Grammarly, which were employed solely for the purposes of grammatical verification, spelling correction, and text refinement.

References

[1] Jianxin Chang, Chenbin Zhang, Zhiyi Fu, Xiaoxue Zang, Lin Guan, Jing Lu, Yiqun Hui, Dewei Leng, Yanan Niu, Yang Song, et al. 2023. TWIN: TWo-stage interest network for lifelong user behavior modeling in CTR prediction at kuaishou. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3785–3794.

[2] Jianxin Chang, Chenbin Zhang, Yiqun Hui, Dewei Leng, Yanan Niu, Yang Song, and Kun Gai. 2023. Pepnet: Parameter and embedding personalized network for infusing with personalized prior information. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3795–3804.

[3] P Kingma Diederik. 2014. Adam: A method for stochastic optimization. *(No Title)* (2014).

[4] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 1725–1731.

[5] Yun He, Steven Zheng, Yi Tay, Jai Gupta, Yu Du, Vamsi Aribandi, Zhe Zhao, YaGuang Li, Zhao Chen, Donald Metzler, et al. 2022. Hyperprompt: Prompt-based task-conditioning of transformers. In *International conference on machine learning*. PMLR, 8678–8690.

[6] Siddharth Krishna Kumar. 2017. On weight initialization in deep neural networks. *arXiv preprint arXiv:1704.08863* (2017).

[7] Pan Li and Alexander Tuzhilin. 2020. Ddtcdr: Deep dual transfer cross domain recommendation. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 331–339.

[8] Jiaqi Ma, Zhe Zhao, Jilin Chen, Ang Li, Lichan Hong, and Ed H Chi. 2019. Snr: Sub-network routing for flexible parameter sharing in multi-task learning. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 216–223.

[9] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1930–1939.

[10] Xiao Ma, Liqin Zhao, Guan Huang, Zhi Wang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. 2018. Entire space multi-task model: An effective approach for estimating post-click conversion rate. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1137–1140.

[11] Xiang-Rong Sheng, Liqin Zhao, Guorui Zhou, Xinyao Ding, Binding Dai, Qiang Luo, Siran Yang, Jingshan Lv, Chi Zhang, Hongbo Deng, et al. 2021. One model to serve all: Star topology adaptive recommender for multi-domain ctr prediction. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 4104–4113.

[12] Zihua Si, Lin Guan, ZhongXiang Sun, Xiaoxue Zang, Jing Lu, Yiqun Hui, Xingchao Cao, Zeyu Yang, Yichen Zheng, Dewei Leng, et al. 2024. Twin v2: Scaling ultra-long user behavior sequence modeling for enhanced ctr prediction at kuaishou. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. 4890–4897.

[13] Pawel Swietojanski, Jinyu Li, and Steve Renals. 2016. Learning hidden unit contributions for unsupervised acoustic model adaptation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24, 8 (2016), 1450–1463.

[14] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 269–278.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[16] Maolin Wang, Sheng Zhang, Ruocheng Guo, Wanyu Wang, Xuetao Wei, Zitao Liu, Hongzhi Yin, Yi Chang, and Xiangyu Zhao. 2025. STAR-Rec: Making Peace with Length Variance and Pattern Diversity in Sequential Recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1530–1540.

[17] Qi Wang, Zhihui Ji, Huasheng Liu, and Binqiang Zhao. 2019. Deep bayesian multi-target learning for recommender systems. *arXiv preprint arXiv:1902.09154* (2019).

[18] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & cross network for ad click predictions. In *Proceedings of the ADKDD'17*. 1–7.

[19] Ruoxi Wang, Rakesh Shivanna, Derek Cheng, Sagar Jain, Dong Lin, Lichan Hong, and Ed Chi. 2021. Dcn v2: Improved deep & cross network and practical lessons for web-scale learning to rank systems. In *Proceedings of the web conference 2021*. 1785–1797.

[20] Jing Yan, Liu Jiang, Jianfei Cui, Zhichen Zhao, Xingyan Bin, Feng Zhang, and Zuotao Liu. 2024. Trinity: Syncretizing Multi-/Long-Tail/Long-Term Interests All in One. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 6095–6104.

[21] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1059–1068.