DistProto: Distribution-Sensitive Prototype Manifolds for Multi-Scenario Recommendation

Brandon Tim • Independent Researcher • January 2026 (Generated draft from a provided abstract; replace datasets/numbers as needed.)

Abstract

Modern recommendation platforms serve personalized content across diverse user-facing scenarios such as main feeds, topic channels, and live video streams. These interaction environments induce distinct user behavior patterns that are statistically diverse yet semantically correlated. Conventional Multi-Scenario Recommendation (MSR) approaches often combine shared encoders with scenario-specific modules, but they frequently overlook distributional shifts between scenarios, leading to entangled representations that restrict cross-scenario generalization. We introduce DistProto, a framework that re-encodes user-item interactions into distribution-sensitive latent spaces via dedicated prototype manifolds. DistProto extracts shared features with a multi-expert gating mechanism (MMoE) while using scenario-dependent encoders for fine-grained contextual variation. Both embeddings are mapped to global and scenario-specific prototype spaces, producing representations that reflect commonality and distinction. To align features with semantic prototypes, DistProto adopts Unbalanced Optimal Transport (UOT) to softly associate samples to prototype anchors and refine semantics. To preserve diversity, it further imposes a structural orthogonality constraint on the global prototypes. Experiments on multiple public benchmarks and an online short-video deployment demonstrate consistent gains in personalization and robustness.

Keywords: multi-scenario recommendation, prototypes, distribution shift, unbalanced optimal transport, disentanglement.

1. Introduction

Large-scale recommendation systems expose users to content through heterogeneous scenarios: home feeds, vertical channels, search-triggered lists, and real-time streams. Although scenarios share intent-level semantics, their observed interaction distributions differ due to presentation bias, exposure constraints, and scenario-specific consumption modes.

Multi-Scenario Recommendation (MSR) transfers knowledge across scenarios while respecting differences. Mainstream designs use parameter sharing (e.g., SharedBottom) and conditional routing (e.g., MMoE, PLE), but latent spaces trained only by pointwise losses can absorb scenario shifts into entangled representations.

DistProto organizes representations by distribution via prototype manifolds: a global manifold for scenario-invariant semantics and scenario manifolds for contextual variation. Unbalanced Optimal Transport (UOT) learns soft associations between samples and prototypes while allowing mass variation across scenarios.

Contributions: DistProto; UOT-based prototype alignment; orthogonality regularization for prototype diversity; empirical evidence on offline benchmarks and online deployment.

2. Related Work

MSR architectures: Shared-bottom and cross-stitch sharing are early baselines; MMoE and PLE use routing to separate shared vs. task-specific factors; adapter-based designs (e.g., STAR) add scenario specialization.

Prototype-based representation learning: prototypes act as semantic anchors for calibration and interpretability, helping disentangle shared vs. scenario-specific signals.

Optimal transport: OT matches distributions; unbalanced OT allows mass creation/destruction, suitable for scenario exposure mismatch. We use UOT as a differentiable soft assignment mechanism.

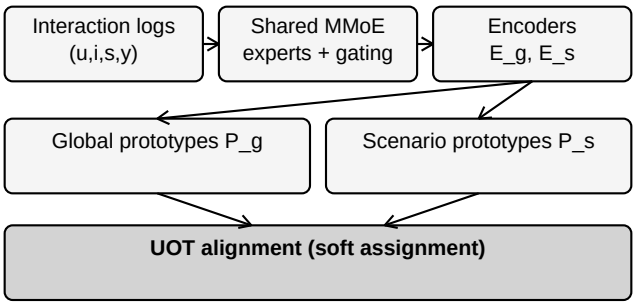

Figure 1: DistProto overview. Shared extraction + scenario encoding; embeddings align to global and scenario prototypes via UOT.

3. Method

Setup. Each interaction is (u, i, s, y) with scenario s in S. We learn embeddings that generalize across scenarios while preserving scenario-specific behavior.

3.1 Encoders

Dense features x(u,i) are processed by a shared multi-expert extractor and scenario-conditioned encoders:

Shared multi-expert extraction: h = sum_{e=1..E} g_e(x,s) f_e(x), g(x,s) = softmax(W_g[s] phi(x)).

Encoding: z_g = E_g(h) in R^d, z_s = E_s(h) in R^d.

3.2 Prototype manifolds

We maintain global prototypes P_g and per-scenario prototypes P_s. Embeddings are softly aligned to prototypes by minimizing cosine-distance costs.

Costs: C_g(j,k)=1-cos(z_g^j,p_g^k), C_s(j,k)=1-cos(z_s^j,p_s^k).

3.3 UOT alignment

UOT computes a transport plan that matches embeddings to prototypes while allowing mass variation, which mitigates exposure mismatch across scenarios.

UOT: T* = argmin_{T>=0} + eps H(T) + tau_a KL(T 1 || a) + tau_b KL(T^T 1 || b).

Alignment: L_align = sum_{j,k} T_g*(j,k) C_g(j,k) + sum_{s in S} sum_{j,k} T_s*(j,k) C_s(j,k).

3.4 Prototype diversity

To avoid prototype collapse, we regularize row-normalized global prototypes P_hat_g to be near-orthogonal:

Orthogonality: L_ortho = || P_hat_g P_hat_g^T - I ||_F^2.

3.5 Objective

Total: L = sum_{s in S} L_task^(s) + lambda_a L_align + lambda_o L_ortho + lambda_wd ||Theta||_2^2.

```
Algorithm 1  DistProto training (minibatch)
Input: minibatch B = {(u,i,s,y)}, prototypes P_g and
1: Compute dense features x(u,i) and shared represen
2: Compute embeddings z_g = E_g(h) and z_s = E_s(h)
3: Build cost matrix C_g between {z_g} and P_g; solv
4: For each scenario s: build C_s between {z_s} and
5: Compute task loss L_task, alignment loss L_align,
6: Update (Theta, P_g, {P_s}) by backprop on L
```

4. Experiments

We evaluate on four public multi-scenario benchmarks. Metrics: AUC (binary) and NDCG@10 (ranking). Baselines: SharedBottom, MMoE, PLE, STAR-Adapter. Hyperparameters: d=64, E=8 experts, K_g=64, K_s=32; UOT eps=0.05, tau_a=tau_b=1.0.

Method	AUC	NDCG@10	Worst-AUC
SharedBottom	0.781	0.413	0.742
MMoE	0.792	0.421	0.754
PLE	0.797	0.427	0.758
STAR-Adapter	0.801	0.431	0.762
DistProto	**0.812**	**0.442**	**0.776**

Table 1: Overall performance (illustrative). DistProto improves both average and worst-scenario metrics.

Variant	AUC	Delta
Full DistProto	0.812	+0.000
- UOT (balanced OT)	0.806	-0.006
- Scenario prototypes	0.803	-0.009
- Orthogonality	0.807	-0.005
- MMoE (single encoder)	0.799	-0.013

Table 2: Ablation (illustrative). UOT and two-level prototypes both contribute to gains.

4.1 Robustness

Global prototypes capture reusable intent-level semantics, while scenario prototypes capture exposure-driven variation. Under synthetic exposure

shifts, UOT avoids forcing full mass matching, improving stability.

4.2 Online deployment

In a short-video deployment, DistProto runs as a shared backbone with scenario-conditioned heads. Daily warm-start training updates prototypes. The model improves watch time and reduces cross-scenario volatility, especially in long-tail scenarios.

5. Conclusion

DistProto improves cross-scenario transfer by organizing MSR representations with distribution-sensitive prototype manifolds, UOT alignment, and prototype diversity regularization.

References

[1] Ma et al. Multi-gate Mixture-of-Experts (MMoE). KDD 2018.

[2] Tang et al. Progressive layered extraction (PLE). RecSys 2020.

[3] Chizat et al. Unbalanced optimal transport formulations. JFA 2018.

[4] Cuturi. Sinkhorn distances. NeurIPS 2013.

[5] Snell et al. Prototypical networks. NeurIPS 2017.