From ID to LLM: Rethinking Representation Learning for Recommendation

Song-Li Wu
Tsinghua University
wsl24@mails.tsinghua.edu.cn

Zhaocheng Du*
Huawei Noah's Ark Lab
zhaochengdu@huawei.com

Weinan Gan
Huawei Noah's Ark Lab
ganweinan1@huawei.com

Jingyi Wang
Tsinghua University
jingyi-w24@mails.tsinghua.edu.cn

Xianquan Wang
University of Science and Technology of China
wxqcn@mail.ustc.edu.cn

Abstract

Recent studies indicate a fundamental incompatibility between ID representations and language model (LM) representations, as they capture behavioral and semantic spaces respectively. This mismatch leads LM representations to consistently underperform ID representations in recommendation tasks. In this work, we revisit this problem and show, from an information-theoretic perspective, that LLM representations retain all discriminative information in ID representations. Based on this, we introduce a Profile-then-Embedding (PtE) framework for recommendation, consisting of a Profile Stage, in which semantic user and item profiles are generated jointly through LLM-based bidirectional reasoning over user-item interactions, and a Personalized Embedding Stage, which encodes these profiles into task-aligned recommendation embeddings. We demonstrate PtE's effectiveness across three benchmark datasets, including cold-start and long-tail scenarios, achieving substantial gains in both discriminative and generative recommendation models.

1 Introduction

Conventional recommender systems rely on ID representations to capture user–item interaction patterns (He et al., 2020; Wu et al., 2023), effectively modeling collaborative filtering signals but lacking semantic transferability, which leads to strong domain dependency and cold-start issues (Chen et al., 2024; Zhu et al., 2021; Wang et al., 2024). In contrast, LLMs provide rich semantic priors that enhance generalization and reasoning (Hu et al., 2024b), yet their representations do not explicitly encode collaborative filtering signals and thus cannot fully replace ID-based collaborative filtering (Hu et al., 2025).

Prior work has explored integrating these two representations to combine their complementary strengths through three main strategies: (i) Semantic reconstruction—randomly initialized ID representations are aligned with language representations via auxiliary modules (Ren et al., 2024); (ii) Semantic initialization—ID representations are initialized with language representations and further updated during training (Wang et al., 2025b; Hu et al., 2024b; Qu et al., 2024); (iii) Adaptive projection—trainable adapters map semantic representations into the behavioral space (Liu et al., 2025; Sheng et al., 2024; Zhang et al., 2024; Yuan et al., 2023). Despite partially injecting semantics, these approaches fail to construct a unified representation space, leaving item semantics and user behaviors misaligned, and still fundamentally rely on ID representations—degrading to random or empty vectors for new users or items, which limits cold-start performance.

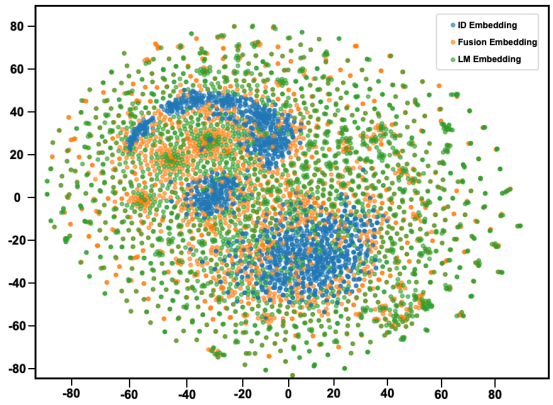

Figure 1: Distribution visualization of ID Embedding, Fusion Embedding and LLM Embedding.

This motivates two key questions: *(1) Can LLM representations fully subsume the collaborative filtering signals encoded by ID embeddings? (2) If so, how can these signals be effectively disentangled and exploited, especially under cold-start settings?* We first conduct an experiment in Figure 1, showing that ID embeddings exhibit a compact distribution, whereas LLM embeddings are more dispersed,

*corresponding author

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3893–3905

with fused embeddings combining both characteristics, indicating a nested relationship. Building on this observation, we formalize a representation sufficiency theory, demonstrating that the collaborative signals captured by ID embeddings are fully representable within the semantic space of LLMs. Moreover, fused embeddings do not contain strictly more information than native LLM embeddings, placing them between ID and LLM representations in the sufficiency hierarchy. Despite the theoretical sufficiency of LLM embeddings, they often underperform empirically. We believe this is because collaborative filtering signals, although encoded within the high-dimensional semantic space, are deeply entangled with semantic information and noise, making them difficult for downstream recommendation models to leverage effectively. This motivates the need for a mechanism that explicitly surfaces and disentangles collaborative signals before embedding, ensuring that these signals are both interpretable and actionable.

To address this gap between theoretical sufficiency and practical effectiveness, we propose a Profile-then-Embedding framework for recommendation. In this framework, LLMs first perform bidirectional reasoning over user-item interactions to generate high-quality semantic profiles, explicitly surfacing collaborative signals that are otherwise entangled with high-dimensional semantic noise. The reasoning process is iterative and interactive, allowing user and item representations to mutually enhance each other, resulting in more accurate and interpretable profiles. These profiles are then encoded into low-dimensional embeddings that are task-aligned and optimized for recommendation objectives, preserving both the collaborative information and personalized characteristics of users and items. By decoupling the reasoning of collaborative signals from embedding generation, our framework effectively bridges the gap between the theoretical representational sufficiency of LLMs and their practical utility in recommendation tasks.

Our contributions are as follows:

- We provide an information-theoretic analysis revealing that LLM embeddings can theoretically capture all collaborative signals in ID embeddings.
- We propose a Profile-then-Embedding framework, where LLMs jointly reason over user-item interactions to generate semantic profiles, which are then encoded into task-aligned embeddings.
- Extensive experiments on real-world benchmarks demonstrate consistent improvements over state-of-the-art methods.

2 Related Work

2.1 Language Model Representations

Large language models (LLMs) have demonstrated remarkable representational capacity, encoding not only lexical and syntactic knowledge (Vulić et al., 2020) but also higher-level abstractions such as color (Patel and Pavlick, 2022), spatial relations (Zha et al., 2025), and game states (Li et al., 2022). These embeddings are often evaluated via techniques such as linear probing (Merullo et al., 2022) or linear mapping (Alain and Bengio, 2016), which reveal that LLM hidden states form a compact, expressive, and highly transferable representation space (Ren et al., 2024; Wang et al., 2025a). Such findings suggest that LLM embeddings can serve as general-purpose representations, with the potential to replace traditional ID embeddings in downstream tasks.

2.2 LLMs for Recommendation

Integrating LLMs into recommender systems has attracted increasing interest, and existing methods can be broadly categorized into three paradigms: **Semantic augmentation:** LM-enhanced recommenders incorporate LLM embeddings alongside traditional ID embeddings (Chen et al., 2025; Singh et al., 2024), capturing fine-grained semantic relationships to improve recommendation quality. **Modality encoding:** LLMs are used as feature extractors to encode textual content, such as item descriptions and reviews, which are then aligned with behavioral data (Lin et al., 2025; Yuan et al., 2023). These approaches still rely primarily on ID embeddings, using LLM features as auxiliary signals. **LM-as-recommender:** Some works reformulate recommendation as a text generation task, prompting or fine-tuning LLMs to generate items directly (Zhang et al., 2025; Tang et al., 2025). While this paradigm achieves competitive results, it faces challenges in scalability and modeling long user-item interaction sequences. Despite these efforts, most methods leverage LLMs indirectly, leaving open the question of whether LLM embeddings alone can serve as a unified backbone for recommendation. Our work aims to address this gap by

exploring fully LM-driven recommendation architectures.

3 Theoretical Foundations

This section revisits the representational relationship among ID embeddings, LLM embeddings, and their fusion from a principled information-theoretic perspective. We ground our analysis in classical results from statistical decision theory, notably *statistical sufficiency* (Casella and Berger, 2024). Our goal is to establish a clear information ordering among representations and to explain why fusion embeddings, despite their empirical utility, cannot exceed the predictive capacity of native LLM embeddings.

3.1 Preliminaries: Statistical Sufficiency and Dominance

We formalize the predictive power of an embedding via the notion of *statistical sufficiency*.

Definition 1 (Statistical Sufficiency). An embedding U is said to be *sufficient* for predicting a downstream variable Y relative to another embedding V if there exists a (possibly stochastic) mapping π such that

$$P(Y \mid V) = \int P(Y \mid U = u)\, \mathrm{d}\pi(u \mid V). \quad (1)$$

Intuitively, this means that V can be simulated from U without any loss of predictive information about Y. This notion induces a partial order over representations, known as the *Blackwell order* (Blackwell, 1953). If U is sufficient for V, then U is said to *Blackwell-dominate* V.

3.2 Information Sufficiency Index

To quantitatively measure predictive informativeness, we adopt the *Information Sufficiency Index (ISI)*. For an embedding U, it is defined as

$$\mathcal{I}_{\mathcal{S}}(U \to V) \triangleq 1 - \frac{\mathbb{E}\big[\inf_{M \in \mathcal{K}_\Theta(V|U)} \mathbb{E}\big[-\log M(V \mid U) \mid U\big]\big]}{\inf_{f \in \mathcal{F}_\Theta(V)} \mathbb{E}\big[-\log f(V)\big]}. \quad (2)$$

Remark. When the conditional model family $\mathcal{K}_\Theta$ is sufficiently expressive, $\mathcal{I}_{\mathcal{S}}$ can be viewed as a normalized form of mutual information. Crucially, it is monotone under Blackwell dominance: if U dominates V, then $\mathcal{I}_{\mathcal{S}}(U \to Y) \geq \mathcal{I}_{\mathcal{S}}(V \to Y)$.

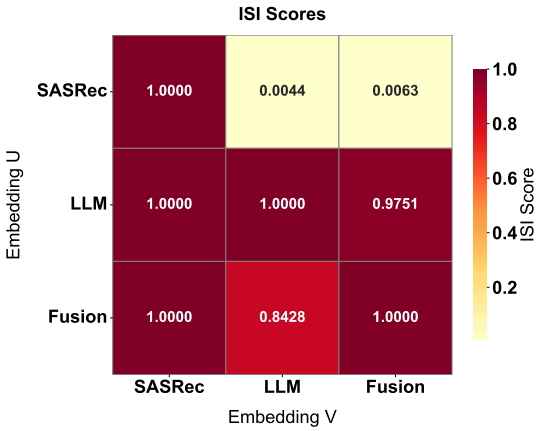

Figure 2: Case study of ID Embedding, Fusion Embedding and LLM Embedding.

3.3 Information Dominance of LLM Embeddings

We now present our main theoretical result and the proof is in Appendix B.

Theorem 1 (Information Dominance of LLM Embeddings)**.** *For any downstream task Y, the following hold:*

$$\mathcal{I}_{\mathcal{S}}(Z_{lm} \to Y) \geq \mathcal{I}_{\mathcal{S}}(Z_{id} \to Y), \quad (3)$$
$$\mathcal{I}_{\mathcal{S}}(Z_{fus} \to Y) \leq \mathcal{I}_{\mathcal{S}}(Z_{lm} \to Y), \quad (4)$$

where $Z_{fus} = g(Z_{id}, Z_{lm})$ denotes any deterministic fusion embedding.

3.4 Case Study

To illustrate the theoretical results in Section 3.3, we present a case study comparing the predictive informativeness of ID embeddings Z_{id}, LLM embeddings Z_{lm}, and their fusion Z_{fus} in Figure 2. Specifically, the ID embeddings are derived from SASRec (Kang and McAuley, 2018), while the LLM embeddings are obtained from LLaMa3-8B (Shui et al., 2024). The fusion embedding Z_{fus} is constructed by processing Z_{id} and Z_{lm} separately through MLPs and then combining them. Our experiments indicate that LLM embeddings effectively preserve all discriminative information contained in the ID embeddings. Furthermore, the fusion embeddings do not contain more predictive information than the LLM embeddings: LLM embeddings account for over 95% of the information in the fusion embeddings, whereas the fusion embeddings only explain less than 85% of the information in the LLM embeddings. These findings empirically corroborate Theorem 1, confirming that

fusion embedding cannot surpass the predictive capacity of LLM embeddings.

4 Methods

We propose Profile-then-Embedding (PtE), a two-stage framework that tightly integrates semantic reasoning from large language models (LLMs) with collaborative signals from user–item interactions. The framework consists of: (i) a *Profile Stage*, which performs bidirectional LLM-based representation generation for users and items, and (ii) a *Personalized Embedding Stage*, which aligns semantic profiles with collaborative structures through preference-aware optimization. This design unifies semantic understanding and collaborative information in a closed-loop manner, enabling robust representation learning under sparse and cold-start scenarios.

4.1 Profile Stage

We introduce a **Bidirectional LLM Representation Generation** mechanism that explicitly semanticizes collaborative information and injects it into the LLM representation space. Formally, our goal is to learn semantic user and item profiles, $\mathbf{p}_u \in \mathcal{T}$ and $\mathbf{p}_v \in \mathcal{T}$, where $\mathcal{T}$ denotes the space of natural language representations generated by an LLM. Unlike conventional approaches that treat users and items independently, we jointly construct $\mathbf{p}_u$ and $\mathbf{p}_v$ through iterative bidirectional generation, enabling mutual enhancement via semantic feedback.

4.1.1 User Representation

User representations form the foundation of the bidirectional profiling framework. For a target user u, we first construct a collaborative neighborhood $\mathcal{N}(u)$ based on historical interaction overlap. Given interaction sets $\mathcal{V}_u$ and $\mathcal{V}_{u'}$, we define the interest overlap ratio as

$$s(u, u') = \frac{|\mathcal{V}_u \cap \mathcal{V}_{u'}|}{|\mathcal{V}_u|}, \tag{5}$$

and select users ranked highest according to $s(u, u')$ to form $\mathcal{N}(u)$. Conditioned on the interaction history of u and its collaborative neighbors, an LLM $\mathcal{M}$ generates a semantic user profile

$$\mathbf{p}_u = \mathcal{M}\big(\mathcal{V}_u,\ \{\mathcal{V}_{u'} \mid u' \in \mathcal{N}(u)\}\big), \tag{6}$$

where $\mathbf{p}_u$ summarizes latent preferences and behavioral patterns in natural language. We fine-tune a base LLM(LLaMa3-8B (Shui et al., 2024)) on such generated profiles, enabling the model to internalize collaborative signals and produce *collaboratively enhanced* user representations even under sparse observations.

4.1.2 Item Representation

Items often lack sufficient textual descriptions. To address this limitation, we infer item semantics from the users who prefer them. For an item v, let $\mathcal{U}(v)$ denote the set of users who have interacted with it. We generate the item profile by conditioning the LLM on the semantic user profiles of $\mathcal{U}(v)$:

$$\mathbf{p}_v = \mathcal{M}\big(\{\mathbf{p}_u \mid u \in \mathcal{U}(v)\},\ x_v\big), \tag{7}$$

where x_v denotes optional item-side metadata. The resulting $\mathbf{p}_v$ captures shared semantic characteristics induced by collaborative user behavior. The generated item profiles are then fed back as contextual signals for refining user profiles, enabling mutual enhancement between user and item representations.

4.1.3 Bidirectional Closed-Loop Optimization

User and item profile generation are integrated into a bidirectional closed-loop process. Starting from an initial user profile $\mathbf{p}_u^{(0)}$, we iteratively update user and item representations as

$$\begin{aligned} \mathbf{p}_v^{(t)} &= \mathcal{M}_v\big(\{\mathbf{p}_u^{(t)} \mid u \in \mathcal{U}(v)\}\big), \\ \mathbf{p}_u^{(t+1)} &= \mathcal{M}_u\big(\mathcal{V}_u,\ \{\mathbf{p}_v^{(t)} \mid v \in \mathcal{V}_u\}\big), \end{aligned} \tag{8}$$

where $\mathcal{M}_u$ and $\mathcal{M}_v$ denote user- and item-conditioned generation operators. This iterative process is repeated for a bounded number of iterations until the representations exhibit negligible changes across successive steps, yielding final profiles $\mathbf{p}_u^*$ and $\mathbf{p}_v^*$. The loop naturally provides a self-supervised learning signal by encouraging consistency between user-induced and item-induced semantics, unifying collaborative structure and textual meaning in a shared representation space. The convergence analysis is provided in Appendix B.2.

4.2 Personalized Embedding

Although LLM-based profiling captures rich semantic preferences, it often leads to homogenized representations across users and items due to shared prompts and global optimization objectives. To preserve individual specificity while maintaining collaborative consistency, we adopt Group Relative Policy Optimization (GRPO; (Shao et al., 2024)). For clarity, we describe the optimization process

for users, with an analogous formulation applied to items.

Group-wise Profile Sampling. For each user u, we sample a group of candidate semantic profiles $\{\mathbf{p}_u^{(k)}\}_{k=1}^{K}$ by varying prompts, decoding temperatures, or random seeds. Each profile $\mathbf{p}_u^{(k)}$ is a natural-language description generated by the LLM.

Reward Decomposition. Instead of optimizing absolute rewards, GRPO (Shao et al., 2024) optimizes relative preference ordering within each group. The reward of a candidate profile $\mathbf{p}_u$ is defined as

$$R(\mathbf{p}_u) = \lambda_c \, R_{\text{collab}}(\mathbf{p}_u) + \lambda_d \, R_{\text{dist}}(\mathbf{p}_u), \quad (9)$$

where R_{collab} enforces alignment with collaborative signals, and R_{dist} encourages personalization via inter-user distinctiveness.

Collaborative Alignment Reward. Let $\mathcal{I}(u)$ denote the historical interacted items of user u, and $f_{\text{enc}}(\cdot)$ be a text encoder shared across users. We define the collaborative reward as

$$R_{\text{collab}}(\mathbf{p}_u) = \frac{1}{|\mathcal{I}(u)|} \sum_{i \in \mathcal{I}(u)} \cos(f_{\text{enc}}(\mathbf{p}_u), f_{\text{enc}}(i)), \quad (10)$$

which encourages the semantic profile to cover the user's historical interests in the embedding space.

Distinctiveness Reward. Let $\mathcal{N}(u)$ denote a neighborhood of users with similar interaction histories (e.g., based on ID embeddings). To discourage profile collapse, we define

$$R_{\text{dist}}(\mathbf{p}_u) = -\frac{1}{|\mathcal{N}(u)|} \sum_{v \in \mathcal{N}(u)} \cos(f_{\text{enc}}(\mathbf{p}_u), \bar{\mathbf{e}}_v), \quad (11)$$

where $\bar{\mathbf{e}}_v$ denotes the current embedding of neighboring user v. This term penalizes semantic similarity to nearby users, thereby promoting individualized representations.

Group-Relative Optimization Objective. Given the group of profiles for user u, we compute the group-average reward $\bar{R}_u = \frac{1}{K}\sum_{k=1}^{K} R(\mathbf{p}_u^{(k)})$. The GRPO (Shao et al., 2024) objective maximizes the expected relative advantage:

$$\max_{\mathcal{M}} \mathbb{E}_u \left[\sum_{k=1}^{K} \log \sigma\Big(R(\mathbf{p}_u^{(k)}) - \bar{R}_u \Big) \right], \quad (12)$$

where $\sigma(\cdot)$ is the sigmoid function. This group-relative formulation explicitly suppresses mode collapse while preserving collaborative consistency.

Embedding Extraction. After optimization, the final personalized user embedding is obtained by encoding the highest-ranked profile:

$$\mathbf{z}_u = f_{\text{enc}}\left(\arg\max_{\mathbf{p}_u^{(k)}} R(\mathbf{p}_u^{(k)}) \right). \quad (13)$$

The resulting embedding $\mathbf{z}_u$ is used as the semantic user representation in downstream recommendation models.

5 Experiment

5.1 Experimental Settings

Table 1: Statistics of datasets.

Dataset	# users	# items	# Interactions	Density
Movies	20,515	44,014	637,157	99.93%
Toys	19,412	11,924	138,444	99.94%
Sports	35,598	18,357	256,598	99.96%

5.1.1 Datasets

We evaluate PtE on three real-world Amazon datasets. The Movies dataset (Hou et al., 2024) contains detailed metadata and user reviews spanning from June 1996 to September 2023. The Toys dataset (McAuley et al., 2015) covers user reviews and product information for toys and games between June 1996 and July 2014. The Sports dataset (McAuley et al., 2015) includes reviews and metadata for sports and outdoor products within the same period. The overall statistics of the processed datasets are summarized in Table 1 and Table 2.

Table 2: The statistics of datasets

Dataset	# Users	# Items	Sparsity	Avg.length
Yelp	15,720	11,383	99.89%	12.23
Fashion	9,049	4,722	99.92%	3.82
Beauty	52,204	57,289	99.99%	7.57

5.1.2 Experimental Settings

For a thorough evaluation, we consider two experimental protocols: **leave-one-out** and **cold-start user** settings. **Leave-One-Out Setting** (Kang and McAuley, 2018): In this protocol, the final item in

each user's interaction history is reserved for testing, the penultimate item is used for validation, and the remaining interactions constitute the training set. This setup is widely adopted in sequential recommendation tasks. However, it assumes that every user appears in the training data, which makes it unsuitable for evaluating cold-start scenarios where users have very limited historical interactions. To address this limitation, we also adopt the **cold-start user setting** (Yang et al., 2023). Here, users are chronologically sorted by the timestamp of their last interaction, and the most recent users are held out for validation and testing, enabling a more realistic evaluation of recommendation performance under cold-start conditions.

5.1.3 Backbones and Implementations

To demonstrate the versatility of PtE, we evaluate it on two representative sequential recommendation models: **SASRec** (Kang and McAuley, 2018) and **DreamRec** (Yang et al., 2023). SASRec exemplifies the traditional discriminative paradigm, whereas DreamRec embodies the generative paradigm via diffusion models. For SASRec, we adopt the InfoNCE loss (Oord et al., 2018) with 64 negative samples, while for DreamRec, we employ the standard MSE loss (Ho et al., 2020) commonly used in Gaussian-based diffusion frameworks. By spanning both discriminative and generative paradigms, this setup highlights the generality of PtE and its adaptability to diverse modeling frameworks.

5.1.4 Baselines

We compare PtE with representative language-guided ID embedding learning approaches, covering semantic initialization, semantic reconstruction, adaptive projection, and hybrid alignment strategies. RLMRec (Ren et al., 2024) is adapted to sequential recommendation with two variants: *RLMRec-Gen*, which maps language embeddings into the behavioral space, and *RLMRec-Con*, which aligns ID embeddings to semantic space. LLMInit (Harte et al., 2023; Hu et al., 2024b; Qu et al., 2024) directly initializes ID embeddings with language embeddings, where PCA is applied for dimensional alignment. MoRec (Yuan et al., 2023) and UniSRec (Hou et al., 2022) employ adaptor-based transformations to bridge semantic and behavioral representations, with UniSRec further leveraging a mixture-of-experts mechanism. WhitenRec (Zhang et al., 2024; Su et al., 2021) decorrelates language embeddings via whitening before projection. LLM-ESR (Liu et al., 2024) integrates semantic initialization and adaptive projection under a dual-view modeling framework to address long-tail challenges. iDreamRec (Hu et al., 2024a) adopts diffusion-based modeling to learn consistent language embeddings for recommendation. Finally, AlphaFuse (Hu et al., 2025) fuses LLM and ID embeddings via SVD-based space decomposition, enabling plug-and-play integration across both discriminative and generative recommenders.

5.2 Cold-start Evaluation

We evaluate the cold-start generalization of **Profile-then-Embedding (PtE)** under both discriminative and generative sequential backbones, instantiating it on **SASRec** and **DreamRec** (Table 3). Across all datasets and metrics, **PtE** consistently outperforms strong baselines such as AlphaFuse, LLMInit, and RLMRec, with particularly large gains on the sparse and long-tailed *Toys* and *Sports* datasets. This highlights PtE's effectiveness in cold-start regimes with limited behavioral supervision. We further observe that many language-enhanced methods suffer substantial degradation under the generative **DreamRec** backbone, despite competitive performance with discriminative models. This reveals a key limitation of directly injecting LLM-derived semantics: under sparse supervision, early semantic commitment can cause misalignment that is difficult to correct and is amplified by autoregressive generation. In contrast, **PtE** remains robust across both backbones due to its profile-first design, which introduces semantic profiles as an explicit intermediate abstraction. By enabling LLM reasoning at the profile level while deferring task alignment to embedding learning, PtE decouples semantic reasoning from behavioral adaptation, reducing irreversible semantic–behavioral misalignment. Overall, these results suggest that effective cold-start recommendation benefits from a principled abstraction barrier rather than tighter semantic fusion, with PtE enabling stable knowledge transfer across data sparsity and modeling paradigms.

5.3 Leave-One-Out Evaluation on Long-Tail Users

To further examine robustness under long-tail distributions, we evaluate our **Profile-then-Embedding (PtE)** framework using a leave-one-out (LOO) protocol following LLM-ESR (Liu et al., 2024). We

Table 3: Performance comparison across different backbones and methods on three datasets with cold-start user settings. Boldface indicates the highest score, while underlining denotes the second-best result among the models.

Backbone	Method	Movies				Toys				Sports			
		N@10	M@10	N@20	M@20	N@10	M@10	N@20	M@20	N@10	M@10	N@20	M@20
SASRec	Base	0.0338	0.0238	0.0429	0.0263	0.0255	0.0191	0.0321	0.0210	0.0073	0.0049	0.0101	0.0057
	MoRec	0.0154	0.0105	0.0205	0.0119	0.0114	0.0069	0.0146	0.0078	0.0098	0.0074	0.0109	0.0077
	UniSRec	0.0232	0.0160	0.0303	0.0179	0.0271	0.0191	0.0311	0.0202	0.0071	0.0051	0.0084	0.0055
	WhitenRec	0.0168	0.0116	0.0223	0.0131	0.0258	0.0181	0.0304	0.0194	0.0115	0.0081	0.0141	0.0088
	RLMRec-Con	0.0346	0.0244	0.0441	0.0269	0.0266	0.0185	0.0304	0.0195	0.0089	0.0058	0.0107	0.0063
	RLMRec-Gen	0.0355	0.0252	0.0449	0.0278	0.0303	0.0246	0.0347	0.0257	0.0080	0.0054	0.0102	0.0060
	LLMInit	0.0370	0.0264	0.0470	0.0291	0.0275	0.0215	0.0313	0.0225	0.0083	0.0055	0.0102	0.0060
	LLM-ESR	0.0139	0.0094	0.0192	0.0108	0.0122	0.0104	0.0153	0.0112	0.0101	0.0075	0.0118	0.0079
	AlphaFuse	0.0459	0.0324	0.0574	0.0355	0.0339	0.0287	0.0376	0.0297	0.0137	0.0098	0.0158	0.0104
	PtE	**0.0523**	**0.0436**	**0.0689**	**0.0472**	**0.0473**	**0.0398**	**0.0511**	**0.0435**	**0.0179**	**0.0145**	**0.0236**	**0.0137**
	Best Impr.	+13.94%	+34.57%	+20.03%	+32.96%	+39.53%	+38.68%	+35.90%	+46.46%	+30.66%	+47.96%	+49.37%	+31.73%
DreamRec	Base	0.0016	0.0013	0.0018	0.0014	0.0383	0.0333	0.0392	0.0336	0.0158	0.0132	0.0170	0.0135
	iDreamRec	0.0226	0.0180	0.0262	0.0189	0.0350	0.0301	0.0373	0.0307	0.0141	0.0119	0.0155	0.0123
	MoRec	0.0002	0.0002	0.0003	0.0002	0.0030	0.0026	0.0034	0.0027	0.0012	0.0010	0.0017	0.0012
	UniSRec	0.0021	0.0014	0.0030	0.0017	0.0014	0.0008	0.0022	0.0010	0.0004	0.0002	0.0008	0.0003
	WhitenRec	0.0007	0.0006	0.0008	0.0006	0.0029	0.0021	0.0034	0.0022	0.0026	0.0019	0.0030	0.0021
	RLMRec	0.0016	0.0013	0.0019	0.0014	0.0376	0.0321	0.0388	0.0325	0.0160	0.0135	0.0172	0.0138
	LLMInit	0.0082	0.0056	0.0113	0.0065	0.0198	0.0179	0.0214	0.0184	0.0075	0.0065	0.0086	0.0068
	LLM-ESR	0.0007	0.0004	0.0010	0.0005	0.0073	0.0061	0.0090	0.0066	0.0045	0.0037	0.0048	0.0037
	AlphaFuse	0.0246	0.0201	0.0279	0.0209	0.0408	0.0348	0.0425	0.0353	0.0165	0.0139	0.0174	0.0142
	PtE	**0.0359**	**0.0317**	**0.0384**	**0.0306**	**0.0517**	**0.0461**	**0.0533**	**0.0469**	**0.0245**	**0.0218**	**0.0224**	**0.0181**
	Best Impr.	+45.93%	+57.71%	+37.63%	+46.41%	+26.72%	+32.47%	+25.41%	+32.86%	+48.48%	+56.83%	+28.74%	+27.46%

Table 4: Performance comparison across different methods on three datasets with long-tail settings.

Dataset	Model	Overall		Tail Item		Head Item		Tail User		Head User	
		R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10
Yelp	SASRec	0.5940	0.3597	0.1142	0.0495	0.7353	0.4511	0.5893	0.3578	0.6122	0.3672
	LLM-ESR	0.6673	0.4208	0.1893	0.0845	0.8080	0.5199	0.6685	0.4229	0.6627	0.4128
	AlphaFuse	0.6631	0.4219	0.1815	0.0775	0.8048	0.5232	0.6617	0.4239	0.6585	0.4141
	PtE	**0.6746**	**0.4263**	**0.2009**	**0.0931**	**0.8173**	**0.5466**	**0.6782**	**0.4296**	**0.6740**	**0.4295**
	Best Impr.	+1.09%	+1.04%	+6.13%	+10.18%	+1.15%	+4.47%	+1.45%	+1.34%	+1.71%	+3.72%
Fashion	SASRec	0.4956	0.4429	0.0454	0.0235	0.6748	0.6099	0.3967	0.3390	0.6239	0.5777
	LLM-ESR	0.5619	0.4743	0.1095	0.0520	0.7420	0.6424	0.4811	0.3769	0.6668	0.6005
	AlphaFuse	0.6008	0.5103	0.2601	0.1646	0.7364	0.6479	0.5352	0.4276	0.6860	0.6175
	PtE	**0.6362**	**0.5429**	**0.3074**	**0.2457**	**0.7682**	**0.6618**	**0.5738**	**0.4621**	**0.7082**	**0.6493**
	Best Impr.	+5.89%	+6.39%	+18.19%	+49.27%	+3.53%	+2.15%	+7.21%	+8.07%	+3.24%	+5.15%
Beauty	SASRec	0.4388	0.3030	0.0870	0.0649	0.5227	0.3598	0.4270	0.2941	0.4926	0.3438
	LLM-ESR	0.5672	0.3713	0.2257	0.1108	0.6486	0.4334	0.5581	0.3643	0.6087	0.4032
	AlphaFuse	0.5793	0.4046	0.1625	0.1006	0.6787	0.4771	0.5692	0.3984	0.6258	0.4326
	PtE	**0.6254**	**0.4395**	**0.2634**	**0.1426**	**0.7052**	**0.5129**	**0.6058**	**0.4165**	**0.6621**	**0.4762**
	Best Impr.	+7.96%	+8.63%	+16.70%	+28.70%	+3.90%	+7.50%	+6.43%	+4.54%	+5.80%	+10.08%

compare against strong baselines, all instantiated with the **SASRec** backbone. Results are summarized in Table 4. Across all datasets and long-tail regimes—including tail items, tail users, and their combinations—**PtE** consistently achieves superior performance. In particular, the improvements are most pronounced for tail users and tail items, where relative gains reach up to 49% in N@10 and R@10 over the strongest baseline. These settings represent an extreme sparsity regime, where removing a single interaction significantly alters the available supervision and exposes the stability of user and item representations. The leave-one-out protocol highlights a key limitation of existing language-enhanced methods: when semantic signals are directly injected or fused at the instance level, representations become highly sensitive to the presence or absence of individual interactions. This sensitivity is especially detrimental for long-tail users, whose preference modeling relies on only a few observations, leading to unstable or inconsistent representations under minimal perturbations. In contrast, **PtE** remains robust under LOO evaluation due to its profile-first design. By prompting LLMs to perform bidirectional reasoning over limited interaction histories, PtE constructs compact semantic profiles that encode high-level preference hypotheses rather than instance-specific signals. These profiles serve as a stable semantic prior that is subsequently embedded under recommendation-

Table 5: Ablation results on the Amazon Movies dataset (Cold-start setting).

Model Variant	Movies				Toys				Sports			
	N@10	**M@10**	**N@20**	**M@20**	**N@10**	**M@10**	**N@20**	**M@20**	**N@10**	**M@10**	**N@20**	**M@20**
PtE$_{Full}$	**0.0523**	**0.0436**	**0.0689**	**0.0472**	**0.0473**	**0.0398**	**0.0511**	**0.0435**	**0.0179**	**0.0145**	**0.0236**	**0.0137**
No Profile (Direct Embedding)	0.0451	0.0368	0.0592	0.0391	0.0398	0.0327	0.0426	0.0358	0.0142	0.0116	0.0189	0.0109
User-only Profiling	0.0489	0.0401	0.0645	0.0438	0.0435	0.0365	0.0472	0.0401	0.0163	0.0132	0.0211	0.0124
Item-only Profiling	0.0495	0.0407	0.0653	0.0445	0.0442	0.0371	0.0480	0.0408	0.0167	0.0135	0.0217	0.0127
Single-pass Profiling	0.0501	0.0415	0.0667	0.0453	0.0456	0.0382	0.0493	0.0419	0.0172	0.0140	0.0225	0.0132
No GRPO	0.0490	0.0403	0.0651	0.0440	0.0440	0.0369	0.0476	0.0404	0.0165	0.0133	0.0214	0.0126
GRPO w/o Distinctiveness	0.0508	0.0421	0.0673	0.0460	0.0462	0.0387	0.0499	0.0423	0.0174	0.0141	0.0229	0.0134
GRPO w/o Collaborative Alignment	0.0467	0.0382	0.0614	0.0410	0.0411	0.0340	0.0448	0.0376	0.0151	0.0122	0.0198	0.0116

Table 6: Ablation results on the Yelp dataset (Long-Tail setting).

Model Variant	Overall		Tail Item		Head Item		Tail User		Head User	
	R@10	**N@10**	**R@10**	**N@10**	**R@10**	**N@10**	**R@10**	**N@10**	**R@10**	**N@10**
PtE$_{Full}$	**0.6746**	**0.4263**	**0.2009**	**0.0931**	**0.8173**	**0.5466**	**0.6782**	**0.4296**	**0.6740**	**0.4295**
No Profile (Direct Embedding)	0.5982	0.3715	0.1523	0.0687	0.7436	0.4892	0.6015	0.3748	0.5978	0.3751
User-only Profiling	0.6354	0.3987	0.1768	0.0812	0.7825	0.5143	0.6389	0.4021	0.6350	0.4024
Item-only Profiling	0.6487	0.4072	0.1851	0.0856	0.7962	0.5278	0.6521	0.4105	0.6483	0.4108
Single-pass Profiling	0.6623	0.4185	0.1934	0.0892	0.8087	0.5385	0.6657	0.4228	0.6619	0.4231
No GRPO	0.6512	0.4106	0.1840	0.0851	0.7998	0.5289	0.6547	0.4139	0.6509	0.4142
GRPO w/o Distinctiveness	0.6669	0.4209	0.1912	0.0884	0.8126	0.5417	0.6703	0.4246	0.6662	0.4248
GRPO w/o Collaborative Alignment	0.6118	0.3826	0.1627	0.0734	0.7604	0.5012	0.6152	0.3859	0.6115	0.3861

driven supervision, reducing representation drift when individual interactions are removed. As a result, PtE achieves stronger generalization across tail users, tail items, and their combinations. Overall, the leave-one-out results demonstrate that effective long-tail recommendation requires not only richer semantic information, but also a representation strategy that preserves preference consistency under extremely sparse supervision. The profile-to-embedding paradigm adopted by **PtE** provides such stability, complementing its strong performance in cold-start settings.

5.4 Ablation Study

We conduct ablation studies on *Amazon Movies* under cold-start settings and on *Yelp* under long-tail distributions, both using the **SASRec** backbone. The evaluated variants are summarized in Table 6. In cold-start scenarios, removing profile generation (**No Profile**) leads to the largest performance drop, confirming that directly embedding dense LLM representations is insufficient under extreme sparsity. Partial profiling (user-only or item-only) improves over direct embedding but consistently underperforms the full model, highlighting the importance of jointly modeling user and item semantics. Single-pass profiling further degrades performance, indicating that iterative refinement is necessary to stabilize semantic abstraction. Moreover, ablating GRPO (Shao et al., 2024) reveals that collaborative alignment is critical for cold-start generalization, with particularly pronounced effects on highly sparse domains such as *Toys* and *Sports*. Under long-tail distributions, **PtE**$_{Full}$ consistently achieves the best performance across overall, tail-item, and tail-user splits. Direct embedding suffers severe degradation on tail entities, while partial profiling yields only limited gains. Removing collaborative alignment in GRPO causes the largest drops on tail items, whereas removing distinctiveness mainly affects overall ranking quality. Together, these results indicate that PtE's effectiveness in long-tail settings arises from combining profile-level semantic abstraction with collaborative-aware embedding optimization.

6 Conclusion

We analyze the gap between the theoretical expressiveness and empirical effectiveness of LLM representations in recommendation. Although LLM embeddings subsume collaborative signals from ID representations, their practical utility is limited by semantic entanglement and objective misalignment. We therefore propose Profile-then-Embedding (PtE), which decouples semantic reasoning from representation learning via explicit semantic profiles and downstream embedding alignment. Experiments demonstrate that PtE consistently outperforms fusion-based methods, highlighting the importance of disciplined semantic utilization over direct representation fusion.

Limitations

Our design prioritizes representational clarity and semantic controllability, which leads the profile generation stage to adopt an iterative formulation. While this choice favors stable and interpretable semantic profiles, it naturally introduces additional computational overhead. Exploring more compact distillation schemes and the use of lightweight, domain-adapted LLMs offers a natural extension to further improve efficiency without altering the core framework. In addition, although our analysis provides principled insights into semantic disentanglement, a more formal characterization of the disentanglement dynamics and their theoretical connection to downstream performance remains an interesting avenue for future investigation.

References

Guillaume Alain and Yoshua Bengio. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.

David Blackwell. 1953. Equivalent comparisons of experiments. *The annals of mathematical statistics*, pages 265–272.

George Casella and Roger Berger. 2024. *Statistical inference*. Chapman and Hall/CRC.

Gaode Chen, Ruina Sun, Yuezihan Jiang, Jiangxia Cao, Qi Zhang, Jingjian Lin, Han Li, Kun Gai, and Xinghua Zhang. 2024. A multi-modal modeling framework for cold-start short-video recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems*, pages 391–400.

Lei Chen, Chen Gao, Xiaoyi Du, Hengliang Luo, Depeng Jin, Yong Li, and Meng Wang. 2025. Enhancing id-based recommendation with large language models. *ACM Transactions on Information Systems*, 43(5):1–30.

Jesse Harte, Wouter Zorgdrager, Panos Louridas, Asterios Katsifodimos, Dietmar Jannach, and Marios Fragkoulis. 2023. Leveraging large language models for sequential recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1096–1102.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*.

Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. 2022. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 585–593.

Guoqing Hu, Zhengyi Yang, Zhibo Cai, An Zhang, and Xiang Wang. 2024a. Generate and instantiate what you prefer: Text-guided diffusion for sequential recommendation. *arXiv preprint arXiv:2410.13428*.

Guoqing Hu, An Zhang, Shuo Liu, Zhibo Cai, Xun Yang, and Xiang Wang. 2025. Alphafuse: Learn id embeddings for sequential recommendation in null space of language embeddings. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1614–1623.

Jun Hu, Wenwen Xia, Xiaolu Zhang, Chilin Fu, Weichang Wu, Zhaoxin Huan, Ang Li, Zuoli Tang, and Jun Zhou. 2024b. Enhancing sequential recommendation via llm-based semantic embedding learning. In *Companion Proceedings of the ACM Web Conference 2024*, pages 103–111.

Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE.

Kenneth Li, Aspen K Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2022. Emergent world representations: Exploring a sequence model trained on a synthetic task. *arXiv preprint arXiv:2210.13382*.

Jianghao Lin, Xinyi Dai, Yunjia Xi, Weiwen Liu, Bo Chen, Hao Zhang, Yong Liu, Chuhan Wu, Xiangyang Li, Chenxu Zhu, and 1 others. 2025. How can recommender systems benefit from large language models: A survey. *ACM Transactions on Information Systems*, 43(2):1–47.

Qidong Liu, Xian Wu, Wanyu Wang, Yejing Wang, Yuanshao Zhu, Xiangyu Zhao, Feng Tian, and Yefeng Zheng. 2025. Llmemb: Large language model can be a good embedding generator for sequential recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12183–12191.

Qidong Liu, Xian Wu, Xiangyu Zhao, Yejing Wang, Zijian Zhang, Feng Tian, and Yefeng Zheng. 2024. Large language models enhanced sequential recommendation for long-tail user and item. *arXiv e-prints*, pages arXiv–2405.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 43–52.

Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. 2022. Linearly mapping from image to text space. *arXiv preprint arXiv:2209.15162*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Roma Patel and Ellie Pavlick. 2022. Mapping language models to grounded conceptual spaces. In *International conference on learning representations*.

Zekai Qu, Ruobing Xie, Chaojun Xiao, Zhanhui Kang, and Xingwu Sun. 2024. The elephant in the room: rethinking the usage of pre-trained language model in sequential recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems*, pages 53–62.

Xubin Ren, Wei Wei, Lianghao Xia, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. Representation learning with large language models for recommendation. In *Proceedings of the ACM web conference 2024*, pages 3464–3475.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Leheng Sheng, An Zhang, Yi Zhang, Yuxin Chen, Xiang Wang, and Tat-Seng Chua. 2024. Language models encode collaborative signals in recommendation.

Hongyi Shui, Yuanjing Zhu, Fan Zhuo, Yibo Sun, and Daoyuan Li. 2024. An emotion text classification model based on llama3-8b using lora technique. In *2024 7th International Conference on Computer Information Science and Application Technology (CISAT)*, pages 380–383. IEEE.

Anima Singh, Trung Vu, Nikhil Mehta, Raghunandan Keshavan, Maheswaran Sathiamoorthy, Yilin Zheng, Lichan Hong, Lukasz Heldt, Li Wei, Devansh Tandon, and 1 others. 2024. Better generalization with semantic ids: A case study in ranking for recommendations. In *Proceedings of the 18th ACM Conference on Recommender Systems*, pages 1039–1044.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Jiakai Tang, Sunhao Dai, Teng Shi, Jun Xu, Xu Chen, Wen Chen, Jian Wu, and Yuning Jiang. 2025. Think before recommend: Unleashing the latent reasoning power for sequential recommendation. *arXiv preprint arXiv:2503.22675*.

Ivan Vulić, Edoardo Maria Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. *arXiv preprint arXiv:2010.05731*.

Yaqing Wang, Hongming Piao, Daxiang Dong, Quanming Yao, and Jingbo Zhou. 2024. Warming up cold-start ctr prediction by learning item-specific feature interactions. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3233–3244.

Yu Wang, Lei Sang, Yi Zhang, and Yiwen Zhang. 2025a. Intent representation learning with large language model for recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1870–1879.

Yuhao Wang, Junwei Pan, Xinhang Li, Maolin Wang, Yuan Wang, Yue Liu, Dapeng Liu, Jie Jiang, and Xiangyu Zhao. 2025b. Empowering large language model for sequential recommendation via multimodal embeddings and semantic ids. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pages 3209–3219.

Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 726–735.

Song-Li Wu, Liang Du, Jia-Qi Yang, Yu-Ai Wang, De-Chuan Zhan, Shuang Zhao, and Zi-Xun Sun. 2023. Re-sort: Removing spurious correlation in multi-level interaction for ctr prediction. *arXiv preprint arXiv:2309.14891*.

Zhengyi Yang, Jiancan Wu, Zhicai Wang, Xiang Wang, Yancheng Yuan, and Xiangnan He. 2023. Generate what you prefer: Reshaping sequential recommendation via guided diffusion. *Advances in Neural Information Processing Systems*, 36:24247–24261.

Zheng Yuan, Fajie Yuan, Yu Song, Youhua Li, Junchen Fu, Fei Yang, Yunzhu Pan, and Yongxin Ni. 2023. Where to go next for recommender systems? id-vs. modality-based recommender models revisited. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2639–2649.

Jirong Zha, Yuxuan Fan, Xiao Yang, Chen Gao, and Xinlei Chen. 2025. How to enable llm with 3d capacity? a survey of spatial reasoning in llm. *arXiv preprint arXiv:2504.05786*.

Junjie Zhang, Ruobing Xie, Yupeng Hou, Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2025. Recommendation

as instruction following: A large language model empowered recommendation approach. *ACM Transactions on Information Systems*, 43(5):1–37.

Lingzi Zhang, Xin Zhou, Zhiwei Zeng, and Zhiqi Shen. 2024. Are id embeddings necessary? whitening pre-trained text embeddings for effective sequential recommendation. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 530–543. IEEE.

Yongchun Zhu, Ruobing Xie, Fuzhen Zhuang, Kaikai Ge, Ying Sun, Xu Zhang, Leyu Lin, and Juan Cao. 2021. Learning to warm up cold item embeddings for cold-start recommendation with meta scaling and shifting networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1167–1176.

A Detailed Experimental Setup

A.1 Baseline Methods

We compare PtE with a broad range of language-guided ID embedding learning strategies, including semantic reconstruction, semantic initialization, adaptive projection, and other representative baselines. Specifically, RLMRec (Ren et al., 2024), originally designed for collaborative filtering, is adapted to sequential recommendation with two variants: *RLMRec-Gen*, which maps language embeddings into behavioral space via a two-layer MLP using similarity deviations from ID embeddings as a regularization term, and *RLMRec-Con*, which projects ID embeddings into semantic space through a two-layer MLP and aligns them against language embeddings. LLMInit (Harte et al., 2023; Hu et al., 2024b; Qu et al., 2024) initializes ID embeddings directly with language embeddings, where we further apply PCA for dimensionality reduction to better align with behavioral space. MoRec (Yuan et al., 2023) employs a pre-trained modal encoder followed by a dense layer for dimension transformation, while UniSRec (Hou et al., 2022) leverages a mixture-of-experts enhanced adaptor to process language embeddings. WhitenRec (Zhang et al., 2024; Su et al., 2021) applies whitening to decorrelate highly similar language embeddings, followed by an MLP adaptor to produce final item embeddings. LLM-ESR (Liu et al., 2024) proposes a dual-view modeling framework that integrates semantic initialization and adaptive projection to address long-tail issues in users and items. Finally, iDreamRec (Hu et al., 2024a) introduces diffusion models to learn consistent language embeddings for recommendation, with a variance-preserving linear transformation. AlphaFuse (Hu et al., 2025) fuses standardized LLM embeddings and learned ID embeddings by SVD-based space decomposition, achieving seamless integration without auxiliary modules and broad adaptability across discriminative and generative recommendation frameworks.

A.1.1 Implementation Detail.

For our experiments, we follow the dataset splits and code from LLM-ESR (Liu et al., 2024), replicating their leave-one-out results and adjusting only the learning rate for AlphaFuse to $\{0.001, 0.0001, 0.00001\}$. In the cold-start user setting, item textual attributes and descriptions are concatenated and encoded using OpenAI's `text-embedding-3` to obtain language embeddings. For SASRec, the final item embeddings are fixed to 128 dimensions and ID embeddings to 64 for both AlphaFuse and LLM-ESR, with InfoNCE loss (Oord et al., 2018) applied using 64 negative samples; additional loss coefficients for baselines are set to 0.1 or 1 following the original works (Liu et al., 2024; Ren et al., 2024). For DreamRec, the original dimension of language embeddings is retained without clipping, making RLMRec-Con and RLMRec-Gen similar, while LLM-ESR allocates equal dimensions to semantic and collaborative views and employs MSE loss (Ho et al., 2020) without negative samples; additional loss coefficients are fixed as in the original papers. For AlphaFuse, the null space is selected with thresholds $\{0.001, 0.01, 0.1, 0.25, 0.5\}$, replacing original values with $EID \in \mathbb{R}^{N \times d_n}$ initialized from a standard Gaussian distribution. All models are implemented in Python 3.7 with PyTorch 1.12.1 on an Nvidia GeForce RTX 3090, trained with a batch size of 256 and learning rates $\{0.01, 0.001, 0.0001, 0.00001\}$ using the Adam optimizer, with early stopping based on validation performance and all random seeds fixed to 22 for reproducibility. Evaluation follows the all-rank protocol (He et al., 2020; Ren et al., 2024; Wu et al., 2021), measuring performance across all items with Normalized Discounted Cumulative Gain (N@K) and Mean Reciprocal Rank (M@K).

B Theoretical Analysis

The analysis in Section 3 establishes an information-theoretic upper bound under exact sufficiency and Bayes-optimal decision rules. We now extend this framework to settings that more faithfully reflect modern representation learning, where representations are *not* sufficient and predictors are constrained to finite hypothesis classes.

B.1 Proof of Theorem 1

We consider two latent variables $X_{\text{lm}}, \quad X_{\text{id}}$, representing distinct semantic sources captured by LLM embeddings and ID-based embeddings, respectively. The downstream target Y depends on both:

$$Y \sim P(Y \mid X_{\text{lm}}, X_{\text{id}}). \tag{14}$$

Representations are generated as

$$Z_{\text{lm}} = \phi(X_{\text{lm}}), \qquad Z_{\text{id}} = \psi(X_{\text{id}}), \tag{15}$$

and the fusion representation is

$$Z_{\text{fus}} = g(Z_{\text{lm}}, Z_{\text{id}}). \tag{16}$$

No sufficiency assumption is imposed between Z_{lm} and Z_{id}.

B.1.1 Prediction Under a Restricted Hypothesis Class

Let $\mathcal{H}$ denote a restricted hypothesis class (e.g., linear predictors or shallow networks). Define the $\mathcal{H}$-restricted Bayes risk:

$$\mathcal{R}_{\mathcal{H}}(Z \to Y) = \inf_{h \in \mathcal{H}} \mathbb{E}[\ell(h(Z), Y)]. \tag{17}$$

This quantity captures both information content and *accessibility* of that information under architectural constraints.

B.1.2 Approximate Sufficiency

Definition 1 (ϵ-Approximate Sufficiency)**.** *A representation U is said to be ϵ-sufficient for Y relative to a hypothesis class $\mathcal{H}$ if*

$$\mathcal{R}_{\mathcal{H}}(U \to Y) \leq \mathcal{R}^*(Y) + \epsilon, \tag{18}$$

where $\mathcal{R}^*(Y)$ denotes the Bayes-optimal risk.

B.1.3 Main Result: Fusion Can Strictly Improve Approximate Sufficiency

Theorem 2 (Fusion Improves Approximate Sufficiency)**.** *There exist distributions $P(X_{\text{lm}}, X_{\text{id}}, Y)$ and hypothesis classes $\mathcal{H}$ such that:*

$$\mathcal{R}_{\mathcal{H}}(Z_{\text{fus}} \to Y) < \mathcal{R}_{\mathcal{H}}(Z_{\text{lm}} \to Y), \tag{19}$$

$$\mathcal{R}_{\mathcal{H}}(Z_{\text{fus}} \to Y) < \mathcal{R}_{\mathcal{H}}(Z_{\text{id}} \to Y), \tag{20}$$

even though

$$\mathcal{R}^*(Z_{\text{fus}} \to Y) = \mathcal{R}^*(Z_{\text{lm}}, Z_{\text{id}} \to Y). \tag{21}$$

Proof Sketch. Consider a binary classification task where Y depends on a nonlinear interaction between X_{lm} and X_{id}. Assume $\mathcal{H}$ consists of linear predictors.

Individually, neither Z_{lm} nor Z_{id} admits a linear decision boundary with low error. However, a fusion representation Z_{fus} that explicitly encodes interaction features (e.g., concatenation followed by a fixed nonlinear map) renders the task linearly separable.

Thus, while no representation dominates in the Blackwell sense, Z_{fus} is strictly more ϵ-sufficient under $\mathcal{H}$. □

B.1.4 Interpretation

Theorem 2 formalizes a key practical insight: fusion gains arise precisely when individual representations are *not sufficient* and when predictors are capacity-limited. In contrast, the Blackwell-based upper bound characterizes an idealized asymptotic regime and should be interpreted as a theoretical ceiling rather than a prescription against fusion.

B.2 Convergence Analysis

Theorem 3 (Stability of Bidirectional Profile Refinement)**.** *Assume that the user- and item-conditioned generation operators $\mathcal{M}_u$ and $\mathcal{M}_v$ are ρ-Lipschitz continuous with respect to a semantic distance metric $d(\cdot,\cdot)$ on the profile space $\mathcal{T}$, where $0 < \rho < 1$. Then the bidirectional closed-loop update in Eq. (9) produces a sequence of user and item profiles $\{\mathbf{p}_u^{(t)}, \mathbf{p}_v^{(t)}\}$ that converges to a stable representation. In particular, the semantic discrepancy between successive iterations decays geometrically, and the profiles reach an approximate fixed point within a finite number of steps.*

Proof. We analyze the evolution of semantic profiles under the bidirectional update. At iteration t, the item profile is generated from the current user profiles as

$$\mathbf{p}_v^{(t)} = \mathcal{M}_v\big(\{\mathbf{p}_u^{(t)} \mid u \in \mathcal{U}(v)\}\big),$$

and the user profile is subsequently updated via

$$\mathbf{p}_u^{(t+1)} = \mathcal{M}_u\big(\mathcal{V}_u,\ \{\mathbf{p}_v^{(t)} \mid v \in \mathcal{V}_u\}\big).$$

By the ρ-Lipschitz continuity of $\mathcal{M}_v$, for any two successive user profile states we have

$$d\Big(\mathbf{p}_v^{(t)}, \mathbf{p}_v^{(t-1)}\Big) \le \rho \cdot d\Big(\mathbf{p}_u^{(t)}, \mathbf{p}_u^{(t-1)}\Big).$$

Similarly, applying the Lipschitz continuity of $\mathcal{M}_u$ yields

$$d\Big(\mathbf{p}_u^{(t+1)}, \mathbf{p}_u^{(t)}\Big) \le \rho \cdot d\Big(\mathbf{p}_v^{(t)}, \mathbf{p}_v^{(t-1)}\Big).$$

Combining the two inequalities gives

$$d\Big(\mathbf{p}_u^{(t+1)}, \mathbf{p}_u^{(t)}\Big) \le \rho^2 \cdot d\Big(\mathbf{p}_u^{(t)}, \mathbf{p}_u^{(t-1)}\Big).$$

Since $0 < \rho < 1$, the discrepancy between successive user profiles decreases geometrically. An analogous argument holds for item profiles. Therefore, the bidirectional update constitutes a contraction mapping on the joint semantic profile space, and the iterative process converges to a stable representation. □