

# Do Reviews Help with Cold-Start in Sequential Recommender Systems?\*

Min-Gyu Jang and Sang-Min Choi<sup>†</sup>

Gyeongsang National University, Jinju-si, Republic of Korea  
{alsrb9936, jerassi}@gnu.ac.kr

## Abstract

Sequential recommender systems are effective in modeling user behavior but struggle in cold-start scenarios due to sparse interaction data. While user reviews provide rich contextual signals, their practical utility under extreme sparsity remains questionable. We present a direct comparison between reviews and structured metadata as side information for cold-start sequential recommendation. By embedding both into a unified semantic space with a pre-trained language model, we evaluate their impact using SASRec and GRU4Rec across multiple Amazon domains. Experimental results show that metadata yields more consistent and stable improvements than reviews, which suffer from sparsity and noise. Embedding analysis further confirms that metadata forms more coherent clusters, highlighting its reliability as an auxiliary signal in cold-start environments.

## 1 Introduction

The explosive growth of digital platforms has provided users with vast amounts of information and content, concurrently giving rise to the problem of information overload. In this environment, sequential recommendation systems, which predict the next action of a user by understanding the sequence and temporal context of their behavior, have become a core technology for enhancing personalized experiences [1]. In this context, the advent of the big data era has further positioned recommendation systems as a core technology, particularly in the field of e-commerce [2, 3].

However, sequential recommendation systems have a significant weakness, which is their heavy reliance on user historical interaction sequences [4]. For new users or new items with insufficient historical interaction, it is impossible to construct a reliable sequence, leading to the “cold-start” problem where recommendations cannot be effectively generated [5]. Furthermore, recent developments such as Apple’s App Tracking Transparency (ATT) policy [6], privacy regulations like GDPR [7], and an increase in users declining cookie collection have limited the data acquisition capabilities of companies [8]. This overall decrease in data availability accelerates the performance degradation of sequential recommendation models and further exacerbates the cold-start problem.

To mitigate this data sparsity issue, previous research has focused on utilizing side information beyond user-item interactions [9, 10, 11]. This includes metadata such as an item’s genre or category, user demographics, and review texts written by users. User reviews, in particular, are known to be a valuable source of information as they contain specific attributes of items and the qualitative reasons for user preferences, making them useful for tracking changes in user tastes and enhancing the explainability of model [11, 12].

---

\*Proceedings of The 2025 IFIP WG 8.4 International Symposium on E-Business Information Systems Evolution (EBISION 2025), Article No. 19, December 16-18, 2025, Sapporo, Japan. © The copyright of this paper remains with the author(s).

<sup>†</sup>Corresponding author

Nevertheless, a fundamental question arises about the practical effectiveness of review data in cold-start scenarios. In particular, the dilemma is that, since reviews are generated from interactions, they are inherently sparse when interaction sequences are missing [13]. Moreover, combining review data from multiple services can pose re-identification risks, placing limitations on further data collection [14, 15]. Prior work has explored both metadata and reviews, but direct comparisons between the two remain lacking in cold-start sequential recommendation. To address this gap, a systematic evaluation is required to clarify their relative effectiveness.

Therefore, this study aims to identify which type of side information, reviews or metadata more effectively contributes to the performance enhancement of sequential recommendation models in cold-start scenarios. To achieve this, we embed these heterogeneous types of information into a unified semantic space using a pre-trained language model and conduct comparative experiments under identical conditions. Our results confirm that in cold-start settings, review data do not always yield superior performance over metadata. We further investigate the underlying causes of this phenomenon through embedding clustering and correlation analysis. The main contributions of this study are as follows:

- **Leakage-free evaluation** : We standardize a leakage-free, time-cutoff evaluation for item cold-start and use it to compare review-based and metadata-based item representations under identical model conditions.
- **Controlled head-to-head analysis** : A direct comparison between review and metadata item representations under identical conditions, clarifying when each source is beneficial
- **Empirical Analysis of Review and Metadata Embeddings** : By embedding both data types into the same semantic space, we conduct visual and quantitative analysis based on clustering and validate the correlation with model performance. This sheds new light on the limitations of review data and the strengths of metadata.

## 2 Related Work

### 2.1 Sequential Recommendation

Sequential recommendation, which predicts a user’s next action by leveraging the order of their recent interactions, originated from models based on Markov Chains and causal transitions like FPMC(Factorizing Personalized Markov Chains) [16], and has since evolved to neural network-based architectures. While RNN(Recurrent Neural Network)-based models —GRU4Rec(Gated Recurrent Unit for Recommendation) [17] — effectively captured sequential dependencies, they faced limitations regarding long-term dependencies and parallelization.

Subsequently, models employing self-attention and Transformers, such as SASRec(Self-Attentive Sequential Recommendation) [18] and BERT4Rec(Bidirectional Encoder Representations from Transformer for Recommendation) [19] have been introduced. By utilizing positional encoding and masked/reconstruction learning objectives, these models capture both short-term and long-term patterns simultaneously and offer parallel processing capabilities suitable for large-scale datasets. More recent studies have further improved predictive performance and generalization by incorporating factors such as time information [20], context [21], contrastive learning [20], and self-supervised learning [22].

Nevertheless, many sequence recommendation models implicitly assume a minimum level of interaction history, and under extreme sparsity the optimization can become unstable and the risk of overfitting may increase [23, 5].

## 2.2 The Cold-Start Problem

The cold-start problem refers to scenarios where a model must make predictions for items or users not seen during training, or where the training dataset itself is insufficient [5].

In sequential recommendation systems, the cold-start problem manifests at both the user and item levels. The core challenge is the absence of the sequence itself [24]. To mitigate this, various approaches have been proposed, such as utilizing content/attribute-based features, augmenting representations through graph propagation techniques, and employing cross-domain transfer methods. But many studies have relied on review-rich public benchmarks such as Amazon, and as a result there is a recurring concern that extreme sparsity settings have been comparatively under-evaluated [25].

## 2.3 Utilization of Side Information

Textual side information can be broadly categorized into structured metadata (e.g., genre, category, brand, attribute tags) and unstructured text data (e.g., reviews, product descriptions). Metadata is relatively easy to collect and normalize. Review data, on the other hand, offers the advantage of containing detailed attributes, context, and the user’s qualitative evaluation, which can enhance explainability and enable the learning of finer-grained user preferences [11, 12].

However, review data is dependent on interaction data and thus can be sparse and noisy. Many review-based recommendation systems have evolved towards constructing user-item embeddings via sentence encoders (e.g., CNN/LSTM, Transformer) [13], weighting the importance of reviews [12].

Metadata-based methods augment sequence representations by directly combining or gating attribute embeddings [26, 27], or by propagating relationships between attributes in a knowledge graph [28]. However, in the ultra-high sparsity of cold-start environments, several factors accumulate: (i) an absolute lack of reviews, (ii) merging reviews across services may be subject to privacy constraints, and (iii) embedding distortion due to a high volume of noise. Consequently, it cannot be definitively concluded that reviews are always superior to metadata in cold-start environments.

# 3 Method

## 3.1 Problem Definition and Notation

Let  $\mathcal{U} = \{u_1, \dots, u_n\}$  be the set of  $n$  users and  $\mathcal{I} = \{i_1, \dots, i_m\}$  be the set of  $m$  items. Each item  $i \in \mathcal{I}$  is associated with a title  $t(i)$  and a set of categories  $C(i) = \{c_1, c_2, \dots, c_l\}$  where  $c_l$  denotes categories of item assigned to item  $i$ . For every user  $u \in \mathcal{U}$ , the interaction history is denoted as a sequence:

$$S_u = \langle (i_1, r_1^u), (i_2, r_2^u), \dots, (i_{T_u}, r_{T_u}^u) \rangle,$$

where  $T_u = |S_u|$  is the sequence length with which user  $u$  interacted, and  $r_{T_u}^u \in \mathcal{R}$  is the review provided by the user for that interaction. And each interaction is defined as a pair consisting of the item and the corresponding review. The review can be described formally by a function:

$$r : \mathcal{U} \times \mathbb{N} \rightarrow \mathcal{R}, \quad r(u, T_u) = r_{T_u}^u.$$

Given the user sequence  $S_u$ , our goal is to predict the target item  $y = i_{T_u}$  using  $S_{u < T_u - 1} = \langle (i_1, r_1^u), (i_2, r_2^u), \dots, (i_{T_u - 1}, r_{T_u - 1}^u) \rangle$  such that:

$$\hat{y} = f(U, I, S_{u < T_u - 1}),$$

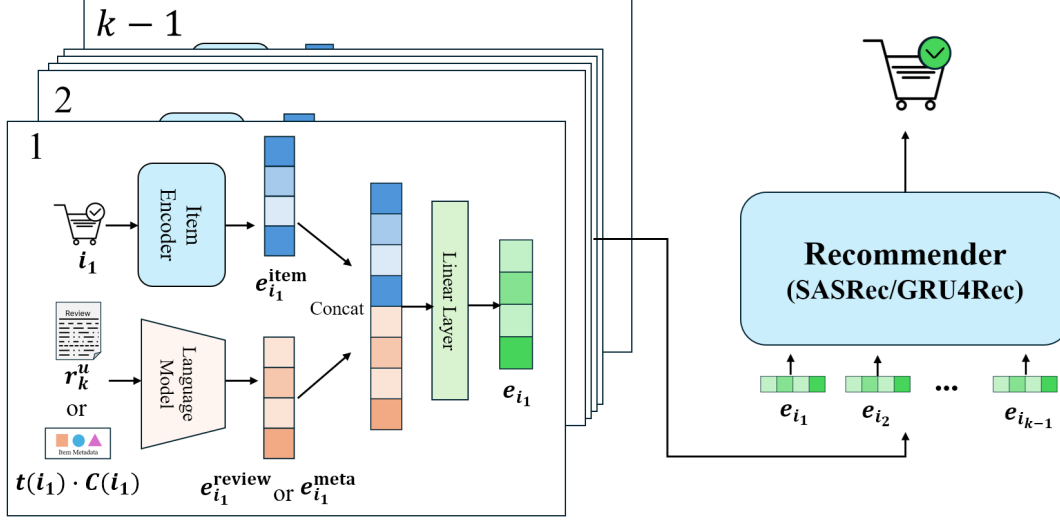


Figure 1: Model embedding approach

where  $f$  is a sequence recommendation function that maps  $(U, I, S_{u < T_u - 1})$  to a probability over candidate items, from which  $\hat{y}$  is selected.

In this study, we restrict our experiments to the *item cold-start* scenario. In such an environment, the last item  $y$  is excluded from the training phase (i.e., it is unseen by the model) and is only observed during the test phase. Formally, the dataset can be expressed as  $S_{u < T_u - 1} = \langle (i_1, r_1^u), (i_2, r_2^u), \dots, (i_{T_u - 1}, r_{T_u - 1}^u) \rangle$ ,  $y = i_{T_u}$  where  $y$  is withheld from training and used solely for evaluation.

### 3.2 Model approaches

The overall model approach is illustrated in Figure 1. We adopt two representative sequential recommendation models, SASRec and GRU4Rec, as our base architectures. Both models rely on item embeddings to capture sequential user preferences. To enrich the representation of items, we additionally incorporate embeddings derived from either item metadata or user reviews. Let  $\text{PLM}(\cdot)$  denote a pretrained language model used to generate dense representations. For each item  $i$ , we define three types of embeddings:

$$\mathbf{e}_i^{\text{item}} \in \mathbb{R}^{d_{\text{item}}},$$

$$\mathbf{e}_i^{\text{meta}} = \text{PLM}(t(i) \cdot C(i)) \in \mathbb{R}^{d_{\text{meta}}}, \quad \mathbf{e}_i^{\text{review}} = \text{PLM}(r_{T_u}^u) \in \mathbb{R}^{d_{\text{review}}},$$

where  $\cdot$  denotes concatenation with whitespace separation,  $\mathbf{e}_i^{\text{item}}$  is the standard item embedding,  $\mathbf{e}_i^{\text{meta}}$  is obtained from  $t(i)$  and  $C(i)$ , and  $\mathbf{e}_i^{\text{review}}$  is obtained from  $r_{T_u}^u$ .

We design two experimental settings:

- **Meta-based fusion:** the item embedding  $\mathbf{e}_i^{\text{item}}$  is combined with  $\mathbf{e}_i^{\text{meta}}$ .
- **Review-based fusion:** the item embedding  $\mathbf{e}_i^{\text{item}}$  is combined with  $\mathbf{e}_i^{\text{review}}$ .

In both cases, the embeddings are concatenated and projected into a unified representation through a learnable linear layer:

$$\mathbf{e}_i = W \cdot [\mathbf{e}_i^{\text{item}} \parallel \mathbf{e}_i^x] + b, \quad x \in \{\text{meta}, \text{review}\},$$

where  $\parallel$  denotes concatenation,  $W$  is a trainable weight matrix, and  $b$  is a bias term. The resulting representation  $\mathbf{e}_i$  is then used as input to SASRec and GRU4Rec for sequential modeling.

### 3.3 Unsupervised Cluster Analysis of Embedding Spaces

We apply  $k$ -means clustering algorithm to the embedding set  $\mathcal{X} = \{x_1, \dots, x_n\}$  [29]. The  $k$ -means assigns each data point  $x_n$  to the nearest cluster centroid  $\mu_a$ , minimizing the following objective function  $J$ :

$$J = \sum_{a=1}^k \sum_{x \in \mathcal{G}_a} \|x - \mu_a\|^2, \quad (1)$$

where  $\mathcal{G}_a$  denotes the  $a$ -th cluster. The optimal number of clusters  $k$  was determined using the Davies–Bouldin Index (DBI) [30], Silhouette Coefficient (SC) [31], and Calinski–Harabasz Index (CH) [32].

**Davies–Bouldin Index (DBI):** The DBI measures clustering quality using intra-cluster compactness and inter-cluster separation. We write  $\delta(x, y)$  for the distance between  $x$  and  $y$ . Defined the average intra-cluster distance:

$$\bar{d}_a = \frac{1}{|\mathcal{G}_a|} \sum_{x \in \mathcal{G}_a} \delta(x, \mu_a), \quad D_{ab} = \delta(\mu_a, \mu_b).$$

Then the DBI is given by

$$\text{DBI} = \frac{1}{k} \sum_{i=1}^k \max_{a \neq b} \left( \frac{\bar{d}_a + \bar{d}_b}{D_{ab}} \right), \quad (2)$$

where  $\bar{d}_a$  is the average distance of samples in cluster  $a$  to its centroid, and  $D_{ab}$  is the distance between the centroids of clusters  $a$  and  $b$ . A lower DBI indicates more compact and well-separated clusters.

**Silhouette Coefficient:** Define the mean intra-cluster distance and the nearest other-cluster mean distance as:

$$\bar{d}_{\text{in}}(n) = \frac{1}{|\mathcal{G}_a| - 1} \sum_{\substack{q \in \mathcal{G}_a \\ q \neq n}} \delta(x_n, x_q), \quad \bar{d}_{\text{out}}(n) = \min_{b \neq a} \frac{1}{|\mathcal{G}_b|} \sum_{q \in \mathcal{G}_b} \delta(x_n, x_q)$$

The per-sample silhouette is

$$s(n) = \frac{\bar{d}_{\text{out}}(n) - \bar{d}_{\text{in}}(n)}{\max\{\bar{d}_{\text{in}}(n), \bar{d}_{\text{out}}(n)\}},$$

and the final Silhouette Score is

$$\text{Silhouette Score} = \frac{1}{N} \sum_{n=1}^N s(n), \quad (3)$$

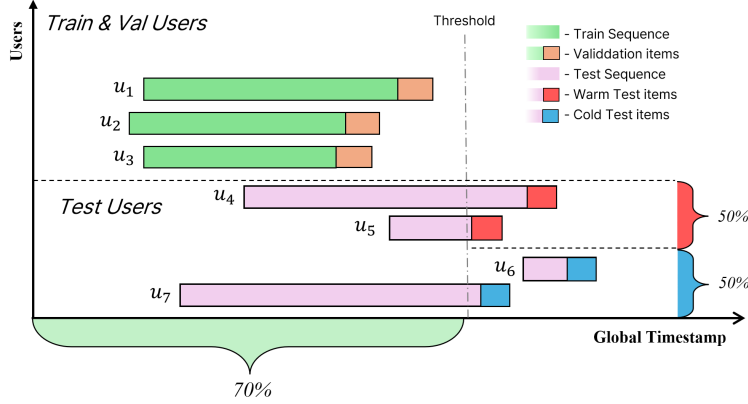


Figure 2: Data splitting strategy

where values closer to 1 indicate better clustering.

**Calinski-Harabasz Index (CH):** The CH index is given as follows based on the dispersion ratio:

$$CH = \frac{B/(k-1)}{W/(n-k)} \quad (4)$$

where  $B$  is the between-cluster dispersion,  $W$  is the within-cluster dispersion,  $n$  is the total number of samples, and  $k$  is the number of clusters. A higher CH score indicates clusters that are well separated and compact.

## 4 Experimental Settings

### 4.1 Datasets & Baselines

Table 1: Amazon dataset statistics

Dataset	Users	Items	Interactions	Sparsity	Item category
Beauty	22363	12101	198502	0.999266%	259
Sports	35598	18357	296337	0.999547%	1470
Digital Music	5541	3568	64706	0.996727%	338
Musical Instruments	1429	900	10261	0.992022%	206

We conducted experiments on four sub-domains of the *Amazon Review 2014* dataset [33], namely **Beauty**, **Digital Music**, **Musical Instruments**, and **Sports & Outdoors**. The dataset information can be found in Table 1. The splitting scheme of the dataset, where cold items are excluded from the training set and appear only in the evaluation set. User-item interactions are treated as implicit positive feedback, and each user’s interaction history is organized into sequences in chronological order. In this study, we focus solely on the *item cold-start* scenario among various cold-start settings [34]. Following the setup of [35] and [36], we adopted a single time split and a leave-one-out strategy for data partitioning. Specifically,

the entire dataset was divided into training and evaluation sets in a 7:3 ratio based on global timestamps [36].

To balance cold and warm items during evaluation, we randomly sampled the two groups at a 5:5 ratio from the evaluation set. Here, cold items are defined as those removed from the training set and appearing only in the evaluation set, thereby simulating a cold item environment. The splitting scheme is illustrated in Figure 2.

To mitigate sparsity and prevent data leakage during evaluation, we applied a minimum frequency filter (i.e., *5-core*) only to the training set partition. Thus, each user and item in the training data must appear at least five times, while interactions in the validation and test sets remain unchanged. We carefully conduct two sequential recommendation models to compare our approach, as shown below.

- **SASRec** : SASRec leverages the self-attention mechanism of the Transformer architecture to capture both short- and long-term dependencies within user interaction sequences. By incorporating positional embeddings, it accounts for the order of items while modeling interactions among all previous items. It has become a widely adopted and strong baseline in sequential recommendation tasks.
- **GRU4Rec** : GRU4Rec employs gated recurrent units (GRUs) to model user behavior sequences. It effectively captures temporal dependencies across items in a session, making it one of the earliest and most representative baselines in session-based recommendation research.

## 4.2 Evaluation Metric

In this evaluation, we use the ranking-based metrics **HitRate@{5,10,20}** and **NDCG@{5,10,20}** [37].

**HitRate@K** : HitRate measures whether the ground-truth item appears in the top- $k$  recommended list:

$$HR@k = \frac{1}{|U|} \sum_{u \in U} \mathbb{I}(\text{relevant item for } u \in R_u^k), \quad (5)$$

where  $U$  is the set of users,  $R_u^k$  is the top- $k$  recommended items for user  $u$ , and  $\mathbb{I}(\cdot)$  is the indicator function.

**NDCG@K** : Normalized Discounted Cumulative Gain (NDCG) considers the position of the relevant item in the ranking:

$$DCG@k = \sum_{i=1}^k \frac{rel_i}{\log_2(i+1)}, \quad (6)$$

$$IDCG@k = \sum_{i=1}^{|REL_k|} \frac{1}{\log_2(i+1)}, \quad (7)$$

$$NDCG@k = \frac{DCG@k}{IDCG@k}, \quad (8)$$

where  $rel_i$  denotes the relevance of the item at position  $i$ , and  $REL_k$  is the set of relevant items within the top- $k$  positions.

Table 2: Results of SASRec &amp; GRU4Rec

(a) SASRec Results

Dataset	HR@10				NDCG@10			
	Origin	Review-only	Meta-only	Rel. $\Delta$ (%)	Origin	Review-only	Meta-only	Rel. $\Delta$ (%)
Beauty	0.0103	0.0081	0.0125	35.20%	0.0050	0.0046	0.0070	34.29%
Sports&Outdoor	0.0051	0.0039	0.0065	40.00%	0.0025	0.0023	0.0036	36.11%
Digital Music	0.0415	0.0289	0.0433	33.26%	0.0181	0.0166	0.0241	31.12%
Musical Instruments	0.0210	0.0140	0.0280	50.00%	0.0073	0.0062	0.0120	48.33%

(b) GRU4Rec Results

Dataset	HR@10				NDCG@10			
	Origin	Review-only	Meta-only	Rel. $\Delta$ (%)	Origin	Review-only	Meta-only	Rel. $\Delta$ (%)
Beauty	0.0081	0.0049	0.0049	0.00%	0.0050	0.0020	0.0021	4.76%
Sports&Outdoor	0.0042	0.0031	0.0045	31.11%	0.0022	0.0018	0.0023	21.74%
Digital Music	0.0271	0.0072	0.0271	73.43%	0.0155	0.0039	0.0123	68.29%
Musical Instruments	0.0210	0.0140	0.0350	60.00%	0.0160	0.0114	0.0191	40.31%

### 4.3 Implementation Details

We aligned our implementation with the original paper and conducted all experiments under the same backbone. The maximum length of a user sequence was capped at 50; if a sequence exceeded this length, only the 50 most recent interactions were used. For the clustering-based analysis, we ran the k-means algorithm with k set to 8. For encoding the metadata and review data, we utilized potion-base-4M, a lightweight Hugging Face model. All metadata and review data were sourced from the entire dataset before any splits were made. The experiments were conducted on a single Quadro RTX 5000 GPU.

### 4.4 Results

At a glance, across our experiments, metadata integration tended to provide a more stable improvement, whereas the effect of review integration varied by model and domain. Given the specific splits, filtering, and candidate-generation settings used here, these observations should be taken as indicative rather than conclusive and would benefit from further validation.

#### 4.4.1 Performance Comparison

Based on the metrics presented in Table 2, the metadata-only approach (Meta-only) is observed to outperform the review-only approach (Review-only) in terms of HR@10 and NDCG@10 across multiple domains. For the SASRec model, the Meta-only approach demonstrates a substantial relative improvement (Rel.  $\Delta$ %) over the Review-only approach. For instance, in the Musical Instruments domain, the relative gains are approximately +50% in HR@10 and +48% in NDCG@10. Similar trends are noted in other domains, such as Sports & Outdoors, Digital Music, and Beauty, with relative improvements ranging from approximately 33–40% for HR@10 and 31–36% for NDCG@10.

A similar pattern is evident for the GRU4Rec model. Notably, in the Digital Music domain, the relative improvement of the Meta-only approach is high, reaching approximately 68–73% for both HR@10 and NDCG@10. In contrast, for the Beauty domain, both integration methods yield low absolute scores, with minimal or limited relative improvements reported. These

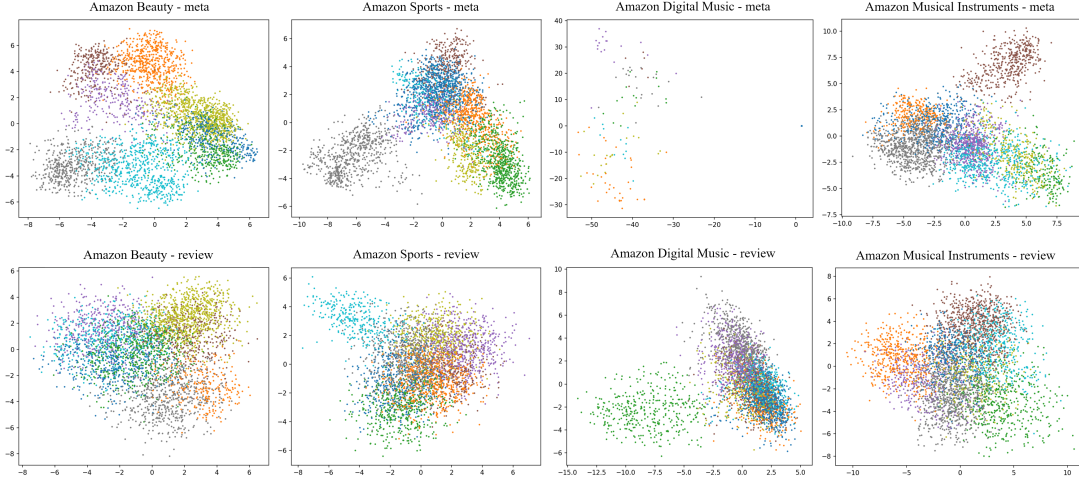


Figure 3: Visualization of clusters

Table 3: Clustering metrics by dataset

Dataset	$k$	Item-based			Review-based		
		Silh.	DBI	CH	Silh.	DBI	CH
Beauty	8	0.095	2.629	195.40	0.034	4.096	92.13
Sports&Outdoor	8	0.081	3.438	150.10	0.027	4.667	65.79
Digital Music	8	0.973	1.038	1753.26	0.046	4.531	79.24
Musical Instruments	8	0.114	2.580	224.24	0.054	3.500	135.55

$k$ : number of clusters.

**Silh.**:  $[-1, 1]$ , higher values indicate better cluster cohesion/separation.

**DBI**: lower is better (greater separation between clusters, stronger cohesion within clusters).

**CH**: higher is better (ratio of between-cluster dispersion to within-cluster dispersion).

findings suggest that in severe item cold-start scenarios, review texts, despite their rich contextual details, may struggle to provide a consistent advantage due to data sparsity. Conversely, structured metadata appears to function as a more stable and reliable auxiliary signal.

#### 4.4.2 Metadata and Reviews Embedding Analysis

Figure 3 visualizes the clustering results for review embeddings and metadata embeddings across the entire dataset. It is visually apparent that the clusters from the review-based embeddings, with the exception of the Digital Music category, exhibit a more entangled distribution compared to those from the metadata-based embeddings. However, as a purely visual analysis lacks definitive comparison, a quantitative evaluation was conducted using unsupervised clustering quality metrics (Silhouette, DBI, CH), as presented in Table 3.

As shown in Table 3, the item metadata-based embeddings tend to demonstrate superior clustering performance overall when compared to the review-based embeddings. This suggests that metadata embeddings form clusters with higher cohesion and separation. Conversely, the relatively lower clustering quality of the review embeddings is presumed to be attributable to domain-specific linguistic noise and data sparsity.

However, in the Digital Music domain, the metadata-based embedding recorded an anomalously high Silhouette score and a lower DBI compared to other datasets. This can be interpreted as a consequence of the titles and metadata within the Digital Music dataset. Specifically, the low-dimensional projection for k-means visually distorts the clusters, making them appear as a single merged entity by compressing and losing the fine-grained variance directions of the high-dimensional embeddings. Furthermore, the assumption of spherical or convex clusters fails to adequately capture the non-linear boundaries or density variations present in the data.

## 5 Conclusion

This study compared review texts and structured metadata by projecting them into a shared Pre-trained Language Model (PLM) embedding space under an item cold-start scenario. Based solely on the metrics presented, it is difficult to assert that the qualitative advantages of review embeddings consistently translate to superior performance over metadata embeddings. On the contrary, our findings indicate that metadata integration exhibited a more stable trend of performance enhancement across multiple domains. The results of the clustering analysis further support this, suggesting that metadata embeddings likely form more coherent representations.

In the future, instead of directly utilizing review embeddings, a potential direction is to explore embeddings that reduce sparsity and noise by leveraging metadata, such as titles and categories, as a guiding signal to align, distill, or smooth the review representations. This approach seems particularly relevant given that reviews are dependent on interactions, which introduces issues of sparsity and noise, and addresses the emerging concerns around re-identification risks when merging services (e.g., in the context of ATT/GDPR and the reliance on review data).

## 6 Acknowledgement

This research was supported by the Regional Innovation System & Education(RISE) program through the RISE Center, Gyeongsangnam-do, funded by the Ministry of Education(MOE) and the Gyeongsangnam-do Provincial Government, Republic of Korea.(2025-RISE-16-001)

## References

- [1] Yanbo Zhou, Gang-Feng Ma, Xilin Wen, Xu-Hua Yang, and Yi-Cheng Zhang. Sequential recommender systems: A methodological taxonomy and research frontiers. *Computer Science Review*, 59:100818, 2026.
- [2] J Ben Schafer, Joseph A Konstan, and John Riedl. E-commerce recommendation applications. *Data mining and knowledge discovery*, 5(1):115–153, 2001.
- [3] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Analysis of recommendation algorithms for e-commerce. In *Proceedings of the 2nd ACM Conference on Electronic Commerce*, pages 158–167, 2000.
- [4] Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z Sheng, and Mehmet Orgun. Sequential recommender systems: challenges, progress and prospects. *arXiv preprint arXiv:2001.04830*, 2019.
- [5] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. Facing the cold start problem in recommender systems. *Expert systems with applications*, 41(4):2065–2073, 2014.
- [6] Apple. A day in the life of your data. Technical report, Apple Inc., January 2021.
- [7] Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A practical guide, 1st ed.*, Cham: Springer International Publishing, 10(3152676):10–5555, 2017.

- [8] Christine Utz, Martin Degeling, Sascha Fahl, Florian Schaub, and Thorsten Holz. (un) informed consent: Studying gdpr consent notices in the field. In *Proceedings of the 2019 acm sigsac conference on computer and communications security*, pages 973–990, 2019.
- [9] Maciej Kula. Metadata embeddings for user and item cold-start recommendations. *arXiv preprint arXiv:1507.08439*, 2015.
- [10] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Jesús Bernal. A collaborative filtering approach to mitigate the new user cold start problem. *Knowledge-based systems*, 26:225–238, 2012.
- [11] Lei Zheng, Vahid Noroozi, and Philip S Yu. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the tenth ACM international conference on web search and data mining*, pages 425–434, 2017.
- [12] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 world wide web conference*, pages 1583–1592, 2018.
- [13] Hoang V. Dong, Yuan Fang, and Hady W Lauw. A contrastive framework with user, item and review alignment for recommendation. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, pages 117–126, 2025.
- [14] Arvind Narayanan and Vitaly Shmatikov. Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 111–125. IEEE, 2008.
- [15] Mishari Almishari and Gene Tsudik. Exploring linkability of user reviews. In *European Symposium on Research in Computer Security*, pages 307–324. Springer, 2012.
- [16] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 811–820, 2010.
- [17] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.
- [18] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE, 2018.
- [19] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019.
- [20] Jiacheng Li, Yujie Wang, and Julian McAuley. Time interval aware self-attention for sequential recommendation. In *Proceedings of the 13th international conference on web search and data mining*, pages 322–330, 2020.
- [21] Shereen Elsayed, Ahmed Rashed, and Lars Schmidt-Thieme. Context-aware sequential model for multi-behaviour recommendation. *arXiv preprint arXiv:2312.09684*, 2023.
- [22] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 1893–1902, 2020.
- [23] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. Sequence-aware recommender systems. *ACM computing surveys (CSUR)*, 51(4):1–36, 2018.
- [24] Emrul Hasan, Mizanur Rahman, Chen Ding, Jimmy Huang, and Shaina Raza. Based recommender systems: a survey of approaches, challenges and future perspectives. *ACM Computing Surveys*, 2024.
- [25] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM conference on recommender systems*, pages 101–109, 2019.
- [26] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: a factorization-

- machine based neural network for ctr prediction. *arXiv preprint arXiv:1703.04247*, 2017.
- [27] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, pages 7–10, 2016.
  - [28] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. Knowledge graph convolutional networks for recommender systems. In *The world wide web conference*, pages 3307–3313, 2019.
  - [29] Abiodun M Ikotun, Absalom E Ezugwu, Laith Abualigah, Belal Abuhaija, and Jia Heming. K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Information Sciences*, 622:178–210, 2023.
  - [30] David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227, 2009.
  - [31] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.
  - [32] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.
  - [33] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 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, 2015.
  - [34] Weizhi Zhang, Yuanchen Bei, Liangwei Yang, Henry Peng Zou, Peilin Zhou, Aiwei Liu, Yinghui Li, Hao Chen, Jianling Wang, Yu Wang, et al. Cold-start recommendation towards the era of large language models (llms): A comprehensive survey and roadmap. *arXiv preprint arXiv:2501.01945*, 2025.
  - [35] Anton Klenitskiy, Anna Volodkevich, Anton Pembek, and Alexey Vasilev. Does it look sequential? an analysis of datasets for evaluation of sequential recommendations. In *Proceedings of the 18th ACM Conference on Recommender Systems*, pages 1067–1072, 2024.
  - [36] Jianling Wang, Haokai Lu, James Caverlee, Ed H Chi, and Minmin Chen. Large language models as data augmenters for cold-start item recommendation. In *Companion Proceedings of the ACM Web Conference 2024*, pages 726–729, 2024.
  - [37] Yan-Martin Tamm, Rinchin Damdinov, and Alexey Vasilev. Quality metrics in recommender systems: Do we calculate metrics consistently? In *Proceedings of the 15th ACM conference on recommender systems*, pages 708–713, 2021.