SDSERF: Sequential Dual-Stage Embedding Retrieval Framework for Obscure Query Matching

Abstract

E-commerce platforms increasingly leverage natural language processing (NLP) to enhance product retrieval and recommendation tasks. Traditional keyword-based systems often fail to capture the nuanced semantics of user queries, especially in complex or ambiguous contexts. To address this challenge, we utilize the Amazon-C4 test set from McAuley's Lab, which comprises ambiguous queries to simulate real-world scenarios.

We propose a novel two-step retrieval framework that combines category prediction using a fine-tuned BERT model with similarity matching based on the E5 embedding model. This approach optimizes retrieval efficiency while maintaining high accuracy, achieving a test set accuracy of 74.07% for top-200 retrieval tasks.

Furthermore, we introduce a strategy to enrich the dataset by incorporating top-ranked items from our model's predictions as additional ground-truth candidates. This paper provides qualitative examples and actionable insights into how dataset construction impacts model performance, laying the groundwork for enhancing future product retrieval systems.

All code and LaTeX source are available here.

1 Introduction

Language plays a pivotal role in e-commerce platforms, serving as a key modality for describing and retrieving products. Tasks such as product retrieval and recommendation are increasingly reliant on advanced language modeling techniques [1, 2]. Traditional recommendation systems have often relied on keyword-based features, which fail to capture the nuanced semantics of natural language.

With the advent of large language models (LLMs)[3, 12], there is a growing interest in leveraging their semantic capabilities to enhance recommendation systems[5]. However, integrating LLMs into large-scale recommendation scenarios involving millions of items remains a significant challenge due to computational expenses. Additionally, user queries in real-world applications are often obscure and lack sufficient context, making it difficult for models to accurately interpret and retrieve the intended items.

Existing methods for such tasks typically follow one of two approaches:

- End-to-end neural models fine-tuned on task-specific data, which often lack generalizability across diverse tasks and domains [2].
- Pre-trained language models (PLMs) used to generate text embeddings, which are not specifically tailored for recommendation contexts, leading to suboptimal performance, especially when dealing with obscure queries [5].

To address these challenges, we propose a novel two-step retrieval framework that integrates pre-trained models with category prediction and similarity matching to enhance retrieval efficiency and accuracy. This framework is designed to capture the complex semantics of user queries while optimizing computational resources. Moreover, we analyze dataset limitations and suggest a strategy to enrich the dataset for improved retrieval experiments.

The main contributions of our work are as follows:

- **Novel Retrieval Framework:** We propose a two-step retrieval framework that combines category prediction using a fine-tuned BERT model and similarity matching with the E5 embedding model. This framework efficiently narrows the search space while maintaining high accuracy.
- **Dataset Analysis and Improvement:** We identify that the Amazon-C4 dataset contains metadata that is too short to meaningfully describe items. By removing such cases, we achieved a slight improvement in accuracy from 73.19% to 74.07%. Representative examples are discussed in the exploratory data analysis section.
- **Dataset Enrichment Proposal:** We recommend enriching the dataset by leveraging the model's predictions. Specifically, the top-5 most relevant items retrieved for each query can serve as supplementary candidates. This approach could enhance future dataset quality by improving recall and mean reciprocal rank in retrieval tasks.
- **Insightful Examples:** Detailed examples of queries, top-ranked items, and their metadata are provided in the Appendix to illustrate the qualitative relevance of the retrieved results.

2 Dataset analysis

2.1 Dataset Description

In this study, we utilize two datasets to evaluate product search performance under complex contexts:

2.1.1 Amazon-C4 Dataset.

The Amazon-C4 dataset, developed by the McAuley Lab, is designed to assess a model's ability to comprehend complex language contexts and retrieve relevant items. It comprises 21,223 user reviews from the Amazon Reviews 2023 dataset, each rephrased by ChatGPT into vague, first-person queries. These queries are paired with corresponding item identifiers, facilitating the evaluation of product search tasks. The dataset is publicly available on Hugging Face at the following link: https://huggingface.co/datasets/McAuley-Lab/Amazon-C4 [6].

2.1.2 Sampled Item Metadata Dataset.

The `sampled_item_metadata_1M.jsonl` file contains around 1 million items sampled from the Amazon Reviews 2023 dataset. Each entry includes:

- **item_id**: A unique identifier corresponding to the `parent_asin` in the original dataset.
- **category**: The item's category, useful for evaluating model performance within specific domains.
- **metadata**: A concatenation of the item's title and description from the original metadata.

This sampled item pool is utilized for evaluation in the BLaIR paper, providing a comprehensive set of items for retrieval tasks. The dataset can be accessed from Hugging Face at: https://huggingface.co/datasets/McAuley-Lab/Amazon-C4 [7].

2.2 Exploratory Data Analysis

As shown in Figure 1, over 90% of metadata lengths are below 400 tokens, with a small number of outliers exceeding 2000 tokens. However, certain metadata entries are excessively short, providing insufficient information about the associated items, which could hinder effective analysis. This will be addressed later in our data processing and improvement is demonstrated in our model evaluation.

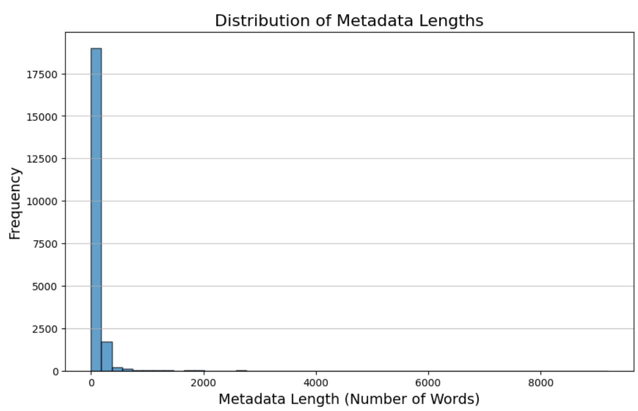

Figure 1. Distribution of Metadata Lengths.

Figure 2 demonstrates that all query lengths are under 220 tokens, with the majority falling well below this threshold. Despite this, short metadata lengths remain a challenge as they can obscure a clear understanding of the items.

Figure 3 illustrates the number of items available across various categories. The "Home" category dominates with the highest number of items, exceeding 3,000, while categories like "Beauty" and "Gift" have significantly fewer items,

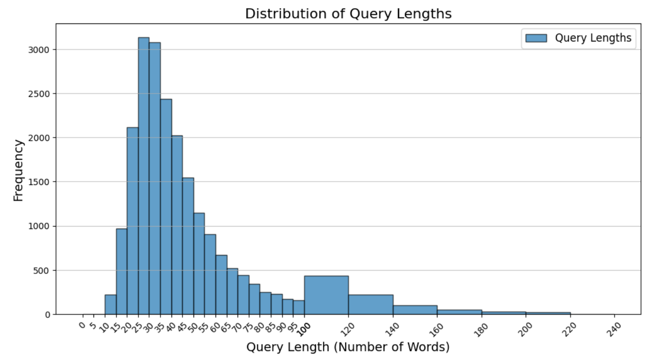

Figure 2. Distribution of Query Lengths.

showing a clear imbalance in category representation. Such distribution indicates that certain categories, such as "Home" and "Electronics," might play a larger role in the dataset, potentially influencing retrieval tasks.

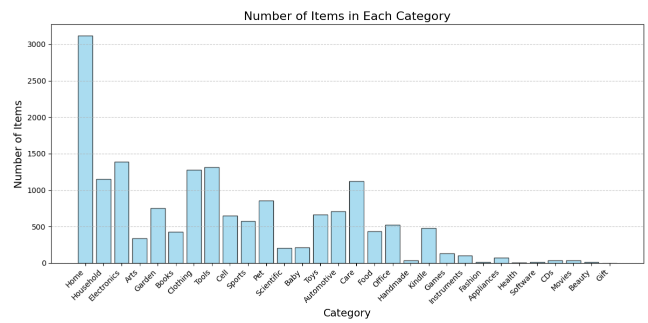

Figure 3. number of items in different categories.

2.3 Dataset Merging and Motivation

The Amazon-C4 Dataset and the `sampled_item_metadata_1M` dataset serve complementary roles in facilitating our analysis. While the Amazon-C4 Dataset provides user queries and their corresponding ground-truth `item_ids`, it lacks detailed descriptions of the items themselves, such as their attributes or categories. This information is crucial for understanding the characteristics of the items that users are interested in and for designing systems capable of accurately matching queries to relevant items.

The `sampled_item_metadata_1M` dataset addresses this gap by offering rich metadata and categorical information for approximately 1 million items.

To fully leverage the strengths of both datasets, we merge them by matching the `item_ids` in the Amazon-C4 Dataset with those in the `sampled_item_metadata_1M` dataset. For instance, consider the following entry from the Amazon-C4 Dataset:

```
{
    "qid": 0,
```

```
    "query": "I need filters that effectively...",
    "item_id": "B0C5QYYHTJ",
    "user_id": "AGREO2G3GTRNYOJK4CIQV2DTZLSQ",
    "ori_rating": 5,
    "ori_review": "These filters work..."
}
```


The ground-truth `item_id` in this entry corresponds to the following entry in the `sampled_item_metadata_1M` dataset:


```
{
    "item_id": "B0C5QYYHTJ",
    "category": "Home",
    "metadata": "Flintar Core 300 True HEPA..."
}
```


By combining the `query`, `item_id`, `category`, and `metadata`, we construct a unified entry, such as:


```
{
    "query": "I need filters that effectively...",
    "item_id": "B0C5QYYHTJ",
    "category": "Home",
    "metadata": "Flintar Core 300 True HEPA..."
}
```


This merging process allows us to enrich the query data with additional contextual information about the items, enabling a more comprehensive evaluation of query-to-item relevance. It also facilitates downstream tasks, such as identifying item features most relevant to user queries or categorizing user preferences.

3 Problem Definition

In this work, we aim to address the challenge of retrieving the most relevant items based on an obscure user query using similarity-based retrieval techniques. Specifically, we have a dataset consisting of 21,223 queries, each paired with corresponding item metadata. The objective is to leverage the metadata to accurately identify and recommend the most relevant item for each user's query.

This problem can be framed as a query-to-item matching task, where the goal is to maximize the semantic relevance between the query and the retrieved metadata. Given the diverse and obscure nature of user queries, as well as the varying quality and length of metadata, achieving high accuracy requires a robust and efficient retrieval framework. Our approach employs both pre-trained language models and domain-specific optimizations to address these challenges effectively.

4 Baseline Model: TF-IDF with Cosine Similarity

As a baseline, we implement a simple yet effective model using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization combined with cosine similarity for passage ranking. This approach is designed to provide a point of comparison for our deep learning-based methods.

4.1 TF-IDF Vectorization

TF-IDF is a widely used technique for representing text data in information retrieval tasks. It computes a weighted representation of terms, emphasizing terms that are frequent in a specific document but rare across the corpus. Formally, the TF-IDF score for a term t in a document d is defined as:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \log\left(\frac{N}{\text{DF}(t)}\right),$$

where:

- $\text{TF}(t, d)$: Term frequency of t in d.
- $\text{DF}(t)$: Document frequency of t (number of documents containing t).
- N: Total number of documents.

We use the `TfidfVectorizer` from the Scikit-learn library to compute TF-IDF vectors for both queries and passages.

4.2 Cosine Similarity for Ranking

To rank passages for a given query, we compute the cosine similarity between the TF-IDF vector of the query and each passage. Cosine similarity is defined as:

$$\text{Sim}(\mathbf{q}, \mathbf{p}) = \frac{\mathbf{q} \cdot \mathbf{p}}{\|\mathbf{q}\| \|\mathbf{p}\|},$$

where $\mathbf{q}$ and $\mathbf{p}$ are the TF-IDF vectors of the query and passage, respectively. Since TF-IDF vectors are L2-normalized by default, the dot product directly yields cosine similarity.

4.3 Performance Evaluation

For each query, the top 200 passages with the highest cosine similarity scores are retrieved. The accuracy of the baseline is measured by checking if the ground truth `item_id` appears among the top-200 ranked passages. While simple, this approach achieves a top-200 accuracy of 42.34%, serving as a benchmark for evaluating the effectiveness of advanced models.

5 Model Structure and evaluations

Our model, in Figure 4 consists of two parts. Due to the large size of the dataset, we first fine-tune a pre-trained BERT model to predict the category of the item that a user wants based on their query.

Next, we use the e5 model to compute the similarity between the query and all items belonging to the top-2 most likely categories predicted by the BERT model.

Finally, we evaluate the probability of the ground truth `item_id` appearing in the top-200 ranked items. The test set accuracy reaches 74.07%.

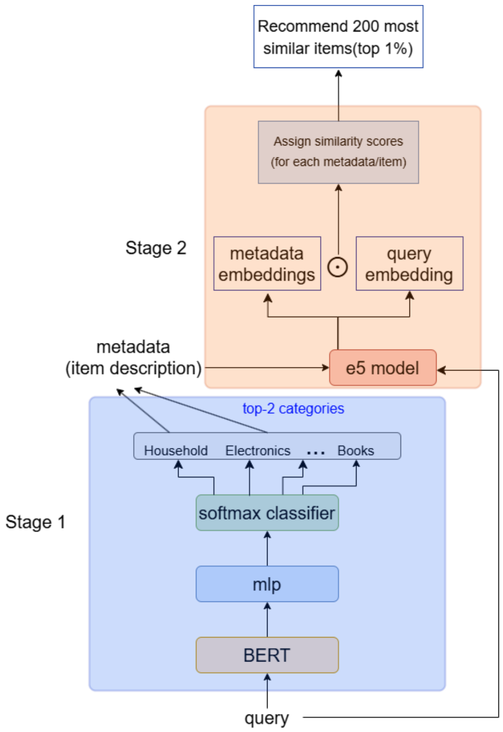

Figure 4. Overview of the model structure, including category prediction with BERT and similarity matching with E5.

5.1 Data Splitting and Preprocessing

The cleaned dataset is first shuffled randomly and then split into training and test sets with a 9:1 ratio. This ensures that the model is evaluated on unseen data, maintaining the integrity of the evaluation process.

5.2 Part 1: Category Prediction with BERT

The first stage of the model is a classifier that predicts the category of the desired item based on the user's query. We employ a pre-trained `bert-base-uncased` model, which is developed by *Devlin et al.* [4]. BERT (Bidirectional Encoder Representations from Transformers) is a transformer-based language model that captures bidirectional context, making it highly effective for understanding complex language semantics.

In our setup, the BERT model is fine-tuned, and a softmax classifier is added on top to predict the category. The model learns to associate semantic patterns in the query text with specific categories. After a training of 3 epochs as shown in Figure 5 the classifier achieves a probability of 93.67% on the test set for correctly identifying the true category within the top-2 predicted categories.

5.3 Part 2: Similarity Matching with E5

The second part of the model uses the E5 embedding model to compute the similarity between the user's query and the metadata of items within the top-2 predicted categories. E5, as proposed in *Text Embeddings by Weakly-Supervised Contrastive Pre-training* [13], is a general-purpose embedding model designed to generate high-quality semantic embeddings for various tasks, including information retrieval and recommendation systems. By projecting both queries and metadata into the same semantic space, E5 enables effective similarity matching.

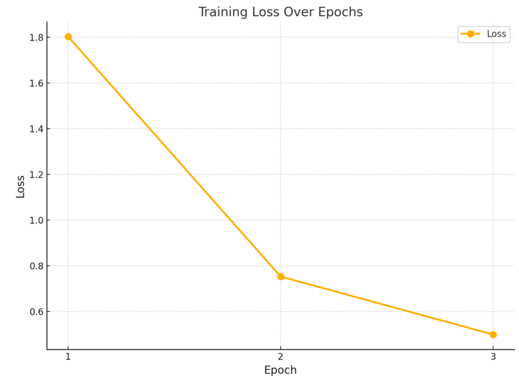

Figure 5. training classifier

For each query in the test set, we compute the embeddings for the query and all items in the top-2 categories. The similarity scores are calculated between the query embedding and each item's metadata embedding. Items are then ranked by their similarity scores. One example of a random query is shown in Figure 6

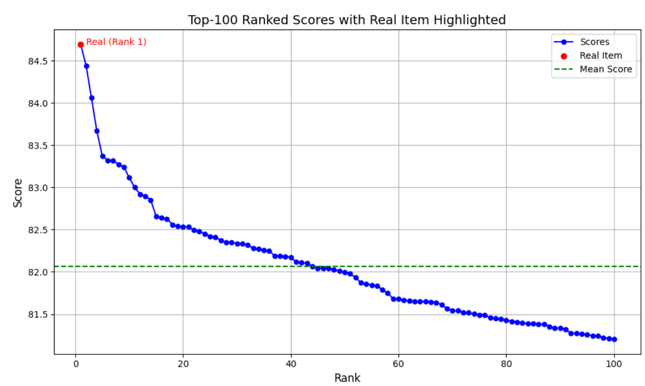

Figure 6. Top-100 ranked score of metadata with real item highlighted for a random query.

5.4 Evaluation

5.4.1 Top-200 Evaluation.

Finally, we evaluate the model's performance by checking whether the ground truth `item_id` appears in the top-200 ranked items (approximately the top 1%). This metric provides insight into the model's ability to retrieve the most relevant items for a given query. The test set accuracy for this evaluation reaches 74.07%, demonstrating the effectiveness of our two-step approach.

5.4.2 Importance of Data Cleaning and Classifier.

As mentioned earlier, some items in the dataset contain problematic metadata, such as empty metadata, metadata with only a title (Books or Movies), or metadata that is excessively short. These issues make it impossible to use similarity-based matching effectively. To address this, we removed all entries with metadata lengths shorter than 10. This data cleaning step reduced the dataset size from 21,223 to 20,250 items. After training, the top-200 accuracy improved from 73.19% to 74.07%.

Additionally, we experimented with skipping the classifier (Stage 1) and directly proceeding to Stage 2. This approach significantly increased training time and caused CUDA out-of-memory errors. Moreover, the accuracy dropped substantially, achieving only 65.21%. These results demonstrate the critical importance of our two-stage framework, which not only improves computational efficiency but also enhances accuracy.

Table 1. Top-200 Accuracy of Different Models and Datasets

Model / Dataset	**Top-200 Accuracy (%)**
Baseline Model (TF-IDF)	42.43
Raw Dataset	73.19
Without Classifier	65.21
Cleaned Dataset with classifier	**74.07**

5.4.3 Further Evaluation.

Beyond this quantitative evaluation, we conducted a qualitative analysis by randomly selecting several queries and examining the top-5 most relevant items ranked by the model. For these top-5 items, we analyzed their `metadata` (see Appendix for detailed examples). We observed that the metadata of these items was highly relevant to the query requirements, even though the ground truth `item_id` was not always included in the top results.

We believe this phenomenon is closely related to the construction of the dataset itself. Since the queries were generated by modifying reviews, they may not fully capture the precise requirements of the ground truth items. As a result, the accuracy did not improve further, despite the model's strong capability to retrieve relevant items.

To address this limitation, we propose an enhancement to the dataset: leveraging our model's predictions. Specifically, by selecting the top-5 most relevant items identified by our approach and using them as supplementary candidates for the dataset, the original dataset could be enriched and better aligned with the actual query intent.

Detailed examples of queries, their top-5 relevant items, and the associated metadata are included in the Appendix to provide further insights into the proposed improvement.

6 Related Work

6.1 related model

6.1.1 Language Models in Recommendation Systems.

Integrating language models into recommendation systems has gained significant traction. Hou et al. introduced BLaIR, a series of pretrained sentence embedding models tailored for recommendation scenarios. BLaIR effectively captures correlations between item metadata and natural language contexts, enhancing retrieval and recommendation tasks [5]. Similarly, Cheng et al. proposed TransRec, a paradigm that employs multi-facet identifiers to bridge large language models (LLMs) with recommendation systems, achieving both distinctiveness and semantic richness in item indexing [8].

6.1.2 Contrastive Learning in Sequential Recommendation.

Contrastive learning has emerged as a powerful technique in sequential recommendation systems. Xu et al. introduces the CL4SRec model, which combines next-item prediction with contrastive learning to enhance user representation learning in recommendation systems. [14]. Zhang et al. provided a comprehensive survey on contrastive self-supervised learning in recommender systems, highlighting its potential in addressing data sparsity and cold-start problems [10].

6.2 related dataset

6.2.1 MS MARCO (Microsoft MAchine Reading COmprehension).

The MS MARCO dataset [11] is a large-scale machine reading comprehension dataset designed for training and evaluating retrieval and question-answering models. It consists of real-world user queries sampled from the Bing search engine, paired with a large collection of passages extracted from web documents. MS MARCO provides relevance annotations for the passage retrieval task, where each query is associated with a set of relevant passages and a much larger pool of irrelevant passages.

Statistics:

- **Passages:** Approximately 8.8 million passages, extracted from millions of web pages.
- **Queries:** Over 500,000 real-world queries, each labeled with at least one relevant passage.

Key Features:

- The dataset contains diverse query types, ranging from keyword-based searches (e.g., "best laptops under $500") to natural language questions (e.g., "What is the capital of France?").
- It supports various tasks, including passage ranking, document ranking, and question answering.
- The negative samples, often randomly sampled from the corpus, provide a robust training signal for retrieval models.

MS MARCO has become a standard benchmark for evaluating information retrieval and retrieval-augmented question-answering systems, particularly for models like ColBERT, DPR, and DensePassageRetrieval.

6.2.2 Natural Questions (NQ).

The Natural Questions (NQ) dataset [9] is a large-scale benchmark for open-domain question answering, developed by Google. It consists of real user queries from Google Search, paired with long-form documents (primarily Wikipedia articles) that contain the answers. NQ is specifically designed to evaluate a model's ability to retrieve relevant documents and locate specific answers within those documents.

Statistics:

- **Questions:** Approximately 300,000 questions, sampled from real-world search engine logs.
- **Documents:** Wikipedia articles, annotated with gold-standard answers and supporting evidence.

Key Features:

- The dataset supports both retrieval and question answering tasks:
  - In the retrieval task, models must identify relevant Wikipedia articles for a given question.
  - In the reading comprehension task, models must extract a precise answer from the retrieved article.
- Queries range from simple fact-based questions (e.g., "Who is the president of the United States?") to complex, multi-faceted questions requiring synthesis of information (e.g., "Explain the causes of the French Revolution.").
- Annotations include both short-form answers (e.g., specific entities) and long-form answers (e.g., descriptive text).

NQ serves as a benchmark for retrieval-focused models like ColBERT and end-to-end question answering systems. It is particularly challenging due to the diversity of query types and the requirement for precise document retrieval.

7 Result and Conclusion

This work presents a novel two-step retrieval framework to address challenges in generalizing the obscure query retrieval framework, leveraging fine-tuned BERT and E5 embedding models to achieve a balance between computational efficiency and semantic richness. Our results demonstrate the effectiveness of this approach, achieving a 74.07% accuracy in retrieving ground-truth items within the top-200 results. Beyond quantitative evaluation, our qualitative analysis highlights the relevance of retrieved items even when ground truth is absent, suggesting a need for improved dataset design.

Through exploratory data analysis, we identified critical limitations in the Amazon-C4 dataset, particularly the impact of short metadata on retrieval accuracy. Addressing these limitations by excluding inadequate entries led to measurable performance improvements. Furthermore, we proposed a dataset enrichment strategy that incorporates top-ranked items predicted by our model to enhance the dataset for future retrieval tasks.

By bridging semantic gaps in product search and proposing actionable enhancements to dataset construction, this work provides both practical methodologies and a foundation for advancing retrieval systems. Future research could explore integrating our enriched dataset with large language models for further improvements in recall and user experience.

References

[1] Qingyao Ai, Yongfeng Zhang, et al. Learning a hierarchical embedding model for personalized product search. In *Proceedings of SIGIR*, 2017.

[2] Keping Bi, Qingyao Ai, and W. Bruce Croft. A transformer-based embedding model for personalized product search. In *Proceedings of SIGIR*, 2020.

[3] Aakanksha Chowdhery et al. Palm: Scaling language modeling with pathways. *JMLR*, 2023.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[5] Yupeng Hou et al. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2303.00345*, 2024.

[6] McAuley Lab Hugging Face. Amazon-c4 dataset. https://huggingface.co/datasets/McAuley-Lab/Amazon-C4.

[7] McAuley Lab Hugging Face. Sampled item metadata dataset. https://huggingface.co/datasets/McAuley-Lab/Amazon-C4.

[8] Mingyue Cheng Joemon M. Jose Chenyun Yu Beibei Kong Xiangnan He Zhijin Wang Bo Hu Zang Li Jie Wang, Fajie Yuan. Transrec: Learning transferable recommendation from mixture-of-modality feedback. *arXiv preprint arXiv:2206.06190*, 2022.

[9] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, et al. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

[10] Tianzi Zang Ke Wang Mengyuan Jing, Yanmin Zhu. Contrastive self-supervised learning in recommender systems: A comprehensive survey. *ar Xiv preprint arXiv:2305.03456*, 2023.

[11] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. Ms marco: A human-generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*, 2016.

[12] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

[13] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*, 2024.

[14] Zhaoyang Liu Shiwen Wu Jinyang Gao Bolin Ding Bin Cui Xu Xie, Fei Sun. Contrastive learning for sequential recommendation. *Proceedings of the ACM Conference on Recommender Systems (RecSys)*, 2021.

A Example Queries and Top-5 Results

A.1 Query 1: Shoes for Work and After-Work Wear

Query: I am looking for shoes that are super comfy, fit wonderfully, and can be paired with professional attire for work as well as after-work wear. I want them to be the perfect height, have a neutral color, and fit well. What more can a girl ask for in a shoe?

Top-5 Ranked Metadata:

1. **Rank 1:** Ryka Women's Devotion Plus 2 Walking Shoe. Get the comfort and performance you need every time you exercise in this light and comfortable walking sneaker with exceptional cushion, shock absorption, and our powerful Made for Women fit. BEST FOR: High-performance fitness walking. PERFORMANCE TECH: RE-ZORB® responsive cushioning for shock absorption + impact protection. MADE FOR WOMEN FIT: Designed for a woman's unique foot shape, muscle movement, and build with a narrower heel, roomier toe, and softer foot cushioning. MATERIALS: Breathable engineered mesh + soft Lycra-lined tongue and collar with built-in cushion. CLOSURE: Lace-up front for a secure fit. INSOLE: Anatomical insole with extra arch + heel support. MIDSOLE: Lightweight EVA for soft cushioning. OUTSOLE: Eight-piece rubber sole for increased traction + durability. WEIGHT: 224 g/7.9 oz per shoe. HEEL-TO-TOE DROP: 11 mm.
2. **Rank 2:** quescu 2Pcs Valentine Gnomes Plush, Valentines Day Gnomes Decor Ornaments, Sweet Valentines Day Gifts for Him Her, Tiered Tray Party Decor Home Table Decorations (2pcs).
3. **Rank 3:** Bico Christmas Gnomes Ceramic Spoon Rest, House Warming Gift, Dishwasher Safe.
4. **Rank 4:** LEVKIDS Christmas Stocking, Swedish Gnome and Snowman Pattern Xmas Stocking, Holiday Party Decorations Fireplace Hanging Ornaments, Pack of 2.
5. **Rank 5:** D-FantiX Gnome Christmas Tree Topper, 27.5 Inch Large Swedish Tomte Gnome Christmas Ornaments Santa Gnomes Plush Scandinavian Christmas Decorations Holiday Home Décor with Plaid Hat.

A.2 Query 2: Cute Decorations with Gnomes

Query: I'm looking for cute and well-made decorations that can add instant adorable-ness to any space. I want something with floppy hats and soft beards, like gnomes. They should have their own unique styles and be decorative and cheery. I plan to add them to my growing collection of decorations. It would be great if they arrive promptly and are well packaged. I'm also looking for a good price point. Highly recommend!

Top-5 Ranked Metadata:

1. **Rank 1:** 3 Pack Christmas Gnomes Decorations Handmade Santa Gnomes Plush Swedish Tomte Elf Ornaments Scandinavian Christmas Decorations Indoor Home Decor for Shelf Table Fireplace Christmas Tree Xmas Gift.
2. **Rank 2:** Hey Dude Women's Wendy Lace-Up Loafers Comfortable & Lightweight Ladies Shoes Multiple Sizes & Colors.
3. **Rank 3:** konhill Women's Casual Walking Shoes Breathable Mesh Work Slip-on Sneakers.
4. **Rank 4:** Shoe Stretcher Women, 4-way Shoe Widener Expander Shoe Tree Shape for Wide Feet.
5. **Rank 5:** somiliss Chunky Sneakers for Women High Top Lace Up Shoes for Women Sneakers Nice Women's Shoes Chunky Trainers Female Sneakers.

A.3 Query 3: Mini Filter for Betta Fish

Query: I am looking for the best mini filter for my Betta fish. It should have adjustable flow since Betta fish don't require a lot of flow.

Top-5 Ranked Metadata:

1. **Rank 1:** AQUANEAT Mini Sponge Filter, Aquarium Sponge Filter for Betta Fish Tank with Airline Tubing and Control Valve, up to 3Gal.
2. **Rank 2:** Kucbraly Fish Tank Filter Cartridge for Aqueon Filter Cartridges.
3. **Rank 3:** FS-TFC 6-Stage Portable Water Filter 0.01 Micron UF and CTO Improving Tastes Water Purifier Survival Gear 1.5L/Min Fast Flow for Hiking, Camping, Travel, and Emergency Preparedness.
4. **Rank 4:** Ameliade Aquarium Decorations Fish Tank Artificial Plastic Plants & Cave Rock Decor Set, Goldfish Betta Fish Tank Accessories Small & Large Fish Bowl Decorations (8PCS).
5. **Rank 5:** CousDUoBe 2 Pack Betta Fish Leaf Pad Improves Betta's Health by Simulating The Natural Habitat - Natural, Organic, Comfortable Rest Area for Fish Aquarium.