GrocLM: Grocery Category Recommendation in E-Commerce with Large Language Models

Yuan Zhong[1] **Chuanwei Ruan**[2] **Moein Hasani**[2]
Tejaswi Tenneti[2] **Haixun Wang**[3] **Fenglong Ma**[1*]
[1]The Pennsylvania State University, USA
[2]Instacart, USA
[3]Evenup, USA
[1]{yfz5556, fenglong}@psu.edu
[2]{chuanwei.ruan, moein.hasani, tejaswi.tenneti}@instacart.com
[3]haixun.wang@evenup.ai

Abstract

The rapid growth of online grocery shopping requires recommendation systems that capture cyclical purchasing behavior and diverse user intents. Traditional item-level methods face scalability and accuracy challenges, motivating category-level recommendation as a more structured and practical alternative. We present GROCLM, a fine-tuned language model for grocery category recommendation in a real-world production environment. GROCLM employs a two-stage LoRA-based training strategy to encode cyclical purchasing patterns directly into model parameters, enabling more effective utilization of rebuying signals compared to prompt-based conditioning. To ensure valid and controllable outputs, we further introduce a trie-based constrained decoding mechanism over a predefined category space. Experiments on both proprietary production data and a public benchmark demonstrate that GROCLM consistently outperforms strong baselines. In a live production restocking task, GROCLM achieves a 7.5% relative improvement in cart-adds per impression, while maintaining efficient inference by generating all categories jointly. These results highlight the effectiveness and practicality of integrating large language models into structured recommendation systems.

1 Introduction

At a large-scale online grocery e-commerce platform, we serve millions of customers daily across thousands of retailers. Unlike traditional e-commerce, grocery shopping is mission-driven and highly repetitive: users frequently repurchase household staples and build baskets spanning multiple complementary categories. This recurring nature requires recommendation systems that model structured shopping habits and replenishment patterns, not just isolated item preferences.

Our platform adopts a hybrid strategy combining item-level and carousel-level models. While item-level recommendation remains essential, category-level recommendation provides a more scalable and business-aligned solution. On the storefront, users interact with category-based carousels that structure product exposure and ensure inventory availability. In practice, category recommendation is typically implemented in a *top-down* manner: predicting categories first and then populating them with items, because grocery scenarios require strict diversification and availability constraints that bottom-up aggregation struggles to satisfy.

Category-level modeling is also better aligned with grocery behavior. Repurchasing patterns often emerge at the category level rather than the item level: while specific products (e.g., *Gala Apple* vs. *Fuji Apple*) may vary, demand for the broader *Apple* category remains stable. Moreover, grocery baskets are typically large and span complementary categories, requiring models to understand sequential intent and evolving shopping missions.

However, designing effective grocery category recommendation models is non-trivial. Existing LLM-based generative recommendation approaches (Rajput et al., 2024; Si et al., 2023; Geng et al., 2022; Zheng et al., 2024; Tan et al., 2024; Wang et al., 2024) are primarily developed for item-level prediction and do not explicitly model cyclical repurchasing behavior. Semantic ID-based methods further introduce potential noise through intermediate ID mapping. Directly feeding purchase histories into LLMs is appealing, yet unconstrained generation produces free-form text that conflicts with the mutually exclusive and discrete nature of category labels. These challenges call for a tailored solution.

We propose GROCLM, a prompt-tuned **L**anguage **M**odel for **Groc**ery category recommendation. Built on a LLAMA 3 (Dubey et al., 2024) backbone with Low-Rank Adapters (LoRA) (Hu

*Corresponding Author.

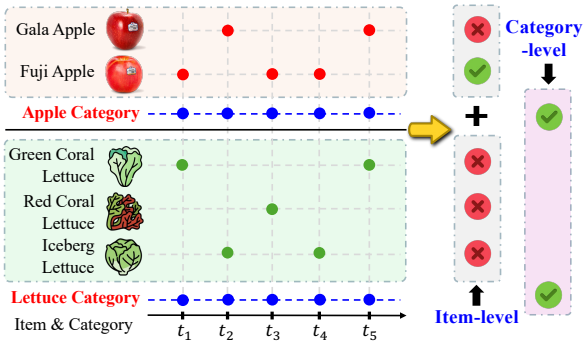

Figure 1: Illustration of grocery category recommendation: aggregating items into categories reveals clear repurchasing patterns.

et al., 2021), GROCLM adopts a two-stage fine-tuning strategy.

In Stage 1, we model **category-level repurchasing behavior** using pre-computed rebuying statistics, enabling the model to capture high-level cyclical patterns and improve robustness under sparse histories. In Stage 2, the model learns **implicit relationships among categories** from sequential purchasing data, augmented with user query context to better reflect immediate intent. To ensure valid and discrete outputs, we introduce a trie tree-based masking mechanism (De Cao et al., 2020) with beam search to constrain decoding to legitimate category labels.

The key contributions of this work are:

- We formalize grocery category recommendation as a practical task characterized by cyclical purchasing patterns, diverse user queries, and discrete output constraints.
- We propose GROCLM, a two-stage prompt-driven LLM framework leveraging rebuying statistics, sequential behavior modeling, and trie-constrained decoding.
- We validate GROCLM on both proprietary and public datasets, demonstrating significant improvements over strong baselines and measurable gains in production metrics.

2 Related Work

Generative Recommendation with LLMs. Recent advances in large language models (LLMs) (Deldjoo et al., 2024; Xiao et al., 2022; Chen et al., 2024; Zhao et al., 2024; Wu et al., 2024; Zhang et al., 2023; Wu et al., 2021) have significantly influenced recommendation systems. Leveraging pre-trained encoders and decoders, these models support diverse outputs, including item identifiers (Harte et al., 2023; Mao et al., 2023; Sanner et al., 2023; Sileo et al., 2022), user ratings (Bao et al., 2023; Kang et al., 2023), and text-based recommendations (Li et al., 2020, 2023; Ni et al., 2019; Hada and Shevade, 2021). Their flexibility enables zero-shot recommendation (Dai et al., 2023; Kang et al., 2023; Zhang et al., 2023; Liu et al., 2023; Sanner et al., 2023; Sileo et al., 2022) and domain adaptation through fine-tuning (Bao et al., 2023; Cui et al., 2022; Hua et al., 2023; Geng et al., 2022; Harte et al., 2023). Generative recommender models extend these capabilities by directly generating item identifiers, reducing reliance on traditional ranking pipelines. Approaches incorporating semantic identifiers (Geng et al., 2022; Rajput et al., 2024; Tan et al., 2024), collaborative filtering (Zheng et al., 2024; Wang et al., 2024; Khattab and Zaharia, 2020; Li et al., 2021; He et al., 2020), LLM-based architectures (Bao et al., 2024; Kim et al., 2024), and autoencoder techniques (Si et al., 2023) have streamlined recommendation workflows. Despite these advances, grocery recommendation remains underexplored. Off-the-shelf LLMs are pre-trained on general-domain corpora (Deldjoo et al., 2024), making grocery data inherently out-of-distribution. Additionally, privacy constraints limit large-scale domain-specific pre-training. As a result, challenges such as cyclical purchasing behavior and diverse user queries are insufficiently addressed.

Pre-LLM Retrieval and Recommendation Methods. Prior to LLM-based approaches, sparse retrieval methods such as vector space models (Salton, 1962; Salton et al., 1975), TF-IDF (Aizawa, 2003; Ramos et al., 2003; Robertson, 2004), and inverted indices (Zobel and Moffat, 2006; Zobel et al., 1998) dominated search and recommendation. While computationally efficient, these methods relied heavily on keyword matching and lacked semantic understanding. Learning-to-rank techniques (Liu et al., 2009) improved relevance through supervised learning but remained constrained by sparse representations. Dense retrieval models (Huang et al., 2013; Guo et al., 2016, 2019; Mitra and Craswell, 2017; Kang and McAuley, 2018) later introduced learned embeddings to capture semantic similarity. Architectures such as Two Tower models and self-attentive sequential networks better modeled user behavior and

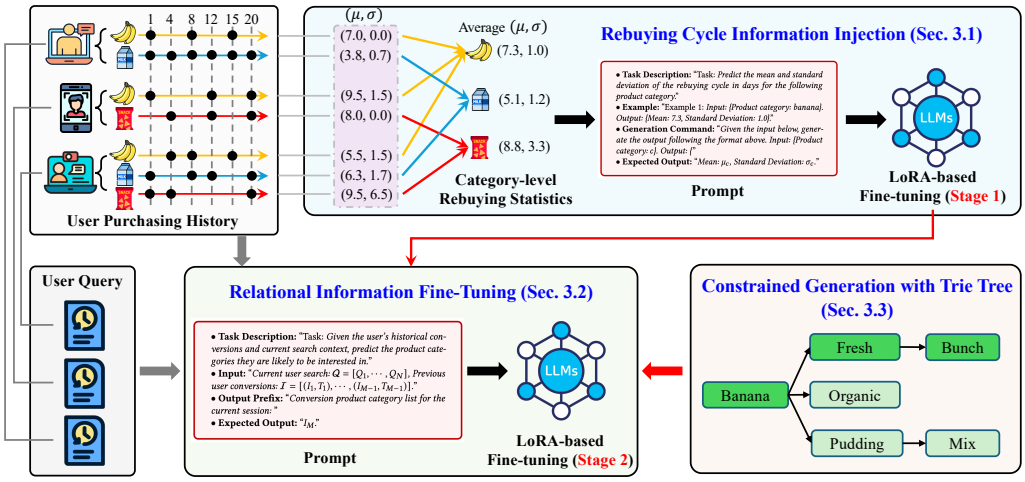

Figure 2: Overview of the proposed GROCLM

item relationships. The emergence of transformer-based models (Vaswani, 2017) and BERT (Devlin, 2018) further enhanced contextualized representation learning. Bi-encoder methods (Karpukhin et al., 2020; Qu et al., 2020; Xiong et al., 2020; Ni et al., 2021a,b; Reimers and Gurevych, 2019) and multi-representation models (Humeau et al., 2019; Luo et al., 2022; Khattab and Zaharia, 2020; Hofstätter et al., 2022) enabled efficient and fine-grained query-document matching. However, even dense retrieval and pre-trained embedding models (Guo et al., 2022; Fan et al., 2022; Yates et al., 2021) do not explicitly model domain-specific constraints such as cyclical repurchasing patterns and discrete category outputs in grocery recommendation.

3 Methodology

This work aims to predict accurate grocery category recommendations based on users' historical interactions and current queries.

Let $\mathcal{I} = [(I_1, T_1), \cdots, (I_{M-1}, T_{M-1})]$ denote a historical interaction sequence of length $M-1$, where each interaction I_m at time T_m is a subset of categories, i.e., $I_m \subset \mathcal{C}$ and $\mathcal{C}$ is the set of all grocery categories. We use $\mathcal{Q} = [Q_1, \cdots, Q_N]$ to denote the current N user queries.

Given $\mathcal{I}$, $\mathcal{Q}$, an LLM $\mathcal{F}$, and a set of prompt templates $\mathcal{P} = [P_1, \cdots, P_K]$, the goal is to predict the category set at time M, denoted as I_M. Since the task requires generating **discrete and mutually exclusive category labels**, we formulate it as autoregressive generation over category tokens:

$$p(I_M) = \sum_{c \in I_M} \prod_{l=1}^{L_c} p(w_l \mid \mathcal{F}, \mathcal{I}, \mathcal{Q}, \mathcal{P}, w_{<l}). \quad (1)$$

where a category $c = [w_1, \cdots, w_{L_c}]$ consists of L_c tokens, and $w_{<l} = [w_1, \cdots, w_{l-1}]$ (or $\emptyset$ if $l = 1$) denotes preceding tokens.

To address this formulation, we propose GROCLM, a generative retrieval framework based on prompt tuning and LLM fine-tuning (Figure 2). The model includes three components: (1) Rebuying Cycle Information Injection, (2) Relational Information Fine-Tuning, and (3) Constrained Generation with a Trie Tree. We detail each module below.

3.1 Rebuying Cycle Information Injection

A naive strategy is to fine-tune the LLM $\mathcal{F}$ directly on user interaction history $\mathcal{I}$ and queries $\mathcal{Q}$. However, this overlooks explicit modeling of cyclical purchasing behavior, which is fundamental in grocery data. These recurring patterns provide an unconditional temporal prior over category demand, enabling the model to make reasonable predictions even with limited interaction history. Therefore, we first inject category-level rebuying statistics before modeling more context-dependent relational patterns.

Category-level Rebuying Statistics Calculation. For each category $c \in \mathcal{C}$ and user $u \in \mathcal{U}$ with at least two conversions, we compute the time gaps between consecutive purchases. Let $\{T_1^{u,c}, \cdots, T_t^{u,c}\}$ denote the timestamps of purchases for category c. The time gaps are defined as $\Delta T_i^{u,c} = T_i^{u,c} - T_{i-1}^{u,c}$ for $i \geq 2$. The mean and standard deviation of these gaps for user u are denoted as $\mu_{u,c}$ and $\sigma_{u,c}$.

Aggregating across users yields the category-

level statistics:

$$\mu_c = \frac{1}{|\mathcal{U}|}\sum_{u=1}^{|\mathcal{U}|}\mu_{u,c}, \quad \sigma_c = \frac{1}{|\mathcal{U}|}\sum_{u=1}^{|\mathcal{U}|}\sigma_{u,c}, \tag{2}$$

where $|\mathcal{U}|$ is the number of users.

Prompt-based Rebuying Pattern Injection with LoRA. To inject the rebuying statistics $\{c, \mu_c, \sigma_c\}$ into the model $\mathcal{F}$, we adopt a prompt-based LoRA (Hu et al., 2021) fine-tuning approach. Specifically, we design the following prompt:

- **Task Description:** "Task: *Predict the mean and standard deviation of the rebuying cycle in days for the following product category.*"
- **Example:** "Example 1: *Input: {Product category: banana}. Output: {Mean: 7.3, Standard Deviation: 1.0}.*"
- **Generation Command:** "*Given the input below, generate the output following the format above. Input: {Product category: c}. Output: {*"
- **Expected Output:** "*Mean: μ_c, Standard Deviation: σ_c.*"

Note that we have $|\mathcal{C}|$ categories in the training set, and the corresponding $|\mathcal{C}|$ category-level statistics $\{c, \mu_c, \sigma_c\}$ ($\forall c \in \mathcal{C}$) are used to fine-tune $\mathcal{F}$ with LoRA.

3.2 Relational Information Fine-Tuning

Although Stage 1 captures high-level cyclical patterns, it does not model personalized preferences or session-level intent. Aggregated rebuying statistics overlook user-specific behaviors and in-session relationships among multiple queries. To address this, we introduce Relational Information Fine-Tuning as the second training stage. Similar to Stage 1, this phase adopts LoRA-based prompt tuning, but focuses on sequential interaction history and current queries to model context-aware user behavior.

Specifically, we treat the user's current search session as the **context**, and historical conversions as the **condition**. The context reflects short-term intent through queries (e.g., "fruit" or "dairy"), while the condition captures long-term purchasing patterns from historical interactions. Our prompt is structured as follows:

- **Task Description:** "Task: *Given the user's historical conversions and current search context, predict the product categories they are likely to be interested in.*"
- **Input:** "*Current user search: $\mathcal{Q} = [Q_1, \cdots, Q_N]$, Previous user conversions: $\mathcal{I} = [(I_1, T_1), \cdots, (I_{M-1}, T_{M-1})]$.*"
- **Output Prefix:** "*Conversion product category list for the current session: "*
- **Expected Output:** "I_M."

A straightforward approach is to apply standard autoregressive training by treating I_M as a sentence (Geng et al., 2022; Tan et al., 2024). However, this is sub-optimal for two reasons. First, categories in I_M are independent and mutually exclusive, differing from natural language tokens. Second, unconstrained generation requires post-hoc matching to valid categories, introducing errors. Even minor token deviations (e.g., "*snack*" vs. "*snacks*") lead to false negatives. Therefore, additional output constraints are necessary beyond LoRA fine-tuning alone.

3.3 Constrained Generation with Trie Tree

To address the discrete and mutually exclusive nature of category outputs, we adopt a trie-based constrained decoding strategy inspired by (De Cao et al., 2020). All valid product categories are tokenized and organized into a trie tree $\mathcal{T}$ (Figure 3). Each node corresponds to a token, and its children define valid subsequent tokens in a category sequence.

During decoding, the next token w_l is constrained to a valid set $\mathcal{V}_l$ defined by the trie. For example, given the prefix "*[Pork, Rib]*", the valid candidates are "*[Eye, Steak]*". This masking ensures that only valid category sequences can be generated. Let $\mathbf{x} = (\mathcal{F}, \mathcal{I}, \mathcal{Q}, \mathcal{P}, \mathcal{T})$ denote the full conditioning context. The constrained probability is defined as:

$$p(w_l) = \begin{cases} p(w_l \mid \mathbf{x}, w_{<l}) & \text{if } w_l \in \mathcal{V}_l, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

The training objective of GROCLM is therefore:

$$\min \mathcal{L} = -\sum_{c \in I_M}\sum_{l=1}^{L_c} \log p(w_l \mid \mathbf{x}, w_{<l}). \tag{4}$$

During inference, we apply beam search while enforcing trie-based constraints at each step. New

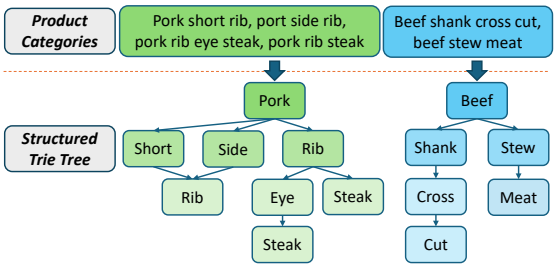

Figure 3: Illustration of trie tree.

categories can be incorporated by updating the trie without retraining the model.

4 Experiments

In this section, we compare our model's performance on various metrics, datasets, and prediction tasks.

4.1 Datasets

We evaluate GROCLM on a proprietary large-scale grocery dataset, referred to as the *Conversion* dataset. Statistics of the *Conversion* dataset are shown in Table 1.[1]

Conversion Dataset. The *Conversion* dataset is derived from a large online grocery e-commerce platform and contains user query and conversion histories. We extract search queries, timestamps, and associated product categories, retaining users with at least five conversions within a six-month window and at least ten total purchases. Each record represents a user's query–conversion sequence in JSON format. We also construct a Rebuying Statistics dataset by computing the mean and standard deviation of purchase time gaps per category. A category is included if at least five users have repeated purchases in that category[2].

Implementation Details.

We use LLAMA-3-8B-Instruct[3] as the backbone and fine-tune it with LoRA on a single NVIDIA A10G (24GB). Training is conducted in float16 with batch size 1 and gradient accumulation of 64. We set LoRA hyperparameters to rank $r = 64$, $\alpha = 16$, and dropout 0.1. Both fine-tuning stages share the same LoRA layers. To reduce memory usage, we apply 4-bit quantization.

[1]In addition to the proprietary *Conversion* dataset, we also conduct experiments on a public benchmark, the InstaCart Online Grocery Basket Analysis Dataset (ins, 2021) (https://www.kaggle.com/datasets/yasserh/instacart-online-grocery-basket-analysis-dataset), referred to as the *Aisle* dataset. Results on this dataset are reported in Appendix A.

[2]The sampled dataset used for experimentation is a subset of production data and does not reflect the full platform scale.

[3]https://huggingface.co/NousResearch/Meta-Llama-3-8B-Instruct

Conversion		Aisle	
Users	80,659	Users	206209
Product Categories	4,904	Aisles	134
Queries	1,563,868	Product Names	49,688
Searches	25,166,299	Orders	32,434,489
Conversion	15,175,567	Distinct Orders	32,14,874
Non-conversion	13,295,855	-	-

Table 1: Statistics of the *Conversion* and *Aisle* dataset.

4.2 Baseline Models

We compare GROCLM with traditional and LLM-based baselines, including GLOVE (Pennington et al., 2014), Cross Encoder (Reimers and Gurevych, 2019), Two Tower (Huang et al., 2013), SASRec (Kang and McAuley, 2018), LightGCN (He et al., 2020), ColBERT (Khattab and Zaharia, 2020), CausalBERT (Li et al., 2021), P5 (Geng et al., 2022), TIGER (Rajput et al., 2024), SEATER (Si et al., 2023), LLAMA3 (Dubey et al., 2024), LLAMA3-ID, RecICL (Bao et al., 2024), and A-LLMRec (Kim et al., 2024). Details are provided in Appendix D.

4.3 Product Category Recommendation with Context Query

Experiment Design and Metrics. We evaluate GROCLM on the **Conversion** dataset for category-level recommendation with contextual queries. Given a user's historical conversions and current session queries, the model predicts the top-k product categories likely to convert. Both predictions and ground truth are sets of categories (ranking is not considered). We report the 5-run mean and standard deviation of Precision@K, Recall@K, and F1@K in Table 2.

Results and Discussion. GROCLM consistently outperforms all baselines across metrics, demonstrating strong modeling of both cyclical purchasing behavior and context-driven intent. Traditional retrieval models (GloVe, Cross Encoder, Two-Tower, ColBERT, CausalBERT) achieve moderate recall but lower precision due to the absence of structured decoding. Graph- and sequence-based methods (LightGCN, SASRec) struggle to align query semantics with category prediction. Semantic-ID approaches (TIGER, SEATER) underperform, likely due to limited supervision for ID construction.

Among LLM-based baselines, P5 and LLAMA3 show competitive recall but lack task-specific con-

Metric\Model	Glove	Cross Encoder	Two Tower	SASRec	LightGCN	ColBERT	CausalBERT	P5	TIGER	SEATER	LLAMA3	LLAMA3-ID	RecICL	A-LLMRec	GROCLM
Precision@5	0.086	0.063	0.100	0.023	0.018	0.048	0.054	0.023	0.082	0.017	0.075	0.049	0.031	0.060	**0.142**
Recall@5	0.104	0.101	0.151	0.120	0.011	0.044	0.053	0.018	0.059	0.010	0.120	0.049	0.026	0.042	**0.186**
F1@5	0.079	0.067	0.099	0.037	0.010	0.037	0.043	0.015	0.060	0.012	0.076	0.042	0.024	0.041	**0.126**
Precision@10	0.060	0.048	0.077	0.018	0.018	0.033	0.041	0.019	0.078	0.020	0.047	0.039	0.020	0.036	**0.094**
Recall@10	0.139	0.161	0.182	0.190	0.009	0.061	0.073	0.029	0.101	0.026	0.148	0.068	0.033	0.049	**0.231**
F1@10	0.069	0.064	0.101	0.035	0.012	0.036	0.043	0.019	0.079	0.020	0.059	0.042	0.021	0.036	**0.111**
Precision@20	0.040	0.033	0.037	0.013	0.019	0.025	0.029	0.013	0.068	0.037	0.030	0.029	0.013	0.022	**0.069**
Recall@20	0.164	0.178	0.217	0.247	0.013	0.083	0.101	0.040	0.152	0.088	0.171	0.105	0.043	0.061	**0.284**
F1@20	0.056	0.047	0.056	0.025	0.007	0.033	0.040	0.017	0.086	0.050	0.048	0.040	0.018	0.028	**0.097**

Table 2: Evaluation on the *Conversion* dataset.

Metric	AS1	AS2	AS3	AS4	GROCLM
P@5	0.074	0.133	0.108	0.000	**0.138**
R@5	0.122	0.175	0.162	0.001	**0.187**
F1@5	0.076	0.126	0.108	0.000	**0.132**
P@10	0.046	0.087	0.067	0.000	**0.094**
R@10	0.152	0.206	0.185	0.000	**0.227**
F1@10	0.060	0.103	0.084	0.000	**0.112**
P@20	0.030	0.061	0.044	0.000	**0.069**
R@20	0.172	0.252	0.221	0.001	**0.272**
F1@20	0.046	0.088	0.068	0.000	**0.099**

Table 3: Ablation study results.

straints. For **RecICL**, we adapt the original binary prediction setup to generate category lists, which deviates from its intended design and may limit performance. Similarly, **A-LLMRec** relies on rich item textual features for user representation learning; due to limited descriptive information in our dataset, we simplify its encoder, potentially restricting its effectiveness. Despite these adjustments, GROCLM maintains a substantial margin over all baselines, highlighting the benefit of explicit rebuying modeling and constrained generation.

4.4 Ablation Study

In this section, we conduct an ablation study to assess the effectiveness of each proposed component of GROCLM on the **Conversion** dataset. The evaluation metrics remain consistent with previous experiments. The ablation includes the following: the vanilla LLAMA3 instruct model (AS1), GROCLM without Rebuying Cycle Information Injection (AS2), without Relational Information Fine-Tuning (AS3), and without Constrained Generation with Trie Tree (AS4). The experiment results are shown in Table 3.

The analysis demonstrates that each component significantly enhances GROCLM ' performance. Notably, AS4 shows that the trie tree and beam search are crucial for generating valid tokens from the predefined category set, reinforcing the importance of Constrained Output Generation. These results underscore the necessity of each fine-tuning step in achieving optimal performance in the grocery recommendation task.

4.5 Performance Comparison in a Production Environment

We further evaluate GROCLM in a real-world production setting on a restocking task, which predicts the product categories a user is likely to purchase in the following week. This task requires jointly modeling user queries, historical conversions, cyclical repurchasing patterns, and inter-category relationships under practical system constraints.

We compare GROCLM with the deployed production model. In this setup, candidate generation is restricted to each user's previously purchased categories, enforced through dynamically loaded user-specific trie trees during decoding. As shown in Figure 5, GROCLM achieves a 7.5% relative improvement in *cart-adds per impression* over the production model. In contrast, removing constrained generation and allowing unconstrained decoding leads to a 5.1% performance regression, mainly due to duplicated or hallucinated category outputs.

Although trie-based masking introduces minor computational overhead, real-time latency remains acceptable for large-scale commercial deployment. In practice, GROCLM generates all categories jointly in $\sim$0.3s, while traditional methods require $\sim$0.5s per category, highlighting a substantial efficiency advantage. To further optimize latency, we are developing an event-driven system that precomputes recommendations via Kafka streams, enabling near real-time responses (sub-second latency) while preserving production constraints.

4.6 Effect of Cycle Injection Strategy

To evaluate the role of Stage-1 cycle injection, we compare GROCLM with an in-context learning (ICL) variant where rebuying statistics are appended directly to the prompt in the form:

Rebuying stats: [<category, mean, std>].

Both approaches use identical inputs, differing only in whether cyclical information is encoded through parameter adaptation (LoRA) or provided

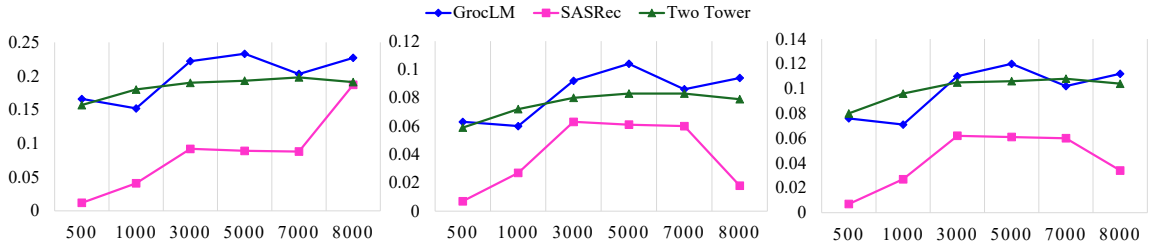

Figure 4: Data size sensitivity analysis. From Left to right: Recall@10, Precision@10, and F1@10.

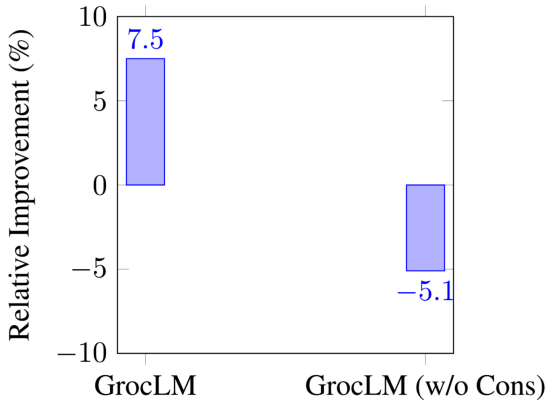

Figure 5: Relative improvement of GROCLM over the production model. (w/o Cons) refers to the variant without constrained generation.

Model	P@5	R@5	F1@5	P@10	R@10	F1@10	P@20	R@20	F1@20
ICL Only	0.010	0.008	0.008	0.006	0.012	0.007	0.004	0.016	0.006
GROCLM	**0.142**	**0.186**	**0.126**	**0.094**	**0.231**	**0.111**	**0.069**	**0.284**	**0.097**

Table 4: Comparison between in-context learning (ICL) and LoRA-based cycle injection.

via prompts. Compared to LoRA-based parameterization, the ICL formulation leads to longer prompts and less effective utilization of cyclical signals.

As shown in Table 4, the ICL-only variant performs substantially worse across all metrics. This demonstrates that directly providing statistical signals via prompts is insufficient, while LoRA-based parameter adaptation enables the model to internalize cyclical purchasing patterns more effectively. This also explains the robustness of GROCLM to prompt variations, as key information is encoded in model parameters rather than relying on prompt design.

5 Data Size Sensitivity Analysis

In this section, we perform a sensitivity analysis on the training data size to evaluate the impact of varying dataset sizes on the performance of different models with the **Conversion** dataset. Specifically, we analyze the models' performance in terms of Recall@10, Precision@10, and F1@10 across dataset sizes of 500, 1,000, 3,000, 5,000, 7,000, and 8,000 users from the original training dataset. We also fixed the testing data size to match the original testing dataset. We chose Two Tower and SASRec as baselines since they are widely used in the industry. We present the results in Figure 4.

As shown in the Figure, our model outperforms others as the dataset size increases, particularly after the dataset exceeds 3000 users. For smaller datasets (500–1,000 users), Two Tower shows marginally better results in precision and recall, likely due to its simpler architecture that performs well with limited data. However, as dataset sizes grow, both SASRec and Two Tower plateau around the 5,000-user mark. For SASRec, although we see an uptrend in recall with the increase of users, its precision drops significantly. In contrast, our model continues to improve, especially in larger datasets of 7,000 and 8,000 users, demonstrating its ability to capture more relevant items and refine predictions as more data becomes available. This trend is particularly evident in the F1@10 results, where our model maintains a better balance between recall and precision, outperforming the others as the dataset size increases. While Two Tower shows steady performance, it fails to achieve the same level of improvement as our model when the dataset grows.

6 Conclusion

We propose GROCLM, a generative language model for grocery category recommendation. By modeling user behavior at the category level and conditioning on historical purchases and queries, GROCLM captures cyclical patterns, diverse intent, and constrained outputs. Experiments show that GROCLM outperforms strong baselines, demonstrating effectiveness for real-world systems. Future work will explore low-resource personalization and broader category control.

7 Ethical Considerations

This work is based on a proprietary large-scale grocery dataset derived from real-world user interactions. All data used in this study were anonymized and aggregated in accordance with internal data governance policies. No personally identifiable information was accessed or utilized during model development or evaluation. The system operates at the product category level and does not infer or generate sensitive personal attributes.

As a deployed recommendation system, GROCLM may influence product exposure and customer purchasing decisions. To mitigate potential risks, we incorporate constrained generation mechanisms that restrict outputs to a predefined category vocabulary, reducing the risk of unsafe or inappropriate predictions. We also conduct offline evaluation and controlled production testing prior to deployment. Continuous monitoring is performed to detect performance degradation, unintended biases, or distribution shifts. Human oversight remains part of the deployment workflow to ensure system reliability and alignment with business and user experience goals.

8 Broader Impact

Category-level recommendation systems can improve user experience by reducing search friction and helping customers efficiently discover relevant products. For large-scale online grocery platforms, improved recommendation accuracy may also enhance operational efficiency and reduce cognitive load for users navigating extensive catalogs.

However, recommendation systems may also amplify existing popularity biases or disproportionately favor certain product groups. While GROCLM does not model sensitive demographic attributes, systematic biases in historical purchasing data may influence predictions. Future work includes more formal bias auditing and fairness-aware optimization strategies to ensure balanced exposure across product categories. By emphasizing constrained generation and deployment safeguards, we aim to promote responsible and trustworthy application of large language models in real-world commerce environments.

References

2021. Instacart online grocery basket analysis dataset.

Akiko Aizawa. 2003. An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1):45–65.

Keqin Bao, Ming Yan, Yang Zhang, Jizhi Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2024. Real-time personalization for llm-based recommendation with customized in-context learning. *arXiv preprint arXiv:2410.23136*.

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2023. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1007–1014.

Jin Chen, Zheng Liu, Xu Huang, Chenwang Wu, Qi Liu, Gangwei Jiang, Yuanhao Pu, Yuxuan Lei, Xiaolong Chen, Xingmei Wang, and 1 others. 2024. When large language models meet personalization: Perspectives of challenges and opportunities. *World Wide Web*, 27(4):42.

Zeyu Cui, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. 2022. M6-rec: Generative pretrained language models are open-ended recommender systems. *arXiv preprint arXiv:2205.08084*.

Sunhao Dai, Ninglu Shao, Haiyuan Zhao, Weijie Yu, Zihua Si, Chen Xu, Zhongxiang Sun, Xiao Zhang, and Jun Xu. 2023. Uncovering chatgpt's capabilities in recommender systems. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 1126–1132.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. *arXiv preprint arXiv:2010.00904*.

Yashar Deldjoo, Zhankui He, Julian McAuley, Anton Korikov, Scott Sanner, Arnau Ramisa, René Vidal, Maheswaran Sathiamoorthy, Atoosa Kasirzadeh, and Silvia Milano. 2024. A review of modern recommender systems using generative models (gen-recsys). In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6448–6458.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yixing Fan, Xiaohui Xie, Yinqiong Cai, Jia Chen, Xinyu Ma, Xiangsheng Li, Ruqing Zhang, Jiafeng Guo, and 1 others. 2022. Pre-training methods in information retrieval. *Foundations and Trends® in Information Retrieval*, 16(3):178–317.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 299–315.

Jiafeng Guo, Yinqiong Cai, Yixing Fan, Fei Sun, Ruqing Zhang, and Xueqi Cheng. 2022. Semantic models for the first-stage retrieval: A comprehensive review. *ACM Transactions on Information Systems (TOIS)*, 40(4):1–42.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 55–64.

Jiafeng Guo, Yixing Fan, Xiang Ji, and Xueqi Cheng. 2019. Matchzoo: A learning, practicing, and developing system for neural text matching. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1297–1300.

Deepesh V Hada and Shirish K Shevade. 2021. Rexplug: Explainable recommendation using plug-and-play language model. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 81–91.

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.

Sebastian Hofstätter, Omar Khattab, Sophia Althammer, Mete Sertkan, and Allan Hanbury. 2022. Introducing neural bag of whole-words with colberter: Contextualized late interactions using enhanced reduction. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 737–747.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Wenyue Hua, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. 2023. How to index item ids for recommendation foundation models. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pages 195–204.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring. *arXiv preprint arXiv:1905.01969*.

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

Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed Chi, and Derek Zhiyuan Cheng. 2023. Do llms understand user preferences? evaluating llms on user rating prediction. *arXiv preprint arXiv:2305.06474*.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Sein Kim, Hongseok Kang, Seungyoon Choi, Donghyun Kim, Minchul Yang, and Chanyoung Park. 2024. Large language models meet collaborative filtering: An efficient all-round llm-based recommender system. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1395–1406.

Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. 2022. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11523–11532.

Lei Li, Yongfeng Zhang, and Li Chen. 2020. Generate neural template explanations for recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 755–764.

Lei Li, Yongfeng Zhang, and Li Chen. 2023. Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems*, 41(4):1–26.

Zhongyang Li, Xiao Ding, Kuo Liao, Bing Qin, and Ting Liu. 2021. Causalbert: Injecting causal knowledge into pre-trained models with minimal supervision. *arXiv preprint arXiv:2107.09852*.

Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. 2023. Is chatgpt a good recommender? a preliminary study. *arXiv preprint arXiv:2304.10149*.

Tie-Yan Liu and 1 others. 2009. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331.

Man Luo, Arindam Mitra, Tejas Gokhale, and Chitta Baral. 2022. Improving biomedical information retrieval with neural retrievers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11038–11046.

Zhiming Mao, Huimin Wang, Yiming Du, and Kam-Fai Wong. 2023. Unitrec: A unified text-to-text transformer and joint contrastive learning framework for text-based recommendation. *arXiv preprint arXiv:2305.15756*.

Bhaskar Mitra and Nick Craswell. 2017. Neural models for information retrieval. *arXiv preprint arXiv:1705.01509*.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith B Hall, Daniel Cer, and Yinfei Yang. 2021a. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv preprint arXiv:2108.08877*.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 188–197.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y Zhao, Yi Luan, Keith B Hall, Ming-Wei Chang, and 1 others. 2021b. Large dual encoders are generalizable retrievers. *arXiv preprint arXiv:2112.07899*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2020. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2010.08191*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, and 1 others. 2024. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36.

Juan Ramos and 1 others. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Stephen Robertson. 2004. Understanding inverse document frequency: on theoretical arguments for idf. *Journal of documentation*, 60(5):503–520.

Gerard Salton. 1962. Some experiments in the generation of word and document associations. In *Proceedings of the December 4-6, 1962, fall joint computer conference*, pages 234–250.

Gerard Salton, Anita Wong, and Chung-Shu Yang. 1975. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Scott Sanner, Krisztian Balog, Filip Radlinski, Ben Wedin, and Lucas Dixon. 2023. Large language models are competitive near cold-start recommenders for language-and item-based preferences. In *Proceedings of the 17th ACM conference on recommender systems*, pages 890–896.

Zihua Si, Zhongxiang Sun, Jiale Chen, Guozhang Chen, Xiaoxue Zang, Kai Zheng, Yang Song, Xiao Zhang, and Jun Xu. 2023. Generative retrieval with semantic tree-structured item identifiers via contrastive learning. *arXiv preprint arXiv:2309.13375*.

Damien Sileo, Wout Vossen, and Robbe Raymaekers. 2022. Zero-shot recommendation as language modeling. In *European Conference on Information Retrieval*, pages 223–230. Springer.

Juntao Tan, Shuyuan Xu, Wenyue Hua, Yingqiang Ge, Zelong Li, and Yongfeng Zhang. 2024. Towards llm-recsys alignment with textual id learning. *arXiv preprint arXiv:2403.19021*.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Yidan Wang, Zhaochun Ren, Weiwei Sun, Jiyuan Yang, Zhixiang Liang, Xin Chen, Ruobing Xie, Su Yan, Xu Zhang, Pengjie Ren, and 1 others. 2024. Enhanced generative recommendation via content and collaboration integration. *arXiv preprint arXiv:2403.18480*.

Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021. Empowering news recommendation with pre-trained language models. In *Proceedings of the 44th international ACM SIGIR conference on*

research and development in information retrieval, pages 1652–1656.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, and 1 others. 2024. A survey on large language models for recommendation. *World Wide Web*, 27(5):60.

Shitao Xiao, Zheng Liu, Yingxia Shao, Tao Di, Bhuvan Middha, Fangzhao Wu, and Xing Xie. 2022. Training large-scale news recommenders with pretrained language models in the loop. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4215–4225.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808*.

Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. 2021. Pretrained transformers for text ranking: Bert and beyond. In *Proceedings of the 14th ACM International Conference on web search and data mining*, pages 1154–1156.

Jizhi Zhang, Keqin Bao, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2023. Is chatgpt fair for recommendation? evaluating fairness in large language model recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 993–999.

Zihuai Zhao, Wenqi Fan, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Zhen Wen, Fei Wang, Xiangyu Zhao, Jiliang Tang, and 1 others. 2024. Recommender systems in the era of large language models (llms). *IEEE Transactions on Knowledge and Data Engineering*.

Bowen Zheng, Yupeng Hou, Hongyu Lu, Yu Chen, Wayne Xin Zhao, Ming Chen, and Ji-Rong Wen. 2024. Adapting large language models by integrating collaborative semantics for recommendation. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 1435–1448. IEEE.

Justin Zobel and Alistair Moffat. 2006. Inverted files for text search engines. *ACM computing surveys (CSUR)*, 38(2):6–es.

Justin Zobel, Alistair Moffat, and Kotagiri Ramamohanarao. 1998. Inverted files versus signature files for text indexing. *ACM Transactions on Database Systems (TODS)*, 23(4):453–490.

A Aisle Recommendation with In-basket Items

Experiment Design and Evaluation Metrics. In this task, we use the **Aisle** dataset to simulate a common grocery scenario where a user has already added items to their basket during a shopping session. The objective is to predict additional aisles the user is likely to visit based on the items already in the basket. In this context, product names are used to predict aisles, which is analogous to predicting categories from queries in the previous tasks. This setup allows us to assess the model's ability to leverage both direct relationships between in-basket items and broader historical patterns to make accurate aisle predictions.

To simulate the in-basket scenario, we randomly sample 50 percent of the products from each user's basket as input to the model and use the corresponding aisles of the remaining products as the ground truth. Historical aisle information is also provided, similar to the previous task. Given *the smaller output space* of the **Aisle** dataset, we evaluate the model's performance with 5-run using the mean and standard deviations of Precision@K, Recall@K, and F1@K with K set at 5 and 10.

Experiment Result Analysis and Discussion. Table 5 shows that GROCLM consistently outperforms all baselines in precision, recall, and F1, confirming its strength in modeling relationships between in-basket items and their corresponding aisles. While Two Tower and LLAMA3 perform relatively well, they still trail behind GROCLM, highlighting the benefit of incorporating structured decoding and historical context. ColBERT and CausalBERT achieve competitive results but are limited by their reliance on shallow token-level similarity. LightGCN also performs poorly, facing the same limitations observed in the previous experiment. P5 underperforms significantly, struggling to map in-basket items to aisles in this constrained setting.

B Case Study

User Buying History. To evaluate the effectiveness of our model in learning from historical purchasing patterns and predicting product categories accurately, we conducted a detailed case study, illustrated in Figure 6. The figure presents the anonymized buying history of a single user, that spans 8 times in total. The first 7 times, denoted as t_1 to t_7, represent the user's historical purchases, while the 8th serves as the ground truth and the prediction target. The vertical axis represents the product categories purchased over time. We applied the same background color to product categories that are alike to highlight related buying patterns.

Metric\Model	Glove	Cross Encoder	Two Tower	SASRec	LightGCN	ColBERT	CausalBERT	P5	TIGER	SEATER	LLAMA3	LLAMA3-ID	GROCLM
Precision@5	0.046	0.048	0.111	0.091	0.007	0.086	0.125	0.002	0.021	0.051	**0.137**	0.000	0.128
Recall@5	0.048	0.051	0.132	0.128	0.001	0.113	0.142	0.002	0.010	0.024	0.133	0.001	**0.152**
F1@5	0.043	0.045	0.108	0.093	0.001	0.080	0.107	0.002	0.013	0.032	0.127	0.001	**0.128**
Precision@10	0.051	0.045	0.085	0.065	0.017	0.092	0.103	0.003	0.024	0.055	0.085	0.055	**0.118**
Recall@10	0.104	0.093	0.194	0.170	0.011	0.216	0.225	0.007	0.022	0.052	0.184	0.124	**0.231**
F1@10	0.064	0.056	0.109	0.065	0.012	0.108	0.135	0.004	0.023	0.053	0.112	0.070	**0.147**

Table 5: Evaluation on the *Aisle* dataset.

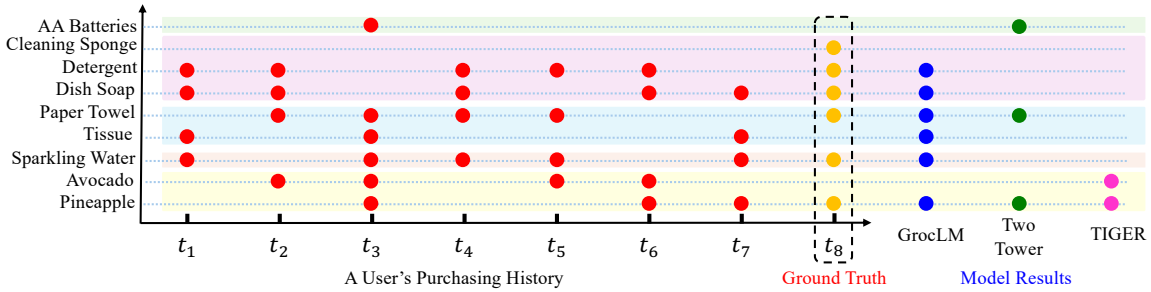

Figure 6: Illustration of the case study with a user's buying history.

In addition to showcasing the predictions made by GROCLM, we compare the predictions of GROCLM against those of the Two Tower model, a widely-used baseline in industry, and TIGER, the recent semantic ID-based model. This comparison helps to assess each model's ability to capture and predict the user's purchasing behavior.

Our case study reveals that certain product categories exhibit consistent purchasing patterns several times, while others are more sporadic. GROCLM demonstrates its ability to accurately capture both types of patterns. For example, it correctly predicted the purchase of "*Sparkling Water*," following a recurring trend of being bought three times with a one-time gap. Additionally, GROCLM successfully captured the co-occurrence of similar categories, such as "*Detergent*" and "*Dish Soap*," which also appear regularly in the user's buying history. In contrast, the Two Tower model misaligned, predicting "*AA Batteries*," a product that is uncommon in the user's purchasing habits. Similarly, TIGER falsely predicted "*Avocado*," a product frequently purchased by the user, but failed to capture its true skipping pattern. This discrepancy underscores GROCLM's strength in leveraging temporal data to accurately identify recurring categories. Furthermore, we observed that "*Cleaning Sponge*" appears in the ground truth, but none of the models—including GROCLM —predicted it. This highlights the challenge of forecasting irregular or one-off purchases, which remains an area for improvement across all models. Despite these occasional missed predictions, GROCLM consistently shows a stronger grasp of recurring product relationships compared to the other models.

Metric\Model	AS1	AS2	AS3	AS4	GROCLM
Acc: both	0.003	0.128	0.085	0.000	**0.144**
Acc: board	0.002	0.115	0.080	0.000	**0.132**
Acc: narrow	0.009	0.224	0.131	0.000	**0.252**

Table 6: Pairwise evaluation.

Rebuying Cycle. In this case study, we compare the vanilla LLAMA3-8b model with our Rebuying-Cycle–injected model to evaluate the effectiveness of the restocking mechanism. Both models were tested using the same prompting and evaluation strategy as the full model, with their predicted categories presented in Figure 7. The injected model clearly demonstrates its ability to leverage the user's historical purchasing trends, resulting in more targeted and personalized predictions. For example, it successfully identifies recurring items like sausage-and-egg breakfast sandwiches (highlighted in gray). Additionally, the injected model responds effectively to the query, generating relevant recommendations (highlighted in blue). In contrast, the LLAMA-8b model produces a broader set of predictions that, while occasionally aligning with the user's categories, often lack the precision needed to reflect true historical favorite, and struggles to generate categories that belongs to the predefined category set.

In summary, the injection stage successfully inject the rebuying cycle into the model. However, the injection model does exhibit some overfitting tendencies, such as returning multiple closely related items rather than offering a more diverse set of suggestions, indicating future improvement directions.

Figure 7: Illustration of the case study of the rebuying cycle.

C Pairwise Evaluation of Query Intent

Testing how the model handles broad and narrow queries is crucial for assessing its ability to generalize across varying levels of specificity, which is essential for real-world applications where user queries can range from highly general to very specific. In this experiment, we evaluate how GROCLM responds to broad queries like "fruits" and "meats" versus narrow queries such as "Coca-Cola." Intuitively, narrow queries should be easier to handle due to their specificity, while broad queries are more challenging due to their vagueness.

For this task, we use the **Conversion** dataset and first classify each user query intent as broad or narrow using the LLAMA3 8B Instruction model as a data pre-processing step. We input the product category and constrain the output to be either broad or narrow. We also exclude queries that did not result in a conversion. Each query is used to predict the paired conversion product category, generating one output candidate per query. Accuracy is used as the evaluation metric to measure the model's ability to correctly predict the intended category.

This experiment is conducted in three parts: evaluating broad queries only, narrow queries only, and a combination of both. We experiment on all model variants from the ablation study. The results are presented in Table 6. The results align with our expectation that narrow queries are easier for the model to handle, resulting in more accurate predictions. The vanilla LLAMA3 model (AS1) performs poorly, likely due to its lack of fine-tuning on the subset of product categories and limited understanding of their relationships. As the level of fine-tuning increases, the model's performance improves, with the best results achieved by the full GROCLM. This demonstrates the effectiveness of the additional tuning steps in enhancing the model's ability to process different query intents accurately.

D Baseline Models

The details of the chosen baseline models are:

- **GLOVE** (Pennington et al., 2014) is an unsupervised learning algorithm for obtaining vector representations of words. GLOVE captures semantic relationships between words by training on aggregated global word-word co-occurrence statistics from a corpus.
- **Cross Encoder** (Reimers and Gurevych, 2019) processes pairs of input texts together and generates a single embedding for the combined input, allowing detailed interaction between the inputs.
- **Two Tower** (Huang et al., 2013) independently encodes the query and candidate items with separate encoders and compares the generated embeddings using a similarity measure. It is efficient for large-scale retrieval tasks as it allows for the pre-computation and indexing of candidate item embeddings.
- **LightGCN** (He et al., 2020) is a graph-based collaborative filtering model that simplifies traditional GCNs by removing feature transformation and nonlinear activation, focusing purely on neighborhood aggregation over a user-item interaction graph.
- **ColBERT** (Khattab and Zaharia, 2020) is a late interaction retrieval model that encodes queries and documents into contextualized token-level embeddings and performs similarity computation via maximum-similarity matching across tokens, allowing efficient and accurate retrieval.

- **CausalBERT** (Li et al., 2021) integrates causal inference with language modeling by estimating treatment effects from observational data. In retrieval, it leverages token-level representations with causality-aware modeling to improve recommendation accuracy.

- **SASRec** (Kang and McAuley, 2018) uses self-attention mechanisms to model user behavior sequences, capturing temporal dynamics and contextual dependencies.

- **P5** (Geng et al., 2022) is a unified model that uses a text-to-text paradigm for various recommendation tasks. By converting all data into natural language sequences, P5 leverages deep semantics for personalized recommendations, enabling zero-shot and few-shot predictions across different tasks.

- **TIGER** (Rajput et al., 2024) encodes item-related information with a Residual-Quantized Variational Autoencoder (RQ-VAE) (Lee et al., 2022) and generates a corresponding semantic ID. With the semantic ID replacing the original items, it utilizes a variant of T5 (Raffel et al., 2020) to provide sequential recommendations.

- **SEATER** (Si et al., 2023) not only constructs semantic IDs for each item, but also further distinguishes items with a distinguishing loss and a ranking loss. It also utilizes a variant of the T5 (Raffel et al., 2020) for recommendations.

- **LLAMA3** (Dubey et al., 2024) is a large language model that leverages extensive pre-training on diverse datasets to perform a wide range of generative tasks, demonstrating the capabilities of state-of-the-art language models in understanding and generating coherent text.

- **LLAMA3-ID** follows a similar training and evaluation procedure as GROCLM, but instead of generating natural text, this model directly outputs the IDs of the items.

- **RecICL** (Bao et al., 2024) is an in-context learning framework for recommendation that performs prediction by constructing task-specific prompts from historical interactions.

- **A-LLMRec** (Kim et al., 2024) combines encoder-based user representation learning with LLM prompting to perform history-to-item recommendation.

E Cold-Start and Sparse User Behavior

• **Task Description:** "*Task: Recommend product categories given a new user's search query.*"
• **Prompt Structure:** "*The following are examples for input and output structure:*"
Example 1: Input: {snacks}. Output: {Vegetable Chips, Crackers, Snack Packs, Seaweed Chips, Packaged Cookies}.
Example 2: Input: {fruit}. Output: {Strawberries, Blackberries, Watermelons, Green Grapes, Bananas}.
• **Generation Instruction:** "*Given the input below, generate output following the format above.*"
• **Query Template:** Input: {<*user query*>}. Output:{

To evaluate GrocLM's performance in cold-start scenarios, we conducted a controlled experiment using a template-style prompt that includes only the user's current search query without any prior conversion history. This setup simulates a realistic case where a new or infrequent user initiates a session with no behavioral history. Additionally, to simulate a complete cold-start condition, we also test the model with an empty query input, removing all contextual signals. The results of these experiments are shown in Table 7. We observe that even in the absence of any user history, GrocLM is able to interpret free-form keyword queries and generate relevant product categories from the predefined vocabulary set, demonstrating its ability to generalize from language cues. However, when both history and query input are absent, the model produces a static output:*Baking Supplies*, likely reflecting common patterns seen during training. This highlights a key limitation in the current framework and motivates future work to enhance cold-start robustness. Specifically, we plan to incorporate auxiliary user-related signals such as demographics, time-of-day, session metadata, or globally trending items to better personalize recommendations in zero-input scenarios and improve model flexibility under sparse conditions.

User Query	Predicted Product Categories
cookie baking chocolate chips flour	Chocolate Chips, Flour
movie night chips soda	Potato Chips, Popcorn, Soft Drinks
breakfast eggs bread milk	Eggs, Bread, Milk
kitchen cleaner	All-Purpose Cleaner, Disinfectant Spray, Scrub Sponges
pretzel snack	Pretzel Bites, Pretzel Rods, Pretzel Sticks, Pretzel Rings
<No User Query>	Baking Supplies

Table 7: Cold-start predictions from GrocLM using only user query keywords.

F Discussion on Generalization to Other Domains

While GROCLM is specifically designed for grocery category recommendation, its architectural components are generalizable to other e-commerce domains that exhibit structured taxonomies and sequential purchase behaviors. In particular:

- The two-stage prompt tuning strategy can be adapted using alternative domain-specific priors (e.g., seasonal trends in fashion or replenishment cycles in pet supplies).
- The trie-constrained decoding mechanism is applicable wherever a fixed, valid label space exists.
- The use of precomputed rebuying statistics or similar global signals can help bootstrap recommendations in sparse settings across domains.

We leave domain transfer experiments to future work, but we believe the modeling framework presented in this paper lays a foundation for extensibility to other structured recommendation tasks.

G Error Analysis through User Example

Based on case study results in Figure 6 and Figure 7, we identify two representative error types exhibited by our model. First, GrocLM occasionally struggles to capture implicit product relationships. For example, it successfully predicts Detergent and Dish Soap but overlooks the complementary Cleaning Sponge, revealing a limitation in its in-context relational reasoning. Second, the model may overfit to frequent patterns learned from rebuying-cycle supervision, leading to redundant or overly similar predictions—such as generating multiple variants of Pretzels—which reduces overall output diversity. These observations highlight opportunities to enhance relational reasoning and improve the balance between consistency and coverage in future model iterations.