Personalized Next-Basket Recommendation with Interpretable Cycle-Aware Purchase Modeling

Fatemeh Fazaeefar[1][0009−0003−8431−8357], Paolo Speziali[1][0000−0002−7418−0396], Szymon Bobek[2][0000−0002−6350−8405], Grzegorz J. Nalepa[2][0000−0002−8182−4225], Diederik M. Roijers[1][0000−0002−2825−2491], Ann Nowé[1][0000−0001−6346−4564], and Mehrdad Asadi[1][0000−0003−3360−9579]

[1] AI Lab, Vrije Universiteit Brussel (VUB)
[2] Department of Human-Centered Artificial Intelligence, Institute of Applied Computer Science, Jagiellonian University, Krakow, Poland

Abstract. Standard collaborative filtering is efficient for large retail datasets but overlooks the temporal dynamics of customer behavior, while neural and complex models capture these dynamics at the cost of heavy computational demands. We propose a hybrid recommendation method that retains the efficiency of classical approaches while incorporating lightweight temporal modeling to capture implicit feedback and personalized cyclic purchasing patterns. The experiments verify that the proposed method achieves comparable performance to state-of-the-art methods, maintaining a linear computational overhead, and provides an interpretable temporal feature. In particular, our method consistently outperforms the frequency-based baseline across all metrics, achieving relative improvements ranging from 1.9% to 3.0%. Applied to a real-world retail use case with large-scale transactional data, the method demonstrates its practicality and effectiveness for personalized product recommendations in physical retail stores.

Keywords: Personalization Recommendation· Temporal Modeling

1 Introduction

Personalized recommendations have become a core capability in digital commerce, where user ratings, reviews, and other forms of explicit feedback enable systems to infer preferences with high accuracy. In this paper, we focus on the physical grocery retail domain and supermarkets in which explicit evaluative signals are rarely available. Customers typically make purchasing decisions without providing direct feedback on products, which limits the applicability of traditional collaborative filtering and many modern neural recommendation approaches that depend on explicit user–item interactions. At the same time, physical retail generates rich volumes of implicit customers' transactional data, including sets of items purchased together and regular purchasing patterns.

Despite the abundance of these signals, implicit feedback presents two main challenges. First, user preferences have to be inferred from item-purchase counts

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

instead of (partial) direct ratings. Second, there is a large amount of missing interaction data to deal with. While methods such as Matrix Factorization (MF) effectively capture user–item interaction patterns and can leverage implicit feedback such as purchase counts, they ignore temporal structure and therefore cannot predict *when* an item is likely to be needed again. While neural nets [2] and Transformer-based models have been successfully applied to Next Item Recommendation (NIR) due to their ability to model sequential user behavior, their computational complexity poses significant challenges in Next Basket Recommendation (NBR) due to highly repetitive/sparse interactions [7]. Long basket sequences, large item catalogs, and the need to predict sets of items exacerbate the training and inference costs of such architectures, motivating the development of more efficient and scalable methods for NBR.

To fill this gap, we propose a hybrid model that combines MF-based preference modeling with a lightweight, interpretable temporal component derived from cyclic analysis of user–item purchase behavior, enabling personalized and time-aware basket recommendation. Our method jointly models basket-level interactions and temporal shopping patterns to more accurately reflect how customers usually shop.

In doing so, it supports recommendation decisions based on personal purchasing habits, aligning with real consumer behavior. The approach is transparent, computationally efficient, and compatible with state-of-the-art implicit-feedback, while remaining simple enough for deployment in real-world retail systems.[3]

Our contributions are threefold. First, we extend the collaborative filtering methods by introducing a method to capture cyclical shopping patterns at scale, enabling traditional retail recommendation systems to consider not only ***what*** to recommend but also ***when***. Second, our model focuses on both personalized user interactions and discoverability while preserving interpretability. Third, we demonstrate that the resulting system achieves competitive results with modern complex recommendation baselines on publicly available datasets and demonstrate its applicability on a large real-world retail dataset as a use case. To support reproducibility, we have publicly released the code on GitHub[4].

2 Related Work

Next Basket Recommendation (NBR) refers to identifying next purchases based on the user's prior basket information. Earlier influential work demonstrated the benefits of focusing on item-to-item relationships in large, sparse, e-commerce environments. Most of the methods use the score-and-rank approach: they first compute scores for all items in catalogue and then return the top-k highest-ranked items as recommendations. Methods such as Item-Based Collaborative

[3] Our work is carried out in industrial collaboration with MC Sonae, a leading company in the food retail sector within the scope of the PEER project. The company we work with provided the case study with real anonymized customer data.

[4] https://github.com/Fazaeefar/HybridCyclicBPR

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

Filtering (ICF) pioneered the strategy of analyzing the user-item matrix to identify the relationships between items [11]. The authors showed that pre-computing an item-item similarity matrix and recommending based on a user's ratings of similar items offered dramatically better scalability and comparable, often superior, prediction quality compared to traditional user-based (e.g., distance-based) methods. This emphasis on item-item relationships remains central to the NBR problem.

Early recommender systems were dominated by Matrix Factorization (MF) models, which became the standard approach after their success in the Netflix Prize and were later formalized in [6]. MF approximates a high-dimensional input matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$ by decomposing it into the product of two lower-rank matrices, expressed as:

$$\mathbf{R} \approx \mathbf{U}\mathbf{V}^T,$$

where $\mathbf{U} \in \mathbb{R}^{m \times k}$ and $\mathbf{V} \in \mathbb{R}^{n \times k}$, represent the preferences of users to a small set of k latent features and values of the items in these k features, respectively. The effect of making this decomposition, when k is sufficiently low, is that $\hat{\mathbf{R}} = \mathbf{U}\mathbf{V}^T$ now becomes a recommendation matrix with recommendation scores for each user-item pair. The recommendation score for a user u and an item i then becomes a linear product, $\hat{r}_{ui} = \boldsymbol{p}_u \cdot \boldsymbol{q}_i$, with $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$ being the appropriate vectors from the U and V matrices. While this approximation process reduces dimensionality and preserves the underlying structure of the data, MF can only capture relatively simple preferences due to the linear product structure. Furthermore, user–item relationships that have a temporal component cannot be captured by this method.

Another key limitation is that the basic MF model does not take into account which items a user has seen already. This motivated research on learning objectives tailored to implicit data [5]. A major step in this direction was Bayesian Personalized Ranking (BPR) [10], which introduced a pairwise ranking loss that focused on correctly ordering observed versus unobserved interactions. Importantly, BPR did not alter MF's architecture; instead, it provided a more appropriate training paradigm for implicit feedback It assumes that a user prefers observed items over unobserved ones. This method substantially improved Top-k recommendation accuracy but still retained MF's fundamental linearity and temporal limitations.

To overcome the representational constraints of traditional MF and BPR-MF, subsequent work shifted toward nonlinear models, drawing heavily from neural networks. Early attempts such as RBM-based collaborative filtering [9] and AutoRec [12] demonstrated the effectiveness of learning higher-order latent structure using autoencoders. This progression culminated in the broader paradigm of deep learning-based recommender systems, where models explicitly replace the MF inner product with expressive nonlinear functions. A prominent example is Neural Collaborative Filtering (NCF) and its NeuMF architecture [4], which use multilayer perceptrons to model user–item interactions as flexible, learnable mappings. These neural models [3] provide superior capacity to capture intricate behavioral patterns, integrate heterogeneous signals, and model

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

user preferences beyond what linear MF permits [13]. While these models are highly effective for partially explicit feedback data, they are less effective for the physical grocery retail domain. Furthermore, these models are not very interpretable. In this paper, we therefore focus on creating a simpler interpretable method tailored to the grocery retail domain with its specific properties.

3 Cyclic-Aware BPR Method

This section describes our proposed method for grocery retail recommendation (Figure 1). The core idea for this method comes from the observation that many

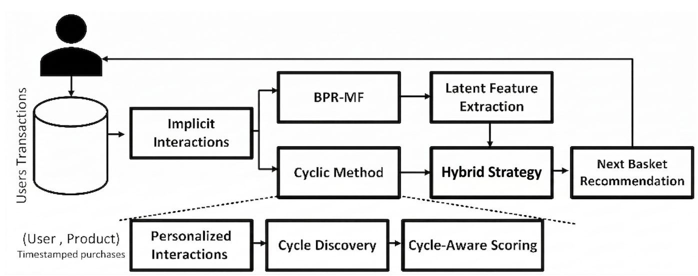

Fig. 1. The proposed light-weight next basket recommendation pipeline

people buy their grocery items in a repeated pattern over time [1]. For example, people buy fruit and dairy on a weekly basis and buy hygiene products on a monthly basis. If we utilize these patterns, we can recommend products that we are more confident the user might buy. Our proposed model combines such cyclic-aware information with classical Matrix Factorization (MF)-based techniques to model both habits and exploration for NBR. The cyclic-aware recommendation pipeline consists of two main components (Algorithm 1).

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

Algorithm 1: Cyclic Behavior-Based Recommendation

Input: Transactions $\mathcal{T} = \{(u, i)\}$, target user u, target date t_{target}, parameters λ, ϵ
Output: Top-k product recommendations

```
// Step 1: Compute intervals and learn user-item cycles
```

for *each* $(u, i) \in \mathcal{T}$ **do**
  $T_{u,i} \leftarrow$ sorted purchase dates of item i by user u;
  if $|T_{u,i}| \geq 2$ **then**
    $I_{u,i} \leftarrow [t_{p+1} - t_p \mid p = 1, \ldots, |T_{u,i}| - 1]$;
    $Med_{u,i} \leftarrow \text{median}(I_{u,i})$, $\sigma_{u,i} \leftarrow \text{std}(I_{u,i})$;
    $F_{u,i} \leftarrow |T_{u,i}|$; `// Purchase Count`

```
// Step 2: Score and rank recommendations
```

$R \leftarrow \emptyset$;
for *each item i with history for user u* **do**
  $\hat{t}_{next} \leftarrow t_n + Med_{u,i}$; `// Next purchase prediction`
  $Z_{u,i} \leftarrow \frac{|\hat{t}_{next} - t_{target}|}{\max(\sigma_{u,i}, \epsilon)}$; `// Equation 2`
  $S_{u,i} \leftarrow \frac{1}{1 + \lambda \cdot Z_{u,i}}$; `// Equation 3`
  $R \leftarrow R \cup \{(i, S_{u,i})\}$;
Sort R by $S_{u,i}$ descending and return top-k;

Cycle Discovery. The first step establishes a personalized purchase cycle for every product a user buys. For this, we create a customer-product timeline from *timestamp* feature. This converts the discrete order history into a continuous timeline for each user, marking the exact day of each purchase since their very first order. Then, for each user-product pair with at least two purchases, the model computes the time intervals between each consecutive purchase. For example, if a user bought milk on Day 1, Day 8, and Day 15, the list of intervals would be $[7, 7]$. To further validate the cycle, for each user-item pair (u, i), we compute the median time between purchases $Med_{u,i}$ and the standard deviation $\sigma_{u,i}$ by using the history of purchase times $T_{u,i}$.

Cycle-based Scoring. Once the cycles are learned, the second step is to generate recommendation candidates. First we predict the next purchase date $\hat{t}_{next}$ by adding the median cycle to the last purchase date t_n:

$$\hat{t}_{next} = t_n + Med_{u,i} \tag{1}$$

During inference at time t_{target}, the model calculates the absolute error in days and then normalizes this error by dividing it by the standard deviation $\sigma_{u,i}$. We set $\epsilon = 1.5$, which can be interpreted as a clipping threshold to avoid mathematical errors. This gives us the Z distance $Z_{u,i}$, which measures the error in a standardized way:

$$Z_{u,i} = \frac{|\hat{t}_{next} - t_{target}|}{\max(\sigma_{u,i}, \epsilon)} \tag{2}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

The model also accounts for urgency score $S_{u,i}$ using a decay formula with a hyperparameter λ representing a balance between frequency and cycle patterns[5]. This formula penalizes predictions that are too early or too late.

$$S_{u,i} = \frac{1}{1 + \lambda \cdot Z_{u,i}} \tag{3}$$

This heuristic was chosen as a strong, interpretable model because it directly models the most intuitive aspect of grocery shopping. Unlike black-box and complex deep learning models, its logic is transparent and directly tied to observable user behavior. By focusing on the temporal dimension, it captures a signal that is often missed by standard collaborative filtering techniques, which can be critical in a domain like retail where timing is as important as preference.

BPR-Hybrid Scoring. As mentioned earlier, a recommendation model in the retail sector needs to predict not only the regular items' purchase cycle but also suggest exploratory products for a given customer. Therefore, we propose a hybrid model where we combine our heuristic *Cyclic Model* with a latent representation, specifically extracted from BPR. By hybridizing the scoring mechanism, we can efficiently model both habit and exploration simultaneously. The process begins by constructing a candidate set for each user including previously interacted items and new discoveries identified via Bayesian Personalized Ranking (BPR). The algorithm then computes three distinct normalized scoring signals: $f_{u,i}$ as frequency signal, $b_{u,i}$ as BPR signal, and $c_{u,i}$ as cyclic signal. These signals are fused into a final ranking score $R(u, i, t)$ using a weighted linear combination, where hyperparameters β_1, β_2 and β_3 control the relative influence of the collaborative and temporal components.

[5] Higher value of λ gives higher importance to learned cycles

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

Algorithm 2: Hybrid Recommendation via Score Fusion

Input: Training set $\mathcal{D}_{\text{train}}$, test set $\mathcal{D}_{\text{test}}$, lookback window L
Output: Top-k recommendations per order (u, t)
`// Step 1: Build candidate set per user`
foreach *user* u **do**
  $\text{Cand}_u \leftarrow \{\text{Historical candidates}\} \cup \{\text{BPR discovery candidates}\}$;
`// Step 2: Normalized scoring signals`
$f_{u,i} \leftarrow \dfrac{\log(1 + \text{count}(u, i))}{\max_{i'} \ \log(1 + \text{count}(u, i'))}$; `// Frequency Signal`
$b_{u,i} \leftarrow \dfrac{\text{BPR}(u, i)}{\max_{i'} \ \text{BPR}(u, i')}$; `// BPR Signal`
$c_{u,i}(t) \leftarrow \dfrac{\text{S}_{u,i}(t) \cdot \log(1 + n_{u,i})}{\max_{i'} \ (\cdot)}$; `// Cyclic Signal`
`// Step 3: Score fusion and ranking.`
foreach *order* (u, t) *in* $\mathcal{D}_{test}$ **do**
  foreach $p \in Cand_u$ **do**
    $R(u, i, t) \leftarrow \beta_1 \cdot f_{u,i} + \beta_2 \cdot b_{u,i} + \beta_3 \cdot c_{u,i}(t)$;
  $\text{Recs}(u, t) \leftarrow$ Top-k products by $R(u, i, t)$;
return Recs;

4 Experiments and Comparative Study

We evaluated the proposed method on a publicly available benchmark dataset and compared its performance against several baselines, using standard Top-k recommendation metrics commonly adopted in the literature. In addition, we conducted a real-world use-case study with transaction data provided by our retail partner in the EU project, demonstrating the method's practical effectiveness in an operational setting. The benchmark experimental pipeline was built using the RecBole framework [6] to ensure reproducibility and standardized evaluation.

Dataset. We evaluate the proposed method on the InstaCart Online Grocery Market Basket dataset [7], a widely used benchmark dataset on Kaggle used for product recommendation tasks. InstaCart statistical properties are described in Table 1. A key characteristic of the InstaCart dataset is its high sparsity. The user-item interaction matrix, representing all possible user-product pairings, has a density of approximately 99%. This means most users have not purchased most products, which is a typical challenge in recommender systems. In order to prepare data for the experiment we filtered users and products with less than 10 interactions across all the dataset.

[6] https://github.com/RUCAIBox/RecBole
[7] https://www.kaggle.com/c/instacart-market-basket-analysis

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

Table 1. Statistical summary of the original Instacart Market Basket Analysis dataset.

Property	**Value**
Number of users	206,209
Total orders	3,422,821
Distinct products	49,688
Order–product pairs (prior set)	32,434,100
Order–product pairs (train set)	1,384,617
Orders per user (range)	4–100
Features per order	Day of week, hour of day, days since prior order

Model. As presented in Section 3, the proposed model is grounded on the core hypothesis that grocery shopping is inherently cyclical. Many MF-based (lightweight) recommender systems focus only on ***what*** items a user likes, ignoring the critical dimension of ***when*** they might need them again. As shown in Figure 2,

Fig. 2. Customer reordering habits (Instacart data analysis).

analysis of the dataset confirms this hypothesis, showing clear peaks in the *days since prior order* at 7, 14, and 30 days, which correspond to weekly, bi-weekly, and monthly shopping habits. This model directly addresses this temporal pattern to predict a user's next basket with higher relevance. It is worth noting that our model can be adapted at different granularities (e.g., at the category level rather than the item level) while keeping the logic intact.

To determine the optimal hyperparameter for the experiment, a grid search was performed across the following parameter space: frequency weights (β_1) of $\{0.5, 1.0\}$, BPR weights (β_2) of $\{0.1, 0.3, 0.5\}$ and cyclic weights (β_3) of $\{0.3, 0.5, 1.0\}$. A secondary grid search was performed to tune the remaining parameters, evalu-

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

ating λ across $\{0.2, 0.4, 0.6, 0.8, 1.0\}$ and ϵ across $\{0.5, 1.0, 1.5, 2.0, 3.0\}$. The final hyperparameters ($\beta_1 = 1$, $\beta_2 = 0.3$, $\beta_3 = 0.5$, $\lambda = 0.6$, $\epsilon = 1.5$) were selected based on the configuration that maximized recommendation performance on the validation set.

Evaluation Methodology and Metrics. We divide the dataset into a training window of specified months and a testing window that immediately follows. The model learns patterns from the training period and makes predictions for users in the test period. The window slides forward one week (or more, depending on the configuration) and the process repeats and logs the metrics. This allows us to assess how well the model performs across different time periods and detect any seasonal variations or trends in recommendation quality. For example as shown in Figure 3, we consider a configuration where the train data is 6 months, the test and the sliding interval are 1 week each. This process continues, with each performance metric recorded for that specific testing window.

Fig. 3. An example of moving train/test window during training

In Top-k recommendation tasks, traditional accuracy metrics are insufficient. Therefore, evaluation must measure the goodness of the recommendations based on their position in the list. The standard set of metrics universally applied in literature [8], particularly for competitive analysis includes *Precision@K*, *Recall@K*, *F1@K*, and *Normalized Discounted Cumulative Gain (NDCG)* described below, where $\mathcal{I}_k$ denotes the set of Top-k recommended items and $\mathcal{R}$ denotes the set of relevant items (ground truth):

– **Precision@k** measures the proportion of relevant items within the Top-k recommendations. It assesses the accuracy of the recommended list:

$$\text{Precision@k} = \frac{|\mathcal{I}_k \cap \mathcal{R}|}{k}$$

– **Recall@k** measures the coverage of the system, defined as the fraction of all relevant items that are successfully retrieved in the Top-k list:

$$\text{Recall@k} = \frac{|\mathcal{I}_k \cap \mathcal{R}|}{|\mathcal{R}|}$$

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

- **F1-score@k** provides a unified metric by computing the harmonic mean of Precision@k and Recall@k, balancing both accuracy and coverage.
- **NDCG@k** (Normalized Discounted Cumulative Gain) evaluates the quality of the ranking by rewarding relevant items placed higher in the list. It is calculated as the ratio of Discounted Cumulative Gain (DCG) to the Ideal DCG (IDCG):

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

  Here, rel_i represents the relevance score (usually binary or graded) of the item at rank i.

Baselines. We benchmark our method against a set of widely used baselines that reflect different strategies that address the NBR problem. Below, we briefly summarize each baseline.
Frequency Based. This baseline serves as the standard starting point for evaluating any recommender system. It operates on a simple premise: it ranks items based on how frequently they have been purchased or viewed by the user, effectively ignoring other preferences. Despite this lack of personalization, the method is surprisingly tough to beat, even by deep learning-based method [8]. Therefore, before implementing complex and expensive deep learning models, it is essential to verify that they actually outperform this basic approach.
Bayesian Personalized Ranking (BPR). BPR frames the problem as a ranking task. Instead of trying to predict an exact "rating" for an item, BPR focuses on learning the relative preferences of a user. Unlike point-wise methods that predict absolute ratings, BPR aims to maximize the likelihood that the predicted score $\hat{x}_{ui}$ of a positive item i is higher than the score $\hat{x}_{uj}$ of a negative item j for a user u. The objective function is defined as:

$$\mathcal{L}_{BPR} = \sum_{(u,i,j)\in D_S} \ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \lambda_\Theta \|\Theta\|^2$$

where D_S is the set of training triplets (user u, positive item i, negative item j), $\sigma(\cdot)$ is the logistic sigmoid function, $\hat{x}_{ui} - \hat{x}_{uj}$ represents the preference difference, and $\lambda_\Theta \|\Theta\|^2$ is the regularization term to prevent overfitting.
Neural Matrix Factorization (NeuMF). A deep learning model that enhances traditional matrix factorization by combining it with a non-linear neural network model. NeuMF uses two parallel pathways. A Generalized Matrix Factorization (GMF) arm that learns linear relationships and a Multi-Layer Perceptron (MLP) arm that feeds the user and item embeddings to learn complex, non-linear interactions. Our approach extends BPR by adding a temporal component to enhance the prediction capability of timestamped user-product interactions.
LightGCN (Light Graph Convolutional Network): state-of-the-art collaborative filtering method that simplifies standard GCNs by removing non-linear activation functions and feature transformations, arguing that these operations add

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

unnecessary complexity for recommendation tasks. It operates on the user-item bipartite graph, learning embeddings by linearly propagating them through the graph structure. The graph convolution operation at layer $k+1$ aggregates information from neighbors as follows:

$$\mathbf{e}_u^{(k+1)} = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u||\mathcal{N}_i|}} \mathbf{e}_i^{(k)}$$

where $\mathcal{N}_u$ denotes the set of items interacted with by user u, and the term $\frac{1}{\sqrt{|\mathcal{N}_u||\mathcal{N}_i|}}$ serves as the symmetric normalization constant. The final embedding for a user (and similarly for items) is computed as the weighted sum of embeddings from all layers $k=0$ to K:

$$\mathbf{e}_u^* = \sum_{k=0}^{K} \alpha_k \mathbf{e}_u^{(k)}$$

These final representations are then used to predict the preference score via the inner product $\hat{y}_{ui} = (\mathbf{e}_u^*)^T \mathbf{e}_i^*$, optimized typically using the BPR loss for implicit feedback.

Results. The proposed method and the baseline models were evaluated using standard Top-k recommendation metrics, namely, *Precision@k*, *Recall@k*, *F1@k*, and *NDCG@k* for $k = \{5, 10, 20\}$. The evaluation was performed on a held-out test set of each user's most recent orders and are shown in Table 2.

In NBR, users' future purchases are strongly influenced by their historical baskets, which introduces a pronounced repetition bias toward previously purchased items [8]. In the InstaCart dataset, a substantial proportion of transactions consist of repetitive purchases, which explains why simple baselines such as popularity-based methods achieve competitive performance. Consequently, the performance margin between these baselines and our proposed method is relatively small. In particular, as shown in Table 2, the frequency-based method constitutes a strong baseline, reflecting the high level of repetitive purchasing behavior in the dataset. Compared to this baseline, the proposed method achieves consistent improvements in all metrics and cutoff levels, most notably, NDCG@5 improves by +2.73% and Precision@10 by +2.98% in relative terms. The improvement is sustained in all evaluation folds, with gains ranging from +1.9% to +3% relatively, and is strongest in ranking quality metrics (NDCG) where the method achieves absolute gains of over 1 percent, suggesting that our method places relevant items higher in the recommendation list compared to the frequency-based method despite the strong repetition bias present in the data.

Figure 4 shows our approaches's effectiveness for balancing the precision and recall at K=10, accurately predicting the next users purchases but still proposing novel items. As this figure demonstrates, as the list of recommended items increases, the models expose discovery behavior, suggesting novel items.

Scalability Analysis. The cyclic scoring module does not add extra computational complexity on top of the BPR method, and is scalable. It requires no iterative optimization passes (epochs), as all statistics are computed in a single scan.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

Table 2. Benchmark results for K = 5,10 and 20 (Instacart Dataset)

Model	@5				@10				@20			
	Prec	Rec	F1	NDCG	Prec	Rec	F1	NDCG	Prec	Rec	F1	NDCG
Our method	**0.3963**	**0.2384**	**0.2607**	**0.4518**	**0.3112**	**0.3484**	**0.2894**	**0.4308**	**0.2224**	**0.4660**	**0.2706**	**0.4467**
Frequency	0.3851	0.2319	0.2534	0.4398	0.3022	0.3393	0.2813	0.4190	0.2175	0.4572	0.2649	0.4361
BPR	0.2300	0.1348	0.1497	0.2634	0.1797	0.1990	0.1668	0.2491	0.1302	0.2743	0.1590	0.2606
LightGCN	0.1406	0.0731	0.0962	0.1603	0.1106	0.1135	0.1120	0.1487	0.0840	0.1702	0.1125	0.1577
NeuMF	0.0921	0.0444	0.0599	0.1015	0.0759	0.0727	0.0743	0.0960	0.0604	0.1150	0.0792	0.1041

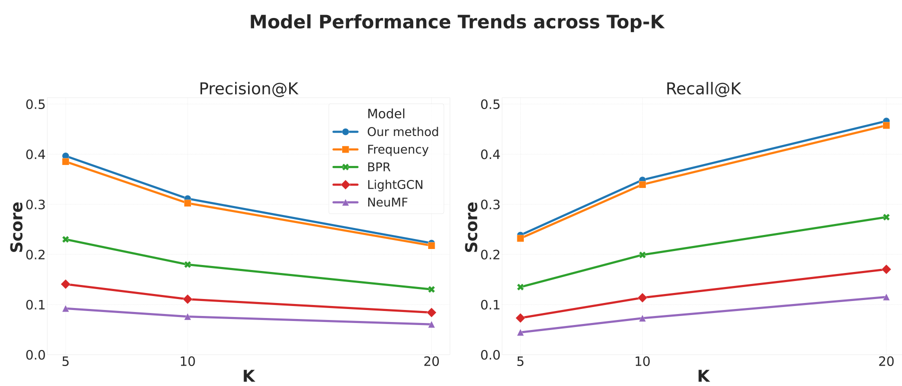

Fig. 4. Precision vs Recall Trade-offs

The cost lies only in the pre-processing step, specifically sorting the interaction history to compute intervals between consecutive purchases.

- **Training.** The extraction of repurchase intervals requires sorting the dataset, for user, product, and Timestamp. Using an efficient sorting algorithm, this operation scales as $O(N \log N)$. The subsequent aggregation of statistics is performed in a linear scan $O(N)$. Thus, the total training complexity of the cyclic component is dominated by the sorting step.
- **Inference.** Generating the cyclic urgency scores involves a vectorized projection of the next purchase date and the calculation of the Z-score for each candidate pair. Since this is a direct arithmetic operation on the candidate set, the inference complexity of this component is $O(N)$.

4.1 Case Study: MC Sonae

Next, we analyzed the performance of the proposed method on real data within an EU-funded project partnership. The company dataset consists of the past 3 years of data from 320k customers and 13 million orders. A sample of this dataset, as reported in Table 3, was used in our experiments.

The dataset includes *anonymized customer id*, *purchase date and time*, *order ID*, *product ID*, *product category and subcategory*. This dataset is similar to open-sourced Instacart dataset in terms of fields, context, and sparsity.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

Table 3. Statistical summary of the MC Sonae dataset (January 1, 2024 up to May 30, 2025.

Property	Value
Number of users	99,992
Total orders	1,582,864
Distinct products	108,894
Transactions	20,598,714

Table 4. MC Sonae Results at different cutoff values K

Model	@5				@10				@20			
	Prec	Rec	F1	NDCG	Prec	Rec	F1	NDCG	Prec	Rec	F1	NDCG
Our method	**0.2093**	**0.0891**	**0.1001**	**0.2391**	**0.1664**	**0.1301**	**0.1165**	**0.2178**	**0.1238**	**0.1800**	**0.1199**	**0.2067**
BPR	0.1062	0.0476	0.0521	0.1223	0.0865	0.0712	0.0618	0.1137	0.0680	0.1039	0.0669	0.1123
Frequency	0.2019	0.0855	0.0961	0.2312	0.1603	0.1246	0.1119	0.2101	0.1198	0.1730	0.1158	0.1994

Results. Table 4 reports the performance of the proposed method across different cutoff values. As expected, precision and NDCG decrease with increasing K, reflecting the growing difficulty of maintaining highly relevant rankings as more items are recommended. Conversely, recall and F1 increase steadily, indicating improved coverage of relevant items at larger cutoffs. Notably, the model achieves its highest ranking quality at K=5 (NDCG@5 = 0.2391), highlighting its effectiveness in prioritizing relevant items at the top of the recommendation list. At K=10, the results exhibits a favorable balance between precision and recall, while K=20 emphasizes recall-oriented performance. This behavior is driven by the inherent trade-off between Precision and Recall as the recommendation list size K grows. At $K = 10$, the model is more selective, resulting in a higher density of relevant items (improved Precision) while still capturing key user interests. When we increase to $K = 20$, it is like expanding the selections. This naturally increases Recall as there are more opportunities to catch relevant items, but usually decreases Precision, as the items ranked lower (11–20) are more likely to be irrelevant 'noise' compared to the top 10. These results demonstrate the robustness of the proposed method across different recommendation scenarios.

5 Discussion

In the context of retail recommendation systems, precision holds greater significance than recall due to its direct impact on customer experience and business performance. When customers receive recommendations, they expect them to be relevant and timely. If the system frequently suggests products that customers do not need, it creates recommendation fatigue and erodes trust in the system. Customers who repeatedly encounter irrelevant suggestions are likely to ignore future recommendations entirely, regardless of their actual relevance. This is particularly critical in cyclic purchasing scenarios where the goal is to anticipate needs at precisely the right moment. A recommendation that arrives too early or suggests

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

a product the customer has no intention of buying wastes both the customer's attention and the company's marketing resources. Therefore, maintaining high precision ensures that each recommendation carries weight and credibility with the customer. From a business perspective, precision directly influences conversion rates, which are fundamental metrics for evaluating the return on investment of any recommendation system. High precision means that a greater proportion of recommended products actually result in purchases, demonstrating clear value to stakeholders and justifying the resources allocated to developing and maintaining the system. A system with lower precision may technically capture more eventual purchases through high recall, but if it overwhelms customers with numerous irrelevant suggestions, the actual purchase rates will suffer dramatically. In competitive retail environments where customer attention is scarce and marketing budgets are finite, each recommendation represents an opportunity cost. Prioritizing precision ensures that these opportunities are used effectively, maximizing both customer satisfaction and business outcomes while building a sustainable foundation for long term customer engagement. Beyond general repetitive patterns, external factors such as price fluctuations and promotions significantly influence consumer behavior. However, these factors are outside the scope of this paper, as our benchmark dataset does not include this information.

6 Conclusion and Future work

Our study introduces a cost efficient approach combining the advantage of traditional collaborative methods with cycle-aware recommendations in the retail domain. Our experiments on an open source benchmark dataset demonstrate the method's capability to improve different metrics up to 3% with comparable performance for larger basket sizes. Additionally, we demonstrated the method's effectiveness on real-world retail data, encouraging researchers and practitioners to explore the potential of scalable and interpretable models for the NBR task. In another direction, we aim to study the NBR problem as a sequence of Next Item Predictions (NIPs), formulating a Reinforcement Learning (RL) problem that can be solved by different policy optimization algorithms.

Acknowledgments. This work is based on the MSc. thesis of FF, carried out at VUB. This paper is part of a PEER project that has received funding from the EU Research and Innovation Programme, under Grant Agreement number 101120406. The paper reflects only the authors' view and the EC is not responsible for any use that may be made of the information it contains. We also gratefully acknowledge the MC Sonae for providing essential use case data, which were instrumental to this work.

Disclosure of Interests. The authors declare that they have no competing interests to declare.

References

1. Anderson, A., Kumar, R., Tomkins, A., Vassilvitskii, S.: The dynamics of repeat consumption. In: Proceedings of the 23rd international conference on World wide web. pp. 419–430 (2014)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20

2. Ariannezhad, M., Jullien, S., Li, M., Fang, M., Schelter, S., De Rijke, M.: Recanet: A repeat consumption-aware neural network for next basket recommendation in grocery shopping. In: Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval. pp. 1240–1250 (2022)
3. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: 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. pp. 639–648 (2020)
4. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the 26th international conference on world wide web. pp. 173–182 (2017)
5. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: 2008 Eighth IEEE international conference on data mining. pp. 263–272. Ieee (2008)
6. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 30–37 (2009)
7. Lashinin, O., Krasilnikov, D., Milogradskii, A., Ananyeva, M.: Saferec: Self-attention and frequency enriched model for next basket recommendation. In: European Conference on Information Retrieval. pp. 294–302. Springer (2025)
8. Li, M., Jullien, S., Ariannezhad, M., De Rijke, M.: A next basket recommendation reality check. ACM Transactions on Information Systems **41**(4), 1–29 (2023)
9. Mnih, A., Salakhutdinov, R.R.: Probabilistic matrix factorization. Advances in neural information processing systems **20** (2007)
10. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: Bayesian personalized ranking from implicit feedback. arXiv preprint arXiv:1205.2618 (2012)
11. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web. pp. 285–295 (2001)
12. Sedhain, S., Menon, A.K., Sanner, S., Xie, L.: Autorec: Autoencoders meet collaborative filtering. In: Proceedings of the 24th international conference on World Wide Web. pp. 111–112 (2015)
13. Zhang, S., Yao, L., Sun, A., Tay, Y.: Deep learning based recommender system: A survey and new perspectives. ACM computing surveys (CSUR) **52**(1), 1–38 (2019)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_20