A Next Basket Recommendation Reality Check

Ming Li, Sami Jullien, Mozhdeh Ariannezhad, and Maarten de Rijke

University of Amsterdam, The Netherlands
{m.li, s.jullien, m.ariannezhad, m.derijke}@uva.nl

Abstract. The goal of a next basket recommendation system is to recommend items for the next basket for a user, based on the sequence of their prior baskets. Recently, a number of methods with complex modules have been proposed that claim state-of-the-art performance. They rarely look into the predicted basket and just provide intuitive reasons for the observed improvements, e.g., better representation, capturing intentions or relations, etc. We provide a novel angle on the evaluation of next basket recommendation (NBR) methods, centered on the distinction between repetition and exploration: the next basket is typically composed of previously consumed items (i.e., repeat items) and new items (i.e, explore items). We propose a set of metrics that measure the repeat/explore ratio and performance of NBR models. Using these new metrics, we analyze the state-of-the-art NBR models. The results of our analysis help to clarify the extent of the actual progress achieved by existing NBR methods as well as the underlying reasons for the improvements. Overall, our work sheds light on the evaluation problem of NBR and provides useful insights into the model design for this task.

Keywords: Next basket recommendation · Repetition and exploration · Evaluation.

1 Introduction

Over the years, next basket recommendation (NBR) has received a considerable amount of interest from the research community [1, 13, 18]. Baskets, or sets of items that are purchased or consumed together, are pervasive in many real-world services, with e-commerce and grocery shopping as prominent examples [5, 12]. Given a sequence of baskets that a user has purchased or consumed in the past, the goal of a NBR system is to generate the next basket of items that the user would like to purchase or consume next. Within a basket, items have no temporal order and are equally important. A key difference between NBR and session-based or sequential item recommendations is that NBR systems need to deal with multiple simultaneous items in one set. Therefore, models designed for item-based recommendation are not fit for basket-based recommendation, and dedicated NBR methods have been proposed [6, 14].

Numerous deep learning techniques have been developed to address sequential item recommendation problems, building on the well-documented capacity of neural methods to capture hidden relations and automatically learn representations [2]. Against this background, recent years have witnessed many proposals

to address different aspects of the NBR task with neural methods, e.g., item-to-item relations [10], cross-basket relations [19], noise within a basket [11].

A recent analysis of neural methods for sequential recommendation problems shows that such methods may be outperformed by simple nearest neighbor or graph-based baselines [3]. In this paper, we perform a similar analysis for the next basket recommendation task (instead the generic sequential recommendation problem). We notice that the same factors that account for the phenomena observed by Dacrema et al. [3] also feature in recent research on the next basket recommendation task. This includes weak or missing baselines, the use of different datasets in different papers, and of non-standard metrics. For example, some methods [1, 10, 11, 17, 18] only compare with previous (deep) learning-based methods and avoid comparing with a simple Personal-TopFreq baseline, which recommends the most k frequent items in the users' historical records as the next basket. Many recent publications [4, 7, 11, 19] do not compare with each other. And in [11], sampled metrics [8] are used to evaluate the performance even though this is not encouraged [9]. Moreover, previous studies into models for NBR tend to use limited case studies and ablation studies to demonstrate the effectiveness of their proposed methods. They forego a deeper analysis of what items are actually recommended in the next basket.

Given the state of affairs sketched above, we believe that it is important to perform a reality check on current NBR methods, in order to find out how much progress we have actually made and whether a specific proposed module is responsible for the observed improvements. We propose a novel *repetition and exploration* perspective to evaluate next basket recommendation methods. Baskets recommended in a NBR scenario consist of *repeat items* (items that the user consumed before, in previous baskets) and *explore items* (items that are new to the user). In order to improve our understanding of the relative performance of NBR models, especially regarding repeat items and explore items, we introduce a set of task-specific metrics for NBR. Those metrics help us to understand what type of items are present in the recommended basket and assess the performance of models when proposing new items vs. already-purchased items.

With the proposed metrics for NBR and a new baseline, GP-Freq, we evaluate the performance of state-of-the-art NBR models on three benchmark datasets, with different repeat-explore patterns. We arrive at several important findings: (1) No NBR method can consistently outperform all other methods across different datasets. (2) All published methods are heavily skewed to either repetition or exploration compared to the ground-truth, which might harm long-term engagement. (3) There is a large performance gap between repetition and exploration; repeat item recommendation is much more easier. (4) Several deep learning-based neural methods are inferior to simple frequency baselines P-TopFreq and GP-TopFreq in many settings. (5) Being biased towards repetition and its improvements in repetition accounts for most of the performance gains of recently published methods, even though many complex modules or strategies specifically target explore items. (6) GP-TopFreq is a highly competitive baseline across different datasets, and comparing to it, observed improvements of several state-of-the-art methods are limited.

Overall, our work sheds light on the state-of-the-art of NBR, provides suggestions to revise the metrics of NBR evaluation, helps to understand the current progress and the underlying reasons for improvements, and provides further insights for the design of NBR models.

2   Datasets and Methods

2.1   Datasets

In order to ensure the reproducibility of our study, we conduct our experiments on three publicly available real-world datasets, which vary in the ratio of repetition and exploration:

– **TaFeng** – contains four months of shopping transactions collected from a Chinese grocery store. All products purchased on the same day by the same consumer are treated as a basket.[1]
– **Dunnhumby** – covers two years of household-level transactions at a retailer. All products bought by the same customer in the same transaction are treated as a basket. We use the first two months of the data in our experiments.[2]
– **Instacart** – contains over three million grocery orders of Instacart users. We treat all items purchased in the same order as a basket.[3]

Table 1: Statistics of the processed datasets.

Dataset	#elements	#users	#avg. basket size	#avg. baskets/user	repeat ratio	explore ratio
TaFeng	11,997	13,858	6.27	6.58	0.188	0.812
Dunnhumby	3,920	22,530	7.45	9.53	0.409	0.591
Instacart	13,897	19,435	9.61	13.21	0.597	0.403

In each dataset, users with a basket size between three to 50 are sampled to conduct experiments. We also remove rare items. The remainder covers more than 95% of the interactions. The statistics of the processed datasets are summarized in Table 1, where the repetition ratio and exploration ratio are calculated based on the ground truth basket as the proportion of *repeat items* and *explore items* in the ground truth basket, respectively. For our experiments, we split every dataset into a training, validation, and test set with a ratio of 72%, 8%, and 20% for every user.

2.2   Baseline methods and reproducible methods

We follow the same strategy as [3] to collect reproducible NBR papers. Papers targeting sequential set prediction are also considered to be relevant to NBR. For a fair comparison, we do not include methods that perform quite poorly

---

[1] https://www.kaggle.com/chiranjivdas09/ta-feng-grocery-dataset

[2] https://www.dunnhumby.com/source-files/

[3] https://www.kaggle.com/c/instacart-market-basket-analysis/data

[13, 15] or leverage additional information like explicit timestamp [16], item attributes [1], etc. Apart from the two widely known frequency-based baselines (global top-k and personal top-k), we propose a simple combination of the two as an additional baseline, i.e., GP-TopFreq.

Frequency-based baselines.
- **G-TopFreq** – uses the k most popular items in the dataset to form the recommended next basket. It is widely used in recommendation systems due to its effectiveness and simplicity.
- **P-TopFreq** – a personalized TopFreq method that treats the most frequent k items in the users' historical records as the next basket. This method only has repeat items in the prediction.
- **GP-TopFreq** – a combination of P-TopFreq and G-TopFreq that first uses P-TopFreq to fill the basket, then uses G-TopFeq to fill any remaining slots.

Neighbor-based methods.
- **TIFUKNN:** – models the temporal dynamics of frequency information of users' past baskets to introduce Personalized Frequency Information (PIF), then uses a nearest neighbor-based method on the PIF [7].
- **UP-CF@r** – a combination of the recency-aware user-wise popularity and user-wise collaborative filtering. The recency of shopping behavior is considered in this method [4].

Deep learning-based methods.
- **Dream** – the first neural-based method that models users' global sequential basket history for NBR. It uses a pooling strategy to generate basket representations, which are then fed into an RNN to learn user representations and predict the corresponding next set of items [18].
- **Sets2Sets** – uses a pooling operation to get basket embeddings and an attention mechanism to learn a user's representation from their past interactions. Furthermore, item frequency information is adopted to improve performance [6].
- **DNNTSP** – leverages a GNN and self-attention techniques. It encodes item-item relations via a graph and employs a self-attention mechanism to capture temporal dependencies of users' basket sequence [19].
- **Beacon** – an RNN-based method that encodes the basket considering the incorporating information on pairwise correlations among items [10].
- **CLEA:** – an RNN-based method that uses a contrastive learning model to automatically extract items relevant to the target items and generates the representation via a GRU-based encoder [11].

For each method, we use the suggested hyper-parameters in the paper or related GitHub repository.

3 Performance Comparison Using Conventional Metrics

3.1 Conventional metrics

We adopt three conventional metrics: Recall, Normalized Discounted Cumulative Gain (NDCG), and Personalized Hit Ratio (PHR), which are commonly used

Table 2: Overall performance comparison of simple baselines and published state-of-the-art methods. Highlights indicate the highest score per basket size and metric.

Dataset		TaFeng			Dunnhumby			Instacart		
Size	Methods	Recall	NDCG	PHR	Recall	NDCG	PHR	Recall	NDCG	PHR
	G-TopFreq	0.0803	0.0842	0.2489	0.0987	0.1054	0.4624	0.0721	0.0820	0.4543
	P-TopFreq	0.1072	0.0959	0.3487	0.2319	0.2342	0.6569	0.3264	0.3381	0.8437
	GP-TopFreq	0.1215	0.1019	0.3706	0.2356	0.2360	0.6660	0.3273	0.3387	0.8451
10	UP-CF@r	0.1257	0.1110	0.3996	0.2429	0.2471	0.6761	0.3511	0.3634	0.8642
	TIFUKNN	0.1259	0.1020	0.3871	0.2398	0.2411	0.6774	0.3608	0.3726	0.8640
	Dream	0.1143	0.1030	0.2991	0.0974	0.1049	0.4639	0.0722	0.0818	0.4560
	Beacon	0.1180	0.1075	0.3006	0.0991	0.1055	0.4655	0.0724	0.0820	0.4575
	CLEA	0.1184	0.1048	0.3083	0.1548	0.1726	0.5533	0.1227	0.1444	0.5633
	Sets2Sets	0.1360	0.1132	0.4104	0.1708	0.1491	0.5854	0.2125	0.1923	0.7185
	DNNTSP	0.1537	0.1321	0.4487	0.2388	0.2409	0.6771	0.3337	0.3401	0.8498
	G-TopFreq	0.1071	0.0937	0.3284	0.1267	0.1195	0.5315	0.1001	0.0970	0.5254
	P-TopFreq	0.1403	0.1103	0.4349	0.3031	0.2697	0.7183	0.4306	0.3940	0.8944
	GP-TopFreq	0.1703	0.1215	0.4854	0.3146	0.2747	0.7387	0.4353	0.3962	0.8999
20	UP-CF@r	0.1710	0.1288	0.4900	0.3186	0.2852	0.7347	0.4594	0.4224	0.9072
	TIFUKNN	0.1828	0.1237	0.5063	0.3201	0.2804	0.7427	0.4708	0.4324	0.9090
	Dream	0.1479	0.1162	0.3986	0.1315	0.1216	0.5399	0.1006	0.0971	0.5313
	Beacon	0.1479	0.1142	0.4001	0.1316	0.1218	0.5413	0.1005	0.0975	0.5307
	CLEA	0.1478	0.1165	0.3955	0.1869	0.1861	0.6281	0.1542	0.1600	0.6395
	Sets2Sets	0.1917	0.1350	0.5285	0.2574	0.1903	0.6940	0.3077	0.2402	0.8284
	DNNTSP	0.2087	0.1540	0.5518	0.3235	0.2825	0.7410	0.4434	0.3989	0.9048

by previous NBR studies for evaluation. Recall measures the ability to find all relevant elements; NDCG is a ranking quality measurement metric, which takes the order into consideration; PHR focuses on user-level performance and calculates the ratio of predictions that capture at least one item in the ground truth basket. $F1@k$ and $Precision@k$ are also employed as evaluation metrics in recent work. However, we argue that these two metrics are not necessary when $Recall@k$ is used. In the NBR setting, we usually recommend a basket (i.e., positive instances) with a fixed size k to the user, which means that $Precision@k$ and $F1@k$ are proportional to $Recall@k$ for each user. We will therefore not use $F1@k$ and $Precision@k$.

3.2 Results with conventional metrics

The performance results for the conventional metrics are shown in Table 2. First, the performance of different methods varies across datasets and there is no method that unanimously outperforms all other methods, independent of datasets and basket size. This calls for a further analysis of the factors impacting performance, which we conduct in the next section.

Next, mong simple baselines, P-TopFreq outperforms G-TopFreq in all scenarios, which suggests that personalization improves the recommendation performance. P-TopFreq could only recommend items that appeared in historical sequences and some users only have interacted with items fewer than the basket size. P-TopFreq is a competitive repetition baseline, as it only contains repeat items in the prediction. It is worth noting that the number of repetition candidates of the user may be smaller than the basket size, which means there might be empty slots in the recommended basket of P-TopFreq. In contrast, GP-TopFreq could take full use of given basket slots by filling the empty slots via top-k items derived by G-TopFreq. GP-TopFreq beat P-TopFreq with no surprise, and the gain shrinks as the repeat ratio of the dataset increases. It is also worth noting that all recent methods can fulfill the given basket. For a fair comparison, we believe the proposed GP-TopFreq should be the basic baseline that every NBR method should compare with, especially in a high exploration scenario.

As to the neighbor-based methods, we can see that TIFUKNN and UP-CF@r have similar performance across different scenarios and outperform all simple frequency-based baselines. The two methods are similar in the sense that they both model the temporal information, combined with a user-based nearest neighbor method. Their performance in high exploration scenario is lower than several learning-based methods, but when it comes to Dunnhumby and Instacart datasets with relatively low exploration ratio, they are among the best performing methods.

We observe that most of the training-based deep learning methods outperform G-TopFreq, which is the only simple frequency-based baseline in many papers. Surprisingly, P-TopFreq and GP-TopFreq also achieve a highly competitive performance and outperform four deep learning methods (i.e., Dream, Beacon, CLEA and Sets2sets), by a large margin in the Dunnhumby and Instacart datasets, e.g. the improvement w.r.t. $Recall@10$ is from 35.8% to 141.9% and from 53.6% to 353.3% respectively. Moreover, the proposed GP-Topfreq baseline could even outperform the deep-learning based Beacon, Dream and CLEA algorithm in Tafeng, the scenario with a high exploration ratio. Among these methods, DNNTSP is the only one to have a consistently high performance in all scenarios.

4 Performance w.r.t Repetition and Exploration

Conventional metrics only provide a general idea of the performance of the methods. In order to understand the underlying reasons of the performance, we dive deeper into the basket components from a repetition and exploration perspective.

4.1 Metrics for repetition vs. exploration

In order to evaluate the quality of the recommended basket, we propose several novel metrics with regard to the repetition and exploration in a basket.

An item is considered to be a *repeat item* e^r for a user i if it appears in the user's historical basket sequence, that is $e^r \in E_i^{rep} = B_i^1 \cup B_i^2 \cup \cdots \cup B_i^t$.

Otherwise, the item is an *explore item*, denoted $e^e \in E^{expl}$. The predicted next basket B_i^{t+1} is also denoted as P_{u_i}, which comprises of *repeat items* and *explore items*, that is, $P_{u_i} = \{e_1^r, e_2^r, \ldots, e_k^e\}$, where k denotes the size of basket. It is worth noting that the basket can consist only of explore or repeat items.

To understand the elements in the predicted basket, we propose the *repetition ratio RepR* and the *exploration ratio ExplR* to measure the tendency of the recommendation. The overall *RepR* and *ExplR* are calculated over all test users as:

$$RepR = \frac{1}{N} \sum_{i=1}^{N} \frac{\left|P_{u_i}^{rep}\right|}{K}, \quad ExplR = \frac{1}{N} \sum_{i=1}^{N} \frac{\left|P_{u_i}^{expl}\right|}{K}, \tag{1}$$

where N denotes the number of test users, K is the size of the model's predicted basket for user u_i, $P_{u_i}^{rep}$ and $P_{u_i}^{expl}$ is the set of all repeat items $e^r \in P_{u_i}$ and all exploration items $e^e \in P_{u_i}$ respectively.

Furthermore, we pay attention to the basket's ability to fulfill the users' need for repetition and exploration, and we propose the following metrics to evaluate the performance. $Recall_{rep}$ and PHR_{rep} are used to evaluate the *Recall* and *PHR* w.r.t. the repetition performance. Similarly, we use $Recall_{expl}$ and PHR_{expl} to assess the exploration performance. These metrics are calculated as follows:

$$Recall_{rep} = \frac{1}{N_r} \sum_{i=1}^{N_r} \frac{\left|P_{u_i} \cap T_{u_i}^{rep}\right|}{\left|T_{u_i}^{rep}\right|} \qquad PHR_{rep} = \frac{\sum_{i=1}^{N_r} \varphi(P_{u_i}, T_{u_i}^{rep})}{N_r} \tag{2}$$

$$Recall_{expl} = \frac{1}{N_e} \sum_{i=1}^{N_e} \frac{\left|P_{u_i} \cap T_{u_i}^{expl}\right|}{\left|T_{u_i}^{expl}\right|} \qquad PHR_{expl} = \frac{\sum_{i=1}^{N_e} \varphi(P_{u_i}, T_{u_i}^{expl})}{N_e}, \tag{3}$$

where N_r and N_e denote the number of users whose ground truth basket contains repeat elements and exploration elements respectively; $\varphi(P, T)$ returns 1 when $P \cap T \neq \emptyset$, otherwise returns 0.

4.2 The components of a recommended basket

We analyze the components of the recommended basket for each method to understand what forms up the recommendation. The results are shown in Figure 1. First, we can see that all the recommended baskets are heavily skewed towards either item repetition or exploration, relative to the ground-truth baskets that are much more balanced between already seen and new items. This means none of the existing published methods can properly balance the repeat items and explore items of users' future interests, and might have negative effects on the long-term engagement of user.

According to the basket components, we can divide the existing methods into repeat-biased methods (i.e. P-TopFreq, Gp-TopFreq, Sets2sets, DNNTSP, UP-CF@r, and TIFUKNN) and explore-biased methods (i.e. G-TopFreq, Dream, Beacon, and CLEA) and a large gap exist between the two types. Note that TIFUKNN and UP-CF@r both have hyper-parameters which could implicitly influence the tendency of recommendation, here we show the results of the hyper-parameters that account for the best overall performance. Among explore-biased
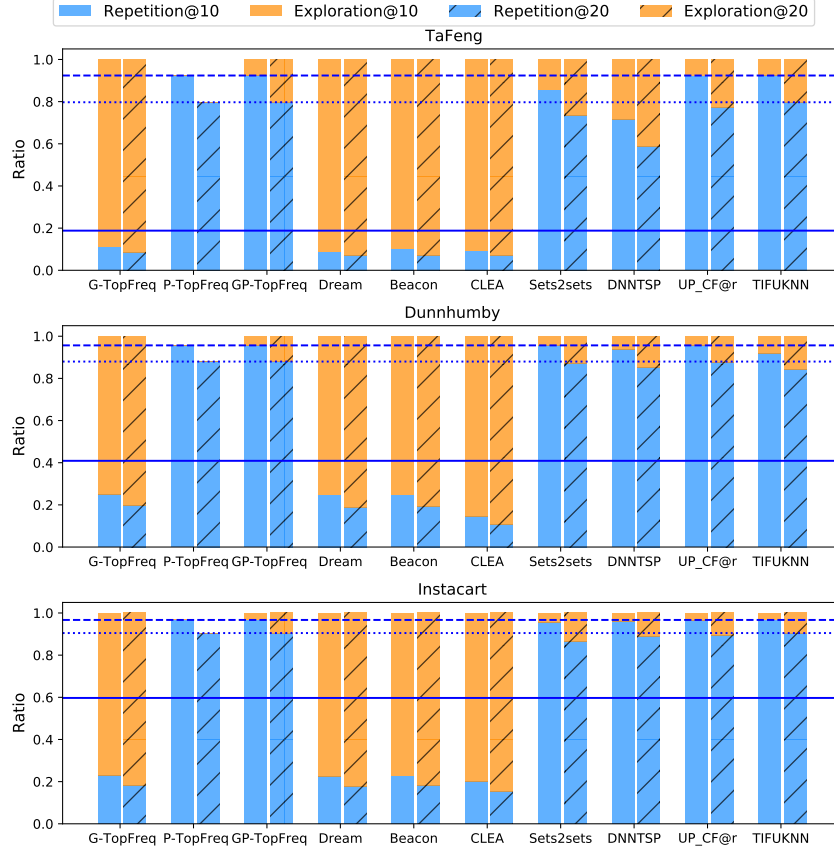
Fig. 1: The repetition and the exploration ratio of basket components; solid line indicates the repetition ratio of the ground-truth; dashed line indicates the upper bound of the repetition ratio for the basket size 10; dotted line indicates the upper bound of the repetition ratio for the basket size 20.

methods, G-TopFreq is not a personalized method and always provides the popular items of the scenario. Dream, Beacon, and CLEA treat all items without any discrimination, which means the exploration items are more likely to be in the predicted basket and their basket components are similar to G-TopFreq. Together with the performance in Table 2, we can see that repeat-biased methods could generally perform much better than explore-biased methods on conventional metrics across the datasets, especially when the dataset has a relatively high repeat ratio.

It is worth noting that the repeat ratio of P-TopFreq and GP-TopFreq serves as the upper bound repeat ratio of the recommended basket, and most recommended basket of the repeat-biased method are close to or reach this upper bound, even when the datasets have quite a low ratio of repeat behavior in ground-truth, except for two cases (Sets2sets and DNNTSP on the Tafeng dataset). Apart from that, the exploration ratio of repeat-biased methods increases from basket size 10 to 20, however, we believe this is because there are

Table 3: Repetition and exploration performance comparison of simple baselines and published state-of-the-art methods. Highlights indicate the highest score per basket size and metric.

Size	Methods	TaFeng Recall-rep	Recall-expl	PHR-rep	PHR-expl	Dunnhumby Recall-rep	Recall-expl	PHR-rep	PHR-expl	Instacart Recall-rep	Recall-expl	PHR-rep	PHR-expl
	G-TopFreq	0.1268	0.0573	0.1947	0.1738	0.1882	0.0393	0.4954	0.1590	0.1002	0.0382	0.4138	0.1575
	P-TopFreq	0.5234	0.0000	0.6766	0.0000	0.5612	0.0000	0.8553	0.000	0.5388	0.0000	0.9085	0.0000
	GP-TopFreq	0.5234	0.0157	0.6766	0.0266	0.5612	0.0049	0.8553	0.0148	0.5388	0.0014	0.9085	0.0046
	UP-CF@r	0.6046	0.0086	0.7515	0.0153	0.5913	0.0018	0.8752	0.0064	0.5805	0.0011	0.9295	0.0038
10	TIFUKNN	0.5616	0.0176	0.7037	0.0284	0.5726	0.0082	0.8646	0.0257	0.5931	0.0018	0.9287	0.0060
	Dream	0.1312	0.0921	0.1874	0.2378	0.1843	0.0412	0.4906	0.1668	0.1007	0.0367	0.4181	0.1503
	Beacon	0.1456	0.0822	0.2126	0.2223	0.1893	0.0404	0.4963	0.1605	0.1010	0.0365	0.4193	0.1492
	CLEA	0.1442	0.0935	0.2073	0.2398	0.2492	0.0569	0.5702	0.1803	0.1715	0.0337	0.5435	0.1297
	Sets2Sets	0.5647	0.0271	0.7222	0.0495	0.4260	0.0026	0.7583	0.0064	0.3515	0.0005	0.7713	0.0013
	DNNTSP	0.5291	0.0556	0.6912	0.1294	0.5861	0.0073	0.8684	0.0224	0.5477	0.0018	0.9110	0.0054
	G-TopFreq	0.1637	0.0789	0.2530	0.2385	0.2279	0.0609	0.5565	0.2353	0.1335	0.0602	0.4767	0.2251
	P-TopFreq	0.7251	0.0000	0.8439	0.0000	0.7399	0.0000	0.9353	0.0000	0.7193	0.0000	0.9631	0.0000
	GP-TopFreq	0.7251	0.0339	0.8439	0.0724	0.7399	0.0153	0.9353	0.0470	0.7193	0.0084	0.9631	0.0285
	UP-CF@r	0.7785	0.0259	0.8742	0.0560	0.7718	0.0078	0.9430	0.0261	0.7612	0.0083	0.9735	0.0215
20	TIFUKNN	0.7664	0.0410	0.8707	0.0806	0.7474	0.0159	0.9344	0.0677	0.7747	0.0108	0.9728	0.0318
	Dream	0.1723	0.1234	0.2572	0.3264	0.2316	0.0687	0.5612	0.2596	0.1348	0.0586	0.4849	0.2222
	Beacon	0.1748	0.1230	0.2621	0.3262	0.2333	0.0672	0.5628	0.2568	0.1338	0.0599	0.4822	0.2252
	CLEA	0.1799	0.1217	0.2675	0.3189	0.2808	0.0874	0.6168	0.2648	0.2081	0.0578	0.6051	0.2044
	Sets2Sets	0.7478	0.0558	0.8703	0.1166	0.6317	0.0086	0.8881	0.0273	0.5095	0.0032	0.8873	0.0073
	DNNTSP	0.6820	0.0899	0.8130	0.2103	0.7758	0.0183	0.9391	0.0609	0.7284	0.0095	0.9657	0.0320

no extra repeat items available, and does not mean that the model actively increases the explore ratio in a larger basket setting.

4.3 The performance w.r.t Repetition and Exploration

The results w.r.t. repetition and exploration performance are shown in Table 3. First of all, using our newly proposed metrics, we observe that the repetition performance $Recall_{rep}$ is always higher than the exploration performance, even when the explore items form almost 90% of the recommended basket. This shows that the repetition task (recommending repeat items) and the exploration task (recommending explore items) have different levels of difficulty and capturing users' repeat behavior is much easier.

Second, three recurrent neural network (RNN)-embedding based methods perform worst w.r.t. the repeat item recommendation and best w.r.t. the explore item recommendation, as they are heavily skewed towards explore items. Moreover, the pooling strategy to get basket embeddings leads to information loss, which also limits performance on the repetition task. Also, we can see that there are indeed improvements in the exploration performance compared to G-TopFreq with the same level of exploration ratio, which indicates that the representation learned by these methods does capture the hidden sequential transition relationship between items. Repeat-biased methods perform better w.r.t. repetition in all settings, since the baskets they predict contain more repeat items.

9

Similarly, we can see that DNNTSP, UP-CF@r, and TIFUKNN perform better than P-TopFreq w.r.t. the repeat performance with the same or a lower level of repetition ratio.

Third, explore-biased methods spend more resources on the more difficult and uncertain task of explore item prediction, which is not an ideal option when considering the general performance. On the other side, being biased towards the easier task of repeat item prediction leads to performance gains in overall performance, which is positively correlated with the repeat ratio of the dataset.

4.4   The contribution of repetition and exploration

Even though a clear improvement w.r.t. either repeat or explore performance is observed in the previous section, it does not mean that is the reason for the better overall performance, since repeat and explore items account for different ground-truth proportions in different datasets. To better understand where the performance gains of the well-performing methods in Table 2 come from, we remove repeat items or explore items in the predicted basket and compare the performance with the original basket, respectively. Experimental results on three datasets are shown in Figure 2. We consider G-TopFreq, P-TopFreq, and GP-TopFreq as simple baselines to compare with.

From Figure 2, we conclude that Dream and Beacon perform better than G-TopFreq on the TaFeng dataset, as the main performance gain is from improvements in exploration prediction. As a consequence, in the Dunnhumby and Instcart datasets, Dream, Beacon, and G-TopFreq achieve similar performance, and the repeat prediction contributes the most to the performance, even when their recommended items are heavily skewed towards the explore items. Also, we observe that CLEA outperforms other explore-biased methods due to the improvements in the repetition performance without sacrificing the exploration performance.

At the same time, it is clear that TIFUKNN, UP-CF@r, Sets2Sets, and DNNTSP outperform explore-biased methods because of the improvements in the repetition performance, even at the detriment of exploration. The repeat items form up the majority of their correct recommendation. Specifically, the repeat recommendation contributes to over 97% of their overall performance in Dunnhumby and Instacart dataset.

An interesting comparison is between Sets2Sets and P-TopFreq, the strong performance gain of Sets2sets on the TaFeng dataset is mainly due to the exploration part, whereas P-TopFreq outperforms it by a large margin on the other two datasets at the same level of repeat ratio, even though the personal frequency information is considered in the Sets2sets model. Apart from the information loss when embedding, we believe this indicates another issue of training-based method, which is that the repetition loss seems to be suppressed by the exploration loss during the training process, which weakens the influence of the frequency information.

Recall that the number of repetition candidates of the user may be smaller than the basket size, which means that there might be empty slots in the recommended basket. From Figure 1, we observe that the empty slots account for
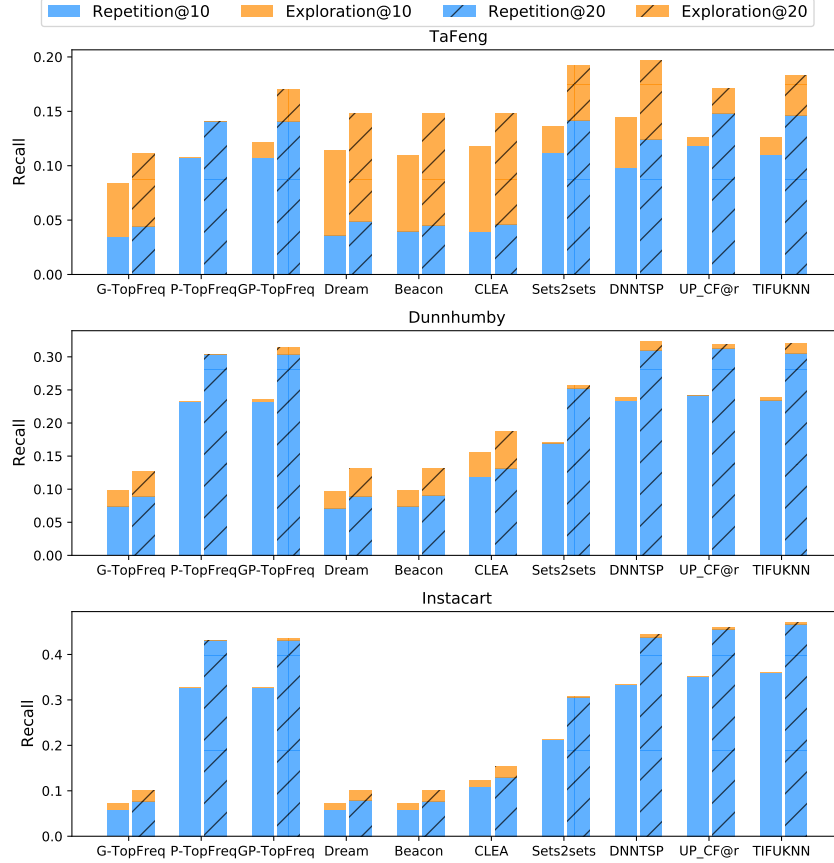
Fig. 2: Performance contribution from repeat and explore recommendation of simple baselines and published state-of-the-art methods.

a significant proportion of exploration slots in many settings. However, existing studies omit this fact when making the comparison with P-TopFreq, leading to an unfair comparison and overestimation of the improvement, as their predictions leverage more slots. For example, Dream, Beacon, and CLEA can beat P-TopFreq, but they are inferior to the GP-TopFreq. TIFUKNN and UP-CF@r model the temporal order of the frequency information, leading to a higher repetition performance than P-TopFreq in general. We can see that even though the contribution of the repetition performance improvement is obvious on the Instacart dataset, it is less meaningful on the other two datasets, where the performance gain is mainly from the exploration part via filling the empty slots. When compared with the proposed GP-TopFreq on Tafeng and Dunnhumby, the improvement is around 3%.

DNNTSP is always among the best-performing methods across three datasets and is able to model exploration more effectively than other repeat-biased methods. What's more, it could also actively recommend novel items, rather than being totally biased towards the repeat recommendation in high exploration

scenarios. However, the improvement is limited due to the relatively high repeat ratios and the huge difficulty gap between repetition and exploration tasks. Specifically, compared with GP-TopFreq, the improvement of DNNTSP w.r.t *Recall*@10 on Dunnhumby and Instacart is merely 1.3% and 1.9% respectively, which is not promising considering the complexity and the training time added by DNNTSP.

Obviously, even though many advanced NBR algorithms learn rich user and/or items representations, the main performance gains stem from the prediction of repeat behavior. Yet, limited progress has been made compared to the simple P-TopFreq and proposed GP-TopFreq baseline methods.

5   Lessons Learned and Summary

From the analysis of our proposed new metrics and the new baselines, we observe that there is a clear difficulty gap and trade-off between the repetition task and the exploration task. Apart from the conventional metrics, the metrics w.r.t repetition and exploration in our paper should be considered as a set of fundamental metrics to understand the performance of NBR methods.

As a rule of thumb, being biased towards the easier repetition task is an important hidden trick favoring the overall NBR performance. However, several training-based methods can not effectively model the repetition behavior due to the embedding information loss or the optimization goal, and they are not able to outperform the simple frequency baseline GP-TopFreq, since they go in the opposite direction. Even though several recent state-of-the-art methods are able to trigger this trick and are skewed towards repetition task unintentionally, e.g. self-links in GNN [19], searching hyper-parameters [4, 7], none of these methods acknowledge this strategy in advance and the improvement is rather limited compared to the complexity and resources introduced by these methods. Specifically, the NBR task boils down to the repetition recommendation task in some cases.

The last layer before the final prediction is often regarded as the "representation" in many training-based neural methods. Neural models claim they learn "better representation" via considering and modeling several factors. These representations are claimed to be better than previous neural methods considering item importance [10], attention [6], noise within the basket [11], cross basket relations [19], etc. Hu et al. [7] argue that RNNs can not capture frequency information, however, these are rather weak or vague conclusions and none of them identify the underlying reasons for the improvement. We argue that without distinguishing between the different components in the baskets and understanding the trade-off between the repeat and the explore items in the recommendation, more neural models with fancy modules will be proposed and claim better representation; while in fact, they might just trigger some hidden tricks which can be easily drawn from some basic analysis.

Even though we are focusing on next basket recommendation, it would also be interesting to contrast the outcomes with an analysis of repetition and exploration behavior in traditional sequential recommendation.

Bibliography

[1] Bai T, Nie JY, Zhao WX, Zhu Y, Du P, Wen JR (2018) An attribute-aware neural attentive model for next basket recommendation. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp 1201–1204

[2] Batmaz Z, Yurekli A, Bilge A, Kaleli C (2019) A review on deep learning for recommender systems: Challenges and remedies. Artificial Intelligence Review 52:1–37

[3] Dacrema MF, Cremonesi P, Jannach D (2019) Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In: Proceedings of the 13th ACM Conference on Recommender Systems, pp 101–109

[4] Faggioli G, Polato M, Aiolli F (2020) Recency aware collaborative filtering for next basket recommendation. In: Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization, pp 80–87

[5] Hao MC, Hsu M, Dayal U, Wei SF, Sprenger T, Holenstein T (2001) Market basket analysis visualization on a spherical surface. In: Visual Data Exploration and Analysis VIII, International Society for Optics and Photonics, SPIE, vol 4302, pp 227 – 233

[6] Hu H, He X (2019) Sets2sets: Learning from sequential sets with neural networks. In: Proceedings of the 25th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp 1491–1499

[7] Hu H, He X, Gao J, Zhang ZL (2020) Modeling personalized item frequency information for next-basket recommendation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 1071–1080

[8] Kang WC, McAuley J (2018) Self-attentive sequential recommendation. In: 2018 IEEE International Conference on Data Mining (ICDM), IEEE, pp 197–206

[9] Krichene W, Rendle S (2020) On sampled metrics for item recommendation. In: Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), ACM, New York, NY, USA, KDD '20, p 1748–1757

[10] Le DT, Lauw HW, Fang Y (2019) Correlation-sensitive next-basket recommendation. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)

[11] Qin Y, Wang P, Li C (2021) The world is binary: Contrastive learning for denoising next basket recommendation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 859–868

[12] Raeder T, Chawla NV (2011) Market basket analysis with networks. Social Network Analysis and Mining 1(2):97–113

[13] Rendle S, Freudenthaler C, Schmidt-Thieme L (2010) Factorizing personalized markov chains for next-basket recommendation. In: Proceedings of the 19th international conference on World wide web, pp 811–820

[14] Sun L, Bai Y, Du B, Liu C, Xiong H, Lv W (2020) Dual sequential network for temporal sets prediction. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 1439–1448

[15] Wang P, Guo J, Lan Y, Xu J, Wan S, Cheng X (2015) Learning hierarchical representation model for next basket recommendation. In: Proceedings of the 38th International ACM SIGIR conference on Research and Development in Information Retrieval, pp 403–412

[16] Wang P, Zhang Y, Niu S, Guo J (2019) Modeling temporal dynamics of users' purchase behaviors for next basket prediction. Journal of Computer Science and Technology 34(6):1230–1240

[17] Wang S, Hu L, Wang Y, Sheng QZ, Orgun M, Cao L (2020) Intention nets: Psychology-inspired user choice behavior modeling for next-basket prediction. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 34, pp 6259–6266

[18] Yu F, Liu Q, Wu S, Wang L, Tan T (2016) A dynamic recurrent model for next basket recommendation. In: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, pp 729–732

[19] Yu L, Sun L, Du B, Liu C, Xiong H, Lv W (2020) Predicting temporal sets with deep neural networks. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp 1083–1091