Personalizing diversity versus accuracy in session-based recommender systems

Alireza Gharahighehi · Celine Vens

Received: date / Accepted: date

Abstract One of the most important concerns about recommender systems is the filter bubble phenomenon. While recommender systems try to personalize information, they tighten the filter bubble around the users and deprive them of a wide range of content. To overcome this problem one can diversify the personalized recommendation list. A diversified list usually presents a broader content to the user. Session-based recommender systems are types of recommenders in which only the current session of the user is available and therefore they should recommend the next item given the items in the current session. While diversifying conventional recommender systems has been well assessed in the literature, it has gained less attention in session-based recommenders. Diversity and accuracy usually have a negative correlation, i.e., by improving one the other one will be declined. In this study we propose diversity and accuracy enhancing approaches based on sequential rule mining and session-based k nearest neighbor methods. Finally, we propose a performance balancing approach that improves both the diversity and accuracy of these session-based recommender systems. We demonstrate the performance of the proposed methods on four music recommender datasets.

Keywords Session-based recommenders · Diversity · Session-based k nearest neighbor · Sequential rule mining · Filter bubble phenomenon

1 Introduction

Electronic services are growing rapidly and people tend more and more to shop, read, listen and communicate online. These activities generate interactions between users and items in different contexts which can be used to personalize user

A. Gharahighehi ✉ · C. Vens

Itec, imec research group at KU Leuven, Kortrijk, Belgium

KU Leuven, Campus KULAK, Department of Public Health and Primary Care, Kortrijk, Belgium

E-mail: {alireza.gharahighehi, celine.vens}@kuleuven.be

experience in online environments. Online service providers use recommender systems (RSs) to model user preferences based on previous user interactions and to recommend services to fulfill user needs.

RSs are classically categorized in two broad categories: Content-based filtering (CB) and Collaborative filtering (CF). While CB only uses a user profile and content meta-data to recommend items, CF considers interactions of other users in its predictions. In some domains users do not have a profile in the system and therefore the only relevant information is the current session. A session starts when a user enters a website, continues when she navigates through it and finishes when she leaves it. Therefore personalization can only depend on the events of the current session. In this case Session-based Recommender Systems (SBRs) are used to predict the next item given the previous items in the current session. SBRs are applied in many domains such as music and video streaming websites, news aggregators and web-shops. For instance, in a music streaming website a SBR can be used to recommend the next song given the songs that an anonymous user has already listened to.

In many applications there is side information associated with the recommendable items. This information can be used to boost accuracy in RSs. For instance a CF recommender system can be hybridized with a CB method to boost accuracy. Although this helps RSs to recommend more relevant items, it escalates the filter bubble phenomenon. This phenomenon occurs when RSs try to filter relevant information and recommend it to the users, but, while doing so, deprive them of diverse set of content. In this situation the user may receive several recommended items with relatively similar content. For instance in a news aggregator website an RS may narrow the range of recommended news topics by only focusing on news that is relevant according to the user histories [1]. This may help accuracy but it tightens the filter bubbles around users. Overcoming the filter bubble phenomenon is an important challenge in RSs [2]. Instead of considering only the relevance, one should consider also the diversity in the recommendation list. A diversified recommendation list is less probable to contain very similar and sometimes redundant content. For example in a music recommendation task diversification provides songs in different genres and from several artists to widen the content bubbles around users. Since the only useful information in SBRs is the sequence of events in the current session, the filter bubble problem is even more concerned as RSs should retrieve relevant items based on a limited number of interactions. Diversity should be considered in the recommendation list to avoid recommending redundant content to the user, however, to the best of our knowledge, current SBRs algorithms only focus on providing relevant recommendations, ignoring diversity. Empirical studies investigating the best strategy to include diversity into SBRs are mostly lacking in the literature.

Although diversification mitigates the filter bubble phenomenon, it usually degrades accuracy. Therefore the challenging task is to find a balance for these two performance dimensions. An important property of the sessions (users) in the platform is that they are different in terms of profile diversity: some users are focused users, who are interested in specific content, and some users prefer to interact with a broader range of content. Therefore, to balance the two performance measures and to increase the overall performance, we propose to personalize the diversity versus accuracy trade-off of recommendation lists. For instance in a music streaming platform the recommender can adapt the diversity and accuracy

of its recommendations based on the genre diversity of previous songs in the active session. If the anonymous user is interested in only one genre (focused user), the recommender should provide more accurate recommendations from the same genre. On the other hand, if the user has interacted with a wider range of genres in the session, the recommender should provide more diverse recommendations to cover the broad preferences of the user.

The contribution of this paper is threefold: first, we introduce diversity in SBRs to burst the filter bubbles around anonymous users. The diversity-aware SBRs provide a wider range of content in their recommendations. Second, we hybridize SBRs with items metadata to boost their accuracy. Finally, we demonstrate the possibility of enhancing the overall performance of SBRs in both accuracy and diversity by personalizing the diversity vs accuracy balance of recommendations. Two state-of-art methods in SBR are extended with diversity and accuracy enhancing strategies. These strategies are compared w.r.t. diversity, accuracy and computational time, using four real-life datasets from the music recommendation domain.

In the following, related studies about diversity definition and measurement, SBRs, diversity enhancing approaches and diversity-aware SBRs are reviewed (Section 2). Then in Section 3 the proposed strategies to enhance diversity and accuracy of two SBRs are explained. In Section 4 four music datasets are described and the experimental settings in designing and testing the proposed models are discussed. Next the performance of several SBR baselines is compared and the results of applying different diversity and accuracy enhancing strategies based on several performance measures are assessed in Section 5 and discussed in Section 6, before concluding in Section 7.

2 Related Work

2.1 Diversity definition and measurement

Diversification was first concerned in the information retrieval (IR) community to disambiguate the user’s query [3]. A classic example is the search term *jaguar* which may refer to the animal, the car or the guitar. If the retrieved document list is more diverse, the chance of covering user’s real intent in the retrieved list would be higher. Therefore considering and optimizing diversity in a ranked list of retrieved documents using document features has attracted a great deal of attention in the IR community [4].

In recommendation systems diversity is a way to address the filter bubble phenomenon. For instance, in a movie streaming website, the diversity of a recommended list that contains action, drama and comedy movies, has higher diversity than a recommended list containing only action movies. The diversity of a recommendation list is commonly defined as the average pairwise distance between items in the list [5]. This diversity measure, which is also called intra-list diversity (ILD), shows to what extent the user will receive a diverse range of items in the recommended list. Calculating pairwise distance between items depends on feature representations of items and the distance measure. For instance features can be text-based features [6, 7], genres [8, 9], taxonomy [10, 11], rating vector [12–14], neighborhood of items in item-based CF or neighborhood of users in user-based

CF [15] and latent features in model-based CF [16–19]. Distance measures can be taxonomy-based [10], cosine, Jaccard, hamming distance and Pearson Correlation [3].

2.2 Session-based Recommender Systems

SBRs are common in domains in which a user’s profile is missing and the only available information is the current session. The main task in this setting is predicting the next item given the sequence of events in the current session. SBRs can be categorized to pattern/rule-based, sequential pattern-based, nearest neighbors-based, Markov chain-based, factorization-based and neural network-based methods [20, 21]. Based on a comprehensive evaluation on state-of-art SBRs using several music, e-commerce and news datasets, Ludewig and Jannach [20] concluded that simple methods such as sequential rule mining and session-based neighborhood methods surprisingly have better performance w.r.t. accuracy measures compared to more complex methods. These results are also supported by other studies [22–28].

Although neighborhood methods are relatively simple, they perform surprisingly well. The simplest version of neighborhood methods, i.e, item-based KNN [29] (IKNN), has been frequently used as a baseline in the SBR literature. In this method, only the last item of the current session is used to form a neighborhood. Session-based KNN (SKNN) [23] methods are a generalization of IKNN that use the whole sequence of events in the current session to form the neighborhood. There are some variants of SKNN that put more emphasis on recency of events in the session (Vector Multiplication SKNN), position of events in the session (Sequential SKNN) and also a more restrictive position aware version (Sequential Filter SKNN) [20].

There are some SBRs that use artificial neural networks to model sequences in sessions. Hidasi et al. [30] proposed a Recurrent Neural Network (RNN) model using Gated Recurrent Units (GRU) to recommend the next item in the session. This model was extended by changing the loss function and sampling approach that shows better performance compared to the previous model [31]. Li et al. [32] introduced NARM method that uses an RNN-based network with attention mechanism to generate recommendations. This network models the sequential information and the main purpose of the session to form recommendations. Wang et al. [33] proposed an attention-based transaction embedding model (ATEM) for next item recommendation task that learns an attentive context embedding to emphasize more on relevant items in the user session. Sheu and Li [34] used a context-aware graph embedding framework for a news SBR. This framework constructs a knowledge graph based on the news entities to improve article embeddings generated by graph neural networks. Liu and Zheng [35] proposed TailNet model which is an SBR that gives more attention to the long-tail items in order to make the recommendations more serendipitous. Wu et al. [36] introduced a SBR with graph neural network (SR-GNN). They modeled the session sequences as graph structured data. Using this structure, the model can represent the current interest and the global preference of the active session.

2.3 Enhancing Diversity in Recommendations

Generally speaking there are two approaches to enhance diversity of RSs: Re-ranking approach and diversity modeling.

Re-ranking approach: Re-ranking is a post-filtering approach which re-ranks an initial list generated by a baseline method to achieve better performance based on some additional objectives. A greedy approach using Maximal Marginal Relevance [37] is the most common form of re-ranking in the literature. Different objectives and distance functions are used to greedily re-rank the initial list [5, 12, 10, 17, 38, 39]. There are some other studies that use constrained optimization [40, 41] and multi-objective optimization [14] instead of a greedy approach to re-rank the initial list.

Diversity modeling: The other way of enhancing diversity of a recommendation list is to consider a diversity component in the main algorithm. Said et al. [42] proposed the k-furthest neighbors method to generate a more diverse recommendation list compared to the k-nearest neighbors method. Shi et al. [19] introduced Latent Factor Portfolio model in which portfolio theory [43] and matrix factorization were used to diversify the result. Some other studies manipulated the original loss functions of algorithms to enhance diversity. Su et al [16] proposed the set-based Bayesian Personalized Ranking (BPR) as an extension of BPR [44] in which, instead of using pairs of positive and negative items, they used pairs of positive and negative item *sets* in the BPR loss function. They considered the diversity of these item sets in their predicted ratings. Hurley [45] adds the distance of two sampled items in the loss functions of the *RankALS* [46] and *RankSGD* [47] optimization approaches to enhance diversity. In another study [48] some regularisation terms are proposed to be added to the *RankALS* and *RankSGD* approaches in order to generate a more diverse recommendation list. Wang et al. [49] proposed the mixture-channel purpose routing networks (MCPRNs) that can cover multiple purposes of the user in the session by recommending a more diverse recommendation list. They showed that their model has better accuracy and diversity compared to the baselines.

2.4 Boosting Accuracy and Diversity in Session-based Recommender Systems

There are some studies that hybridized SBRs with item features to boost their accuracy. Tan et al. [50] improved the performance of Recurrent Neural Network (RNN) based recommenders by using item features. Moreira et al. [51] studied the effect of content on a neural SBR and they showed that hybridizing a SBR with item features boosts the accuracy measures. Wang et al. [52] proposed the neural network-based comprehensive transaction embedding model (NTEM) that uses a shallow and wide network. This network uses the explicit relevance between items from the session data and the implicit relevance from features of items to recommend the most relevant next items and consequently to boost accuracy.

To the best of our knowledge, diversity enhancing approaches in SBRs are very limited in the literature. Just like the concept of diversity that was originated from

the IR community, *xQuAD* was introduced in the IR community [53] and later applied in recommendation systems [54]. *xQuAD* is a greedy re-ranking approach that uses an intent-oriented diversity component in the re-ranking objective function. Anelli et al. [55] introduced a variant of *xQuAD* in which they changed the diversity component in the greedy optimization approach of *xQuAD* to make it time and session aware. Esmeli et al. [56] introduced a variant of IKNN by using not only the last item but also the session context in score calculations. They also used the dissimilarity of recommendable items based on the whole session items in the final score function. At the end, instead of evaluating the proposed method in terms of diversity, they showed that, using this approach, they can improve the precision and recall in an e-commerce dataset.

3 Enhancing Diversity and Accuracy in Session-based Recommender Systems

In this section we propose diversity and accuracy enhancing strategies for two SBRSSs. We used the result of the comprehensive evaluation of [20] (which is confirmed in our comparison of baselines in Section 5) to select SBRSS methods, namely sequential rule mining (SR) and SKNN. These methods are selected as they have competitive performances with low computational costs. We propose three diversity or accuracy boosting strategies: *weight of candidate* (Section 3.1 and Section 3.2), *neighbor selection* (Section 3.2) and *weight of neighbors* (Section 3.2). In addition, we also propose a balancing approach that improves both the diversity and accuracy (Section 3.3). In Section 3.1 only *weight of candidate* strategy used to enhance the diversity or accuracy of SR while in Section 3.2 all the three proposed strategies are applied to boost the diversity or accuracy of SKNN. Finally, in Section 3.3 we explain how one can enhance the overall diversity and accuracy at the same time in these two SBRSSs, by adapting the strategy based on the diversity of individual sessions. As the greedy re-ranking approach is commonly used in literature to enhance diversity, we consider it in our experiments as a baseline.

3.1 Enhancing Diversity and Accuracy in Sequential Rule mining

Sequential rule mining (SR) identifies events that frequently co-occurred in the past to recommend the next item considering the previous items visited by the user. This method is a variant of association rule mining (AR) in which the order of frequent items in the rules is important and the further apart two frequent items are the less weight is assigned to these frequent items. Using this approach and the notation used by [22], the predicted score for a candidate item and the current session is:

$$\hat{r}_{SR}(i, s) = 1_R(r_{j,i}) \times w_{j,i} \quad (1)$$

In Equation 1, $\hat{r}_{SR}(i, s)$ is the predicted score for candidate item i and session s , R is the set of rules of item co-occurrences in previous sessions, j is the last item of s , $r_{j,i}$ is the rule for item j and i , $1_R(r_{j,i})$ is an indicator function that verifies whether there is a rule for item j and i in R and $w_{j,i}$ is the weight of the rule. To calculate $w_{j,i}$, instead of only counting the number of co-occurrences of items in

previous sessions as the weight, SR penalizes the weight based on the distance of these items.

We propose the *weight of candidate* strategy to enhance the accuracy or diversity of SR. To enhance diversity in SR one can assign more weights to candidate items that add more diversity to the current session. This weight ($\delta_{i,s}$) can be computed using Equation 2. An item that has more dissimilarity with the current session is more likely to make the final recommendation list more diverse. Using this weight, diversity-aware SR can be formulated in Equation 3.

$$\delta_{i,s} = \frac{1}{|s|} \sum_{j \in s} \text{dist}_c(i, j) \quad (2)$$

$$\hat{r}_{SR}(i, s) = 1_R(r_{j,i}) \times w_{j,i} \times \delta_{i,s} \quad (3)$$

Using a fairly similar approach one can increase accuracy of SR by replacing $\text{dist}_c(i, j)$ with $\text{sim}_c(i, j)$ in Equation 2. In this setting, SR assigns more weight to an item that is more similar (content-wise) to the current session. $\text{sim}_c(i, j)$ is the content similarity and $\text{dist}_c(i, j)$ is the content dissimilarity of item i and j . $\text{sim}_c(i, j)$ and $\text{dist}_c(i, j)$ can be calculated using a similarity measure such as the cosine similarity. Obviously only one of these approaches (enhancing diversity or accuracy) is possible at the same time.

3.2 Enhancing Diversity and Accuracy in Session-based k-Nearest Neighbors

SKNN is a memory-based method that uses all of the items of the current session to find the k nearest neighbor sessions. To predict the score of a candidate item, SKNN uses the session similarity of neighbor sessions with the current session. Here the session similarity ($\text{sim}_s(s, n)$) is the similarity between binary vectors of sessions over the item space. This predicted score can be calculated using Equation 4:

$$\hat{r}_{SKNN}(i, s) = \sum_{n \in N_s} \text{sim}_s(s, n) \cdot 1_n(i) \quad (4)$$

In which i is the candidate item, s is the current session for which we want to predict the next item, N_s is the set of current session neighbors, $\text{sim}_s(s, n)$ is the similarity of current session s and one of its neighbors n and $1_n(i)$ is an indicator function to check whether item i exists in session n . $\text{sim}_s(s, n)$ can be calculated using a similarity measure such as cosine, Jaccard or Pearson correlation.

We propose three strategies that can be added to SKNN methods to enhance either diversity or accuracy: *neighbor selection*, *weight of neighbors* and *weight of candidate* item. The predicted score for a diversity or accuracy boosting SKNN can be calculated using Equation 5:

$$\hat{r}_{SKNN}(i, s) = \sum_{n \in N'_s} w_n(s, n) \cdot \delta(i, s) \cdot 1_n(i) \quad (5)$$

Neighbor selection: One way to recommend a more diverse recommendation list is to use neighbors that are more diverse. Therefore in Equation 5, instead of using similarity as the only criterion to form neighborhood of s , neighbors (N'_s) will be selected based on their similarity to the current session and their diversity, i.e., instead of $sim_s(s, n)$, $sim_s(s, n) \times diversity(n)$ is used. $diversity(n)$ can be computed using the average intra-list distance (ILD) measure (Equation 6).

$$diversity(n) = \frac{\sum_{i \in n} \sum_{j \in n \setminus \{i\}} dist_c(i, j)}{|n|(|n| - 1)} \quad (6)$$

Weight of neighbors: Higher weights for more diverse neighbors will promote items that make the recommendation list more diverse. To include this in equation 5 $w_n(s, n) = sim_s(s, n) \times diversity(n)$ is used.

Weight of candidate: The candidate item itself affects the diversity of the current session. This idea can be considered in Equation 5 by including $\delta(i, s)$ that is the additional contribution of item i on the current session diversity and can be calculated using Equation 2, in which $dist(i, j)$ is the distance between item i and j and can be computed using the complement of a similarity measure.

To boost accuracy, instead of $diversity(n)$ in the *neighbor selection* and the *weight of neighbors* strategies, one can use $sim_c(s, n)$ which is the content similarity of sessions s and n . To calculate the content similarity of two sessions we aggregate the content of items in each session and calculate the content similarity between these two aggregated vectors. For the third strategy, i.e., *weight of candidate*, $sim_c(i, j)$ should be substituted with $dist(i, j)$.

3.3 Balancing Diversity and Accuracy

Following the aforementioned diversity and accuracy boosting approaches, enhancing one of them degrades the other one. The desired situation is to enhance both accuracy and diversity of recommendations at the same time. This is achievable by selecting approaches based on the current session diversity. If the current session has low diversity, it implies that the user prefers more focused items and if the current session has high diversity, it implies that the user prefers broad content. Using this argument, one can select a strategy to enhance diversity or accuracy based on the diversity of the current session. The threshold, based on which one can decide to select between boosting diversity or accuracy, is a hyper-parameter and is different across datasets. To set this threshold, one can use the diversities of previous sessions and then set the threshold as the x^{th} percentile of the diversities of these sessions. If the diversity of the current session (test session) is higher than the threshold, the diversification approach will be applied, otherwise the accuracy boosting approach will be selected.

3.4 Greedy re-ranking approach

A greedy re-ranking approach is frequently used in the literature because of its simplicity and flexibility. It can be applied on the output of any baseline RS to

greedily maximize an arbitrary objective function. This function ($f(i, R)$) can be adapted to enhance several objectives. In this paper the objectives are relevance and diversity. Therefore the objective function is defined as:

$$f(i, R) = \lambda \cdot \text{relevance}(i) + (1 - \lambda) \cdot \text{diversity}(i, R) \quad (7)$$

In Equation 7, $\text{relevance}(i)$ is the relevance score of item i , $\text{diversity}(i, R)$ is the average pairwise distance of item i with items in list R and λ is a balancing parameter. A greedy re-ranking algorithm starts with an initial recommendation list (S) generated from a baseline recommender system. To maximize the objective function ($f(i, R)$), the algorithm greedily moves items from S to R until it reaches the desired length of R .

4 Datasets and Experiment set-up

One of the applications of session-based recommenders is music recommendation to recommend the next song given the current session. To evaluate the diversity and accuracy enhancing approaches, four music datasets are used in this study which are described in Table 1. For each session the sequence of songs in the session with their artists are given. Artists of items are used to compute the diversity of a recommendation list. The 30music [57], lastfm¹ [58] and Nowplaying [59] datasets are based on listening histories and the Aotm dataset [60] contains playlists which are considered as sessions in this study. For all of these datasets short sessions ,i.e., sessions with less than 4 items and low frequent items, i.e., items with support of less than 2 are excluded from the datasets.

The task of SBRs is to predict the next items in the test sessions. To split the datasets to train and test sessions, the approach by [20] is used. In this approach the datasets are divided into five slices with same duration and then the sessions in the last day of each slice are considered as test sessions. In each of these test sessions, we hide the last two items and compare the predictions with these hidden items. To make predictions for the test sessions, only sessions from the training data within the same slice is used. Final performance measures are computed by averaging the performance measures over the five slices. The average performance in all of these test sessions are computed and reported in the next section.

We consider three different performance measures for evaluation: precision, recall and diversity. Precision and recall are standard formation retrieval accuracy measures that assess the model in predicting the hidden items in the current session. Diversity returns the average intra-list distance of the recommendation list and it shows to what extent the filter bubble phenomenon exists in the recommendation list.

There are some hyper-parameters that should be tuned in our experiments. We consider the sessions in the last day of training set in each slice as the validation set to evaluate hyper-parameter values. Then the validation set is also used in the training phase. In SR, the only hyper-parameter is the maximum number of events between two items in a rule. In SKNN the number of neighbors is the hyper-parameter that should be tuned². The threshold for balancing diversity and

¹ www.last.fm

² For the SR hyper-parameter we used values in [1,15] and for SKNN values in [20,150].

Table 1: Datasets descriptions

	30music	Aotm	lastfm	Nowplaying
# Sessions	169,576	83,619	364,432	138,092
# Items	449,037	138,485	445,106	143,627
# Artists	60,949	20,149	61,481	22,861
Timespan	2014-01 to 2015-01	2016-01 to 2016-12	2006-11 to 2008-08	2014-01 to 2015-06

Table 2: Hyper-parameters

	30music	Aotm	lastfm	Nowplaying
Max # events between items in rules	10	5	10	10
# Neighbors in SKNN	100	100	100	50
Balancing threshold for SR	50	20	50	80
Balancing threshold for SKNN	50	20	50	40

accuracy is another hyper-parameter that should be tuned for each dataset. To tune this hyper-parameter we first select the thresholds in which both *precision@10* and *diversity@10* are higher than the original method (i.e. SR and SKNN without adaptation) in the validation set. Then between these selected thresholds we select one that maximizes the combination of relative changes in accuracy and diversity, still using the validation set (Equation 8). In Equation 8, $s(x)$ is the score that we want to maximize, $P@10_{(x)}$ and $D@10_{(x)}$ are precision and diversity of the model for a recommendation list of length 10 and $P@10_{original}$ and $D@10_{original}$ are precision and diversity of the original method. Here we assume that accuracy and diversity have equal weights but one can set different weights for them across various applications. For the same reason, we set λ in the re-ranking approach to 0.5 in our experiments. It is not tuned since it depends on the application. The final hyper-parameters, which are used in the experiments, are presented in Table 2.

$$s(x) = \frac{(P@10_{(x)} - P@10_{original})}{P@10_{original}} + \frac{(D@10_{(x)} - D@10_{original})}{D@10_{original}} \quad (8)$$

5 Results

In this section we show the performance of our base methods (SR and SKNN) and the proposed approaches on the four music recommender datasets. In order to interpret their performance, we first compare their accuracy and runtime to that of several baselines including rule-based, neighborhood-based, Markov, factorization-based and neural network-based methods³. The assessed baselines are as follows:

- **SKNN**: recommends next items for a session based on session neighbors [23].
- **IKNN**: recommends items similar to the last item of the session [29].
- **AR**: uses the co-occurrence of pairs of items in previous sessions to recommend next items.

³ The hyperparameters of these baselines are tuned using a validation set.

- **SR**: uses the co-occurrence of pairs of items and their positions in the session to recommend next items.
- **Markov Chains (MC)**: recommends next items using rules based on first-order Markov Chain.
- **Session-based Matrix Factorization (SMF)**: a factorization based SBRS that considers sequential dynamics [20].
- **Factorized Personalized Markov Chains (FPMC)**: uses user-item matrix factorization and Markov chains in a three dimensional tensor factorization approach [61].
- **BPR-MF**: a traditional learning to rank matrix completion method that uses *BPR* loss function to learn the user and item low-rank matrices [44].
- **GRU4REC**: an RNN based SBRS that uses the $TOP1_{max}$ loss function to model user sequences [31].
- **NARM**: an RNN-based SBRS that uses attention mechanism to generate recommendation [32].
- **SR-GNN**: uses graph structured data to model session sequences and recommends next items based on global preference and the current focus of the session [36].

The results are shown in Table 3. We can observe that simple methods such as SKNN and SR have very competitive performance with very low computational costs compared to the more complicated methods. As SBRSs needs to deal with a very large set of items in real time, complicated methods such as neural network-based ones are often too computationally demanding and therefore less efficient for online recommendation compared to these simple methods. Therefore these simple methods can be highly effective and fast in online recommendations. These results are also supported by other studies [20, 22–24].

The results of applying the balancing approach introduced in Section 3.3 using four music datasets are shown in Tables 4 (*30music* dataset), 5 (*Aotm* dataset), 6 (*Lastfm* dataset) and 7 (*Nowplaying* dataset). The results of individual strategies are also included for completeness. As introduced in Section 3 there are three strategies to boost either accuracy or diversity. In these tables “C” refers to adding *weight of candidate*, “S” refers to considering *weights of neighbors*, “N” refers to adding new *neighbor selection* criteria strategy. As mentioned in Section 3.2 these

Table 3: Comparison of SBRS baselines based on accuracy and runtime for 30music dataset

	Precision@10	Recall@10	Runtime(s)
SKNN	0.0502	0.2509	4.97
IKNN	0.0394	0.1970	2244.97
AR	0.0375	0.1876	14.20
SR	0.0513	0.2563	8.34
MC	0.0278	0.1393	3.92
SMF	0.0317	0.1585	6120.82
FPMC	0.0069	0.0343	1007.61
BPR-MF	0.0068	0.0338	1030.51
GRU4REC	0.0495	0.2479	1486.91
NARM	0.0327	0.1637	51289.53
SR-GNN	0.0523	0.2617	2030.20

strategies can be used to either boost diversity or accuracy. “D” refers to the diversification approaches, “A” refers to the accuracy boosting approaches and “DA” refers to the balancing approach. “DA” strategy in SKNN includes all three strategies, i.e., “CSN”. Finally “_Re” refers to the re-ranking approach.

As is shown in Table 4 (*30music* dataset), all of the proposed diversity enhancing strategies indeed improve diversity and all of the proposed accuracy enhancing strategies improve accuracy compared to the original method. In this dataset the SR family produces more accurate and more diverse recommendations. For the SKNN method, $SKNN(D : S)$ is the most effective strategy in enhancing diversity and $SKNN(A : C)$ is the most effective strategy in enhancing accuracy from the three (C , N and S) individual components. For enhancing diversity, combining the three components (CSN) results in the best diversity value, but for enhancing accuracy, the combination has similar performance with C strategy. The balancing approach successfully enhances both the accuracy and diversity of SKNN and SR. The re-ranking approach has a big impact on diversity but negative impact on accuracy measures. It is a simple approach but it needs more time for post-filtering the recommendation list.

Table 5 (*Aotm* dataset) confirms the ability of diversity enhancing strategies to improve the diversity and to some extent accuracy enhancing strategies to improve accuracy of the original methods. Interestingly $SKNN(D : C)$ can improve diversity without deteriorating accuracy compared to the original method. The SKNN family produces higher accuracy compared to the SR family. In the SKNN family methods, S is again the most effective strategy in enhancing diversity and C is the most effective strategy in enhancing accuracy. As for the previous dataset, the balancing approach in both methods (SR and SKNN) can improve accuracy and diversity compared to the original method. The re-ranking approach again improves the diversity at the cost of reduced accuracy.

According to Table 6 (*Lastfm* dataset) diversity enhancing strategies can improve diversity measure compare to the original methods. The SR family has

Table 4: Results for 30music dataset. Bold values indicate improvement w.r.t. the base method (SR and SKNN, respectively).

	Precision@10	Recall@10	Diversity@10
SR	0.0513	0.2563	0.7281
SR(DA)	0.0517	0.2584	0.7321
SR(D: C)	0.0489	0.2443	0.7980
SR(A: C)	0.0528	0.2641	0.6735
SR_Re	0.0483	0.2413	0.9068
SKNN	0.0502	0.2509	0.5023
SKNN(DA)	0.0505	0.2527	0.6020
SKNN(D: CSN)	0.0380	0.1898	0.7317
SKNN(D: S)	0.0446	0.2229	0.6788
SKNN(D: C)	0.0459	0.2297	0.5566
SKNN(D: N)	0.0502	0.2510	0.5033
SKNN(A: CSN)	0.0547	0.2737	0.4329
SKNN(A: C)	0.0547	0.2736	0.4388
SKNN(A: S)	0.0505	0.2527	0.4841
SKNN(A: N)	0.0502	0.2509	0.5022
SKNN_Re	0.0458	0.2288	0.6888

Table 5: Results for Aotm dataset. Bold values indicate improvement w.r.t. the base method (SR and SKNN, respectively).

	Precision@10	Recall@10	Diversity@10
SR	0.0027	0.0137	0.9711
SR(DA)	0.0029	0.0144	0.9750
SR(D: C)	0.0025	0.0126	0.9780
SR(A: C)	0.0027	0.0137	0.9685
SR_Re	0.0025	0.0125	0.9887
SKNN	0.0036	0.0181	0.9017
SKNN(DA)	0.0040	0.0202	0.9326
SKNN(D: CSN)	0.0030	0.0151	0.9783
SKNN(D: S)	0.0027	0.0137	0.9775
SKNN(D: C)	0.0037	0.0187	0.9358
SKNN(D: N)	0.0036	0.0181	0.9018
SKNN(A: CSN)	0.0038	0.0189	0.8721
SKNN(A: C)	0.0039	0.0193	0.8845
SKNN(A: S)	0.0038	0.0192	0.7962
SKNN(A: N)	0.0036	0.0181	0.9010
SKNN_Re	0.0031	0.0157	0.9695

Table 6: Results for lastfm dataset. Bold values indicate improvement w.r.t. the base method (SR and SKNN, respectively).

	Precision@10	Recall@10	Diversity@10
SR	0.0667	0.3335	0.6364
SR(DA)	0.0672	0.3360	0.6400
SR(D: C)	0.0646	0.3228	0.7104
SR(A: C)	0.0679	0.3395	0.5839
SR_Re	0.0615	0.3077	0.8855
SKNN	0.0604	0.3018	0.4255
SKNN(DA)	0.0615	0.3074	0.5287
SKNN(D: CSN)	0.0487	0.2433	0.6627
SKNN(D: S)	0.0558	0.2789	0.5796
SKNN(D: C)	0.0560	0.2798	0.4875
SKNN(D: N)	0.0603	0.3015	0.4288
SKNN(A: CSN)	0.0642	0.3208	0.3691
SKNN(A: I)	0.0647	0.3233	0.3611
SKNN(A: S)	0.0602	0.3011	0.4290
SKNN(A: N)	0.0602	0.3011	0.4288
SKNN_Re	0.0548	0.2742	0.6555

higher accuracy compared to the SKNN. In the *SKNN* family methods, *SKNN(D: S)* and *SKNN(A: C)* are the most effective strategies from the three (*C*, *N* and *D*) individual components in enhancing diversity and accuracy. The balancing approach again is able to enhance accuracy and diversity compared to the original SR and SKNN. As before, the re-ranking approach enhances diversity but it degrades the accuracy measures.

As Table 7 (*Nowplaying* dataset) shows, all of the proposed diversity boosting approaches improve diversity measure. The *SKNN* family produces more accurate recommendations but the *SR* family has higher diversity. In the *SKNN* family, *S* is the most effective strategy in enhancing diversity and *C* is the most effective

Table 7: Results for nowplaying dataset. Bold values indicate improvement w.r.t. the base method (SR and SKNN, respectively).

	Precision@10	Recall@10	Diversity@10
SR	0.0261	0.1304	0.8582
SR(DA)	0.0272	0.1361	0.8275
SR(D: C)	0.0253	0.1263	0.8906
SR(A: C)	0.0268	0.1341	0.8376
SR_Re	0.0246	0.1232	0.9399
SKNN	0.0299	0.1496	0.7724
SKNN(DA)	0.0310	0.1549	0.7922
SKNN(D: CSN)	0.0240	0.1199	0.9164
SKNN(D: S)	0.0252	0.1261	0.8802
SKNN(D: C)	0.0282	0.1410	0.8290
SKNN(D: N)	0.0299	0.1493	0.7740
SKNN(A: CSN)	0.0313	0.1564	0.7180
SKNN(A: C)	0.0311	0.1554	0.7298
SKNN(A: S)	0.0299	0.1496	0.7235
SKNN(A: N)	0.0300	0.1499	0.7718
SKNN_Re	0.0260	0.1302	0.9068

strategy in enhancing accuracy. For enhancing diversity and accuracy in *SKNN*, combining the three components (*CSN*) results in the best diversity and accuracy respectively. The *SKNN* balancing approach is able to generate more accurate and more diverse recommendation compared to the original *SKNN*, but this is not true for SR. *SR(DA)* produces more accurate recommendations but with less diversity. For the re-ranking approach, the same conclusions as before hold.

The balancing (DA) and re-ranking (Re) approaches have different effects on the average time for generating a recommendation list. The re-ranking approach needs on average 55% more time compared to the original approaches to generate a final recommendation list. On the other hand, the balancing approaches do not have considerable effect (around 6%) on the average time of generating a recommendation list. We conclude that the balancing approaches improve the performance w.r.t. diversity and accuracy, with only a very small impact on computational cost.

6 Discussion

As is shown in the previous section, *SKNN* and *SR* have promising performance w.r.t accuracy and computational cost compared to the other complicated baselines. Low computational cost is crucial in SBRs as these RSs should be highly agile in processing the new coming sessions and recommending online. Moreover, these models are flexible in boosting accuracy and diversity. Diversification is essential in RSs to widen the range of provided information/items to avoid redundancy in recommendations and burst the filter bubbles.

The results in Section 5 show that the diversity enhancing strategies and accuracy enhancing strategies (mainly *A : C*) in all four music datasets indeed enhance the corresponding performance measure but in most cases these strategies also slightly deteriorate the other measure. Diversity and accuracy boosting

approaches are able to improve performance while their runtimes are almost the same as the original models. On the other hand, the re-ranking approach has a considerable impact on diversity but its runtime is much higher than the original model and it degrades accuracy. This longer runtime is more concerned in SBRs than in other RSs as they only depend on the current session. In other RSs, re-ranked lists can be generated based on user profiles and stored offline but SBRs should use the current sequence of visited items to generate recommendation lists online.

SR is a simple method and has shorter runtime compared to SKNN. It is only based on frequent item pairs in training sessions and therefore it is not too flexible in adding diversity or accuracy enhancing components. On the other hand, SKNN is based on neighbor sessions and therefore there are more opportunities to add these components. The *weight of neighbors* strategy ($D : S$) has the highest impact on diversity and *weight of candidate* strategy ($A : C$) has the highest impact on accuracy in all four music datasets compared to the three (C , S and N) individual components.

The goal of balancing approach (DA) is to personalize the level of diversification based on the current session diversity. The focused users prefer items that are more related to their history and users with broad interests would prefer a wider range of items in their recommendation lists. The results in Section 5 confirm that the balancing approach improves both accuracy and diversity compared to the original methods. The improvement over accuracy and diversity of the original methods indicates that the balancing approach can enhance the overall performance of the platform across all users.

In this study we used four music datasets in which only artists of items are given as the item meta-data. The proposed diversity and accuracy enhancing approaches in this study can be applied with other types of meta-data such as genres, tags and lyrics. Then a suitable similarity measure should be selected based on the type of meta-data to calculate similarities or dissimilarities between items. The proposed strategies and approaches can also be applied to other domains where the item meta-data is given.

7 Conclusion

The filter bubble phenomenon is a pitfall in personalized environments. It limits the user experience to a small part of available content. Diversification is a way to address the filter bubble problem. While diversification is well-studied in general Recommendation Systems (RSs), it has received far less attention in the context of Session-based Recommendation Systems (SBRs). The aim of this study was to enhance diversity in SBRs while avoiding drop in accuracy. Two SBRs were assessed in this study, namely Sequential Rule mining (SR) and Session-based k Nearest Neighbor (SKNN). To enhance diversity (resp., accuracy) in SR we considered more weights for items that have higher impact on the diversity (resp., content similarity) of the current session. SKNN is a neighborhood method that uses sessions with similar items to the current session to predict the next item. Three strategies are introduced to enhance diversity and accuracy of SKNN: *neighbor selection*, *weight of neighbors* and *weight of candidate*.

In this article, we empirically compared all these diversity and accuracy enhancing methods on four music recommendation datasets. Our results showed that these performance enhancing strategies indeed improve diversity or accuracy but deteriorate the other one. To decide upon the diversification or accuracy boosting approaches, we proposed to use the current session diversity, in order to personalize the adaptation. If diversity of the current session is higher than a threshold (a hyper-parameter), the corresponding user is interested in broad content and diversification is more effective. Otherwise, the user is interested in more focused content and accuracy boosting strategies have more impact on overall performance. The results of our empirical evaluations confirmed this hypothesis, as the overall performance, i.e., both accuracy and diversity, was improved over the original methods.

For future work, we propose to investigate diversity enhancing approach in model-based SBRS such as GRU4REC [30], CHAMELEON [62] and Session-based Matrix Factorization (SMF) [20]. In this study we only assessed music datasets. Further study is needed on datasets from other domains with different types of meta-data such as tags, text and named entities in news recommendations [63] or genres, actors and actresses in movie recommendations to evaluate the impact of the introduced methods on diversity and accuracy. Another interesting future work is to investigate the effect of using more graded feedback such as listening time and interaction timestamp [64] instead of binary implicit feedback in session based music recommendations.

Funding

This work was executed within the imec.icon project NewsButler. The NewsButler project is co-financed by imec and received project support from Flanders Innovation & Entrepreneurship (project nr. HBC.2017.0628).

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Alireza Gharahighehi and Celine Vens. Making session-based news recommenders diversity-aware. In *OHARS'20: Workshop on Online Misinformation- and Harm-Aware Recommender Systems*, page (to appear), 2020.
2. Tien T Nguyen, Pik-Mai Hui, F Maxwell Harper, Loren Terveen, and Joseph A Konstan. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *Proceedings of the 23rd international conference on World wide web*, pages 677–686, 2014.

3. Marius Kaminskis and Derek Bridge. Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 7(1):2, 2017.
4. Pablo Castells, Neil J Hurley, and Saul Vargas. Novelty and diversity in recommender systems. In *Recommender Systems Handbook*, pages 881–918. Springer, 2015.
5. Barry Smyth and Paul McClave. Similarity vs. diversity. In *International conference on case-based reasoning*, pages 347–361. Springer, 2001.
6. Lei Li, Dingding Wang, Tao Li, Daniel Knox, and Balaji Padmanabhan. Scene: a scalable two-stage personalized news recommendation system. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 125–134. ACM, 2011.
7. Maunendra Sankar Desarkar and Neha Shinde. Diversification in news recommendation for privacy concerned users. In *2014 International Conference on Data Science and Advanced Analytics (DSAA)*, pages 135–141. IEEE, 2014.
8. Saúl Vargas, Linas Baltrunas, Alexandros Karatzoglou, and Pablo Castells. Coverage, redundancy and size-awareness in genre diversity for recommender systems. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 209–216. ACM, 2014.
9. Tommaso Di Noia, Jessica Rosati, Paolo Tomeo, and Eugenio Di Sciascio. Adaptive multi-attribute diversity for recommender systems. *Information Sciences*, 382:234–253, 2017.
10. Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, pages 22–32. ACM, 2005.
11. Fuguo Zhang. Research on recommendation list diversity of recommender systems. In *2008 International Conference on Management of e-Commerce and e-Government*, pages 72–76. IEEE, 2008.
12. John Paul Kelly and Derek Bridge. Enhancing the diversity of conversational collaborative recommendations: a comparison. *Artificial Intelligence Review*, 25(1-2):79–95, 2006.
13. Saúl Vargas and Pablo Castells. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 109–116. ACM, 2011.
14. Marco Tulio Ribeiro, Anisio Lacerda, Adriano Veloso, and Nivio Ziviani. Pareto-efficient hybridization for multi-objective recommender systems. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 19–26. ACM, 2012.
15. Cong Yu, Laks VS Lakshmanan, and Sihem Amer-Yahia. Recommendation diversification using explanations. In *2009 IEEE 25th International Conference on Data Engineering*, pages 1299–1302. IEEE, 2009.
16. Ruilong Su, Li’Ang Yin, Kailong Chen, and Yong Yu. Set-oriented personalized ranking for diversified top-n recommendation. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 415–418. ACM, 2013.
17. Saul Vargas, Pablo Castells, and David Vallet. Intent-oriented diversity in recommender systems. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1211–1212. ACM, 2011.
18. Martijn C Willemsen, Bart P Knijnenburg, Mark P Graus, Linda CM Velter-Bremmers, and Kai Fu. Using latent features diversification to reduce choice difficulty in recommendation lists. *RecSys*, 11(2011):14–20, 2011.
19. Yue Shi, Xiaoxue Zhao, Jun Wang, Martha Larson, and Alan Hanjalic. Adaptive diversification of recommendation results via latent factor portfolio. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 175–184. ACM, 2012.
20. Malte Ludewig and Dietmar Jannach. Evaluation of session-based recommendation algorithms. *User Modeling and User-Adapted Interaction*, 28(4-5):331–390, 2018.
21. Shoujin Wang, Longbing Cao, and Yan Wang. A survey on session-based recommender systems. *arXiv preprint arXiv:1902.04864*, 2019.
22. Iman Kamehkhosh, Dietmar Jannach, and Malte Ludewig. A comparison of frequent pattern techniques and a deep learning method for session-based recommendation. In *RecTemp@ RecSys*, pages 50–56, 2017.
23. Dietmar Jannach and Malte Ludewig. When recurrent neural networks meet the neighborhood for session-based recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 306–310. ACM, 2017.
24. Michael Jugovac, Dietmar Jannach, and Mozhgan Karimi. Streamingrec: a framework for benchmarking stream-based news recommenders. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 269–273. ACM, 2018.

25. Malte Ludewig, Noemi Mauro, Sara Latifi, and Dietmar Jannach. Empirical analysis of session-based recommendation algorithms. *User Modeling and User-Adapted Interaction*, pages 1–33, 2020.
26. Malte Ludewig, Noemi Mauro, Sara Latifi, and Dietmar Jannach. Performance comparison of neural and non-neural approaches to session-based recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 462–466, 2019.
27. Pigi Kouki, Ilias Fountalis, Nikolaos Vasiloglou, Xiquan Cui, Edo Liberty, and Khalifeh Al Jadda. From the lab to production: A case study of session-based recommendations in the home-improvement domain. In *Fourteenth ACM Conference on Recommender Systems*, pages 140–149, 2020.
28. Panagiotis Symeonidis, Andrea Janes, Dmitry Chaltsev, Philip Giuliani, Daniel Morandini, Andreas Unterhuber, Ludovik Coba, and Markus Zanker. Recommending the video to watch next: An offline and online evaluation at youtv. de. In *Fourteenth ACM Conference on Recommender Systems*, pages 299–308, 2020.
29. Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, (1):76–80, 2003.
30. Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.
31. Balázs Hidasi and Alexandros Karatzoglou. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 843–852. ACM, 2018.
32. Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1419–1428, 2017.
33. Shoujin Wang, Liang Hu, Longbing Cao, Xiaoshui Huang, Defu Lian, and Wei Liu. Attention-based transactional context embedding for next-item recommendation. In *AAAI*, pages 2532–2539, 2018.
34. Heng-Shiou Sheu and Sheng Li. Context-aware graph embedding for session-based news recommendation. In *Fourteenth ACM Conference on Recommender Systems*, pages 657–662, 2020.
35. Siyi Liu and Yujia Zheng. Long-tail session-based recommendation. In *Fourteenth ACM Conference on Recommender Systems*, pages 509–514, 2020.
36. Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 346–353, 2019.
37. Jaime G Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, volume 98, pages 335–336, 1998.
38. Saúl Vargas and Pablo Castells. Improving sales diversity by recommending users to items. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 145–152. ACM, 2014.
39. Andrea Barraza-Urbina, Benjamin Heitmann, Conor Hayes, and Angela Carrillo-Ramos. Xplodiv: An exploitation-exploration aware diversification approach for recommender systems. In *The Twenty-Eighth International Flairs Conference*, 2015.
40. Tamas Jambor and Jun Wang. Optimizing multiple objectives in collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 55–62. ACM, 2010.
41. Mi Zhang and Neil Hurley. Avoiding monotony: improving the diversity of recommendation lists. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 123–130. ACM, 2008.
42. Alan Said, Ben Fields, Brijnesh J Jain, and Sahin Albayrak. User-centric evaluation of a k-furthest neighbor collaborative filtering recommender algorithm. In *Proceedings of the 2013 conference on Computer supported cooperative work*, pages 1399–1408. ACM, 2013.
43. Harry Markowitz. Portfolio selection. *The journal of finance*, 7(1):77–91, 1952.
44. Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 452–461. AUAI Press, 2009.
45. Neil J Hurley. Personalised ranking with diversity. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 379–382. ACM, 2013.

46. Gábor Takács and Domonkos Tikk. Alternating least squares for personalized ranking. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 83–90. ACM, 2012.
47. Michael Jahrer and Andreas Töschner. Collaborative filtering ensemble for ranking. In *Proceedings of the 2011 International Conference on KDD Cup 2011-Volume 18*, pages 153–167. JMLR. org, 2011.
48. Jacek Wasilewski and Neil Hurley. Incorporating diversity in a learning to rank recommender system. In *The Twenty-Ninth International Flairs Conference*, 2016.
49. Shoujin Wang, Liang Hu, Yan Wang, Quan Z Sheng, Mehmet A Orgun, and Longbing Cao. Modeling multi-purpose sessions for next-item recommendations via mixture-channel purpose routing networks. In *IJCAI*, pages 3771–3777, 2019.
50. Yong Kiam Tan, Xinxing Xu, and Yong Liu. Improved recurrent neural networks for session-based recommendations. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 17–22. ACM, 2016.
51. Gabriel de Souza P Moreira, Dietmar Jannach, and Adilson Marques da Cunha. On the importance of news content representation in hybrid neural session-based recommender systems. *arXiv preprint arXiv:1907.07629*, 2019.
52. Shoujin Wang, Liang Hu, and Longbing Cao. Perceiving the next choice with comprehensive transaction embeddings for online recommendation. In Michelangelo Ceci, Jaakko Hollmén, Ljupčo Todorovski, Celine Vens, and Sašo Džeroski, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 285–302, Cham, 2017. Springer International Publishing.
53. Rodrygo LT Santos, Craig Macdonald, and Iadh Ounis. Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th international conference on World wide web*, pages 881–890. ACM, 2010.
54. Saúl Vargas and Pablo Castells. Exploiting the diversity of user preferences for recommendation. In *Proceedings of the 10th conference on open research areas in information retrieval*, pages 129–136. Le Centre De Hautes Etudes Internationales D’informatique Documentaire, 2013.
55. Vito W Anelli, Vito Bellini, Tommaso Di Noia, Wanda La Bruna, Paolo Tomeo, and Eugenio Di Sciascio. An analysis on time-and session-aware diversification in recommender systems. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 270–274. ACM, 2017.
56. Ramazan Esmeli, Mohamed Bader-El-Den, and Hassana Abdullahi. Improving session based recommendation by diversity awareness. In *UK Workshop on Computational Intelligence*, pages 319–330. Springer, 2019.
57. Roberto Turrin, Massimo Quadrona, Andrea Condorelli, Roberto Pagano, and Paolo Cremonesi. 30music listening and playlists dataset. In *RecSys Posters*, 2015.
58. O. Celma. *Music Recommendation and Discovery in the Long Tail*. Springer, 2010.
59. Eva Zangerle, Martin Pichl, Wolfgang Gassler, and Günther Specht. # nowplaying music dataset: Extracting listening behavior from twitter. In *Proceedings of the First International Workshop on Internet-Scale Multimedia Management*, pages 21–26. ACM, 2014.
60. Brian McFee and Gert RG Lanckriet. The natural language of playlists. In *ISMIR*, volume 11, pages 537–541, 2011.
61. Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 811–820, 2010.
62. Gabriel de Souza Pereira Moreira, Dietmar Jannach, and Adilson Marques da Cunha. Contextual hybrid session-based news recommendation with recurrent neural networks. *arXiv preprint arXiv:1904.10367*, 2019.
63. Alireza Gharahighehi, Celine Vens, and Konstantinos Pliakos. Multi-stakeholder news recommendation using hypergraph learning. In *Proceedings of the 8th International Workshop on News Recommendation and Analytics*, page (to appear). Springer, 2020.
64. Alireza Gharahighehi and Celine Vens. Extended bayesian personalized ranking based on consumption behavior. In *Postproceedings of the 31st Benelux Conference on Artificial Intelligence (BNAIC 2019) and the 28th Belgian Dutch Conference on Machine Learning (Benelearn 2019)*, page (to appear). Springer, 2020.