Adaptation and Evaluation of Recommendations for Short-term Shopping Goals

Dietmar Jannach
TU Dortmund, Germany
dietmar.jannach@tu-dortmund.de

Lukas Lerche
TU Dortmund, Germany
lukas.lerchet@tu-dortmund.de

Michael Jugovac
TU Dortmund, Germany
michael.jugovac@tu-dortmund.de

ABSTRACT

An essential characteristic in many e-commerce settings is that website visitors can have very specific short-term shopping goals when they browse the site. Relying solely on long-term user models that are pre-trained on historical data can therefore be insufficient for a suitable next-basket recommendation. Simple "real-time" recommendation approaches based, e.g., on unpersonalized co-occurrence patterns, on the other hand do not fully exploit the available information about the user's long-term preference profile.

In this work, we aim to explore and quantify the effectiveness of using and combining long-term models and short-term adaptation strategies. We conducted an empirical evaluation based on a novel evaluation design and two real-world datasets. The results indicate that maintaining short-term *content-based* and *recency-based* profiles of the visitors can lead to significant accuracy increases. At the same time, the experiments show that the choice of the algorithm for learning the long-term preferences is particularly important at the beginning of new shopping sessions.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering

Keywords

E-Commerce; Algorithms; Context; Evaluation

1. INTRODUCTION

Modern online shops are no longer static catalogs of items, but meet the user's interest by providing personalized product recommendations. In the best case, these recommendations should match both the users' long-term preferences as well as their current shopping goals. On Amazon.com, for example, each product page contains multiple personalized recommendation lists with different purposes. They comprise complements or alternatives to the currently viewed product, remind the user of recently viewed items, or represent recommendations that should appeal to the general taste of the user (Figure 1). Therefore, the displayed content not only depends on the features of the currently viewed article, like the product category, but can also be influenced, e.g., by a combination of the user's past shopping behavior (long-term preferences) and his most recent navigation actions (short-term shopping goals).

Research in recommender systems (RS) has made impressive advances over the last decade in particular with respect to algorithms capable of modeling long-term user preferences. Note, however, that the first four recommendation lists of the real-world site shown in Figure 1 actually appear to be non-personalized and merely depend on the currently viewed item. The last list, in contrast, is based on recent navigation actions and furthermore seems to integrate recommendations that are based on a longer-term user profile. Overall, relying solely on long-term models seems to be insufficient in this situation, as these models cannot easily adapt to the user's short-term shopping goals.

The goal of our work is to explore, quantify, and compare the effectiveness of using such short-term and long-term user profiles in online shopping scenarios. We will therefore first design and evaluate three different "real-time" recommendation strategies that can be found on real shops and that are based on item co-occurrence patterns, content-based similarity and recent item views. We will then compare these approaches with state-of-the-art long-term recommendation models and finally test hybrids that combine short-term and long-term models.

We will base our evaluations on real-world navigation log data from two real-world shopping sites. Since the standard evaluation approaches from the literature [13] do not cover our specific situation, we first propose a general and domain-independent time-based and session-based evaluation protocol which can in particular help us to assess how quickly the different recommendation strategies can adapt their recommendations to the visitor's short-term goals.

Generally, our work is related to recent works in context-aware recommendation approaches, which try to consider information about the user's current situation, environment, as well as temporal dynamics in the recommendation process [1, 7]. However, only limited works exist that use navigation logs to estimate the user's shopping goals as the recommendation context and to our knowledge no work exists to date that aims to compare and quantify the individual and combined effects of short-term and long-term profiles using real-world navigation log data.

RecSys'15, September 16–20, 2015, Vienna, Austria.
© 2015 ACM. ISBN 978-1-4503-3692-5/15/09 ...$15.00.
DOI: http://dx.doi.org/10.1145/2792838.2800176.

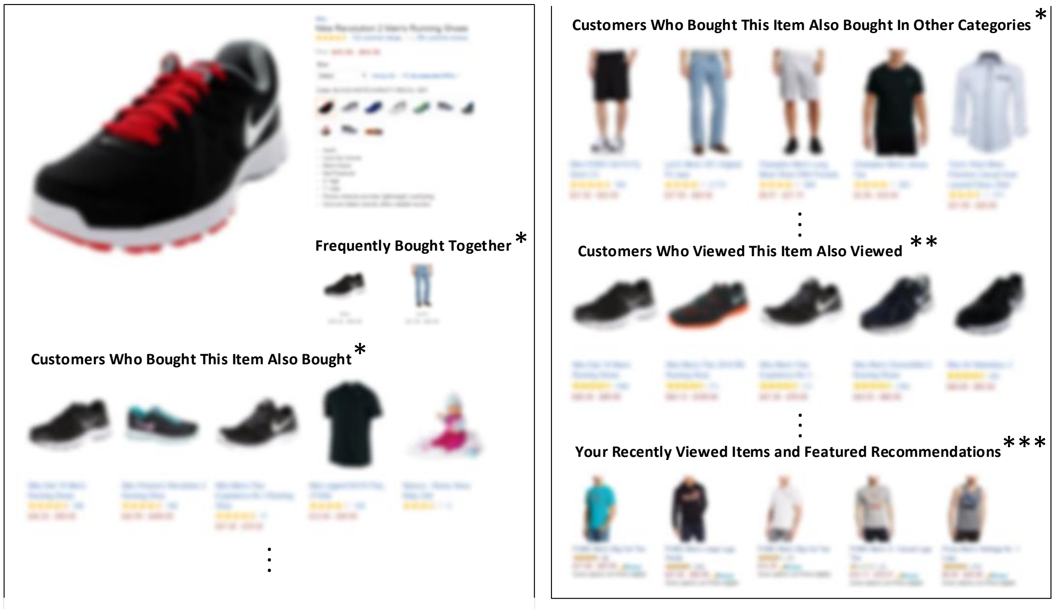
Frequently Bought Together *
Customers Who Bought This Item Also Bought *
Customers Who Bought This Item Also Bought In Other Categories *
Customers Who Viewed This Item Also Viewed **
Your Recently Viewed Items and Featured Recommendations ***

Figure 1: Simplified structure of an Amazon.com product detail page as of 2014, which contains personalized recommendations for *complements, **alternatives and ***reminders.

In the next section, we present the details of the proposed evaluation protocol, which is inspired by the protocol used by our industrial partner. The evaluated recommendation schemes and the results of an empirical evaluation on two real-world datasets are described in later sections.

2. EVALUATION PROTOCOL

The proposed protocol is designed to help us evaluate both the prediction accuracy and the capability of algorithms to adapt to short-term shopping goals in a realistic way. The general goal is that it can be used with various types of time-ordered implicit feedback signals which can be acquired, e.g., from web shop navigation logs. Such information is typically available on e-commerce platforms and furthermore corresponds to what is sometimes shared by companies for research purposes as in the RecSys 2015 challenge[1]. Regarding the long-term models, state-of-the-art recommendation algorithms – in particular those relying on implicit feedback – can be applied and evaluated with standard accuracy measures like precision, recall, or the mean reciprocal rank (MRR).

Figure 2 visualizes the main idea and the different steps of the protocol. We first split the available data (i.e., the sequence of log actions) into a training and a test set. Since the time and the sequence of the actions is important when considering short-term and long-term interests, we use a time-related splitting criterion. We can, for example, put all but the last two shopping *sessions*[2] of each user in the training set, which serves as a basis to learn a long-term user model.

Then, instead of using an RS algorithm to recommend one single ranked list of items given the training set, the task is rather to predict user actions, e.g., purchases, *for each session* in the test set. The underlying idea is that the user's shopping goals can be different for each session. In order to assess how quickly different strategies adapt their recommendations to the session-specific goals, the protocol furthermore provides the means to "reveal" a defined number of very *recent* user actions.

These most recent actions – which we will call *context* from here on – are, for example, "view" actions during the currently evaluated session or actions from a limited number of preceding sessions. Since no k-fold cross-validation is possible due to the time-based split, we propose to apply repeated random sub-sampling to avoid random effects.

Generally, as an alternative to hiding only the last n sessions, the proposed protocol can be varied to implement a "sliding window" technique, e.g., in order to vary the amount of available training data for the long-term models. Such analyses are however not in the focus of our current work.

The steps of the protocol can be summarized as follows.

Training phase.

1. Create a time-ordered list of user sessions and their actions from the log data.

2. Split the session list into training and test sessions while retaining the sequence. Splitting criteria could be, e.g., "retain the last session containing a purchase", or "retain the last 20% of the sessions". Remove users with session lists that are too short to be split (e.g., the user only visited the shop once).

3. Use an arbitrary recommendation algorithm to learn a long-term user model from the training data.

[1] http://2015.recsyschallenge.com/

[2] A session is a set of user actions identified by the *session ID* of each action assigned by the system of the dataset provider. In general, a session represents actions of a user within a particular time period or for completing a particular task.

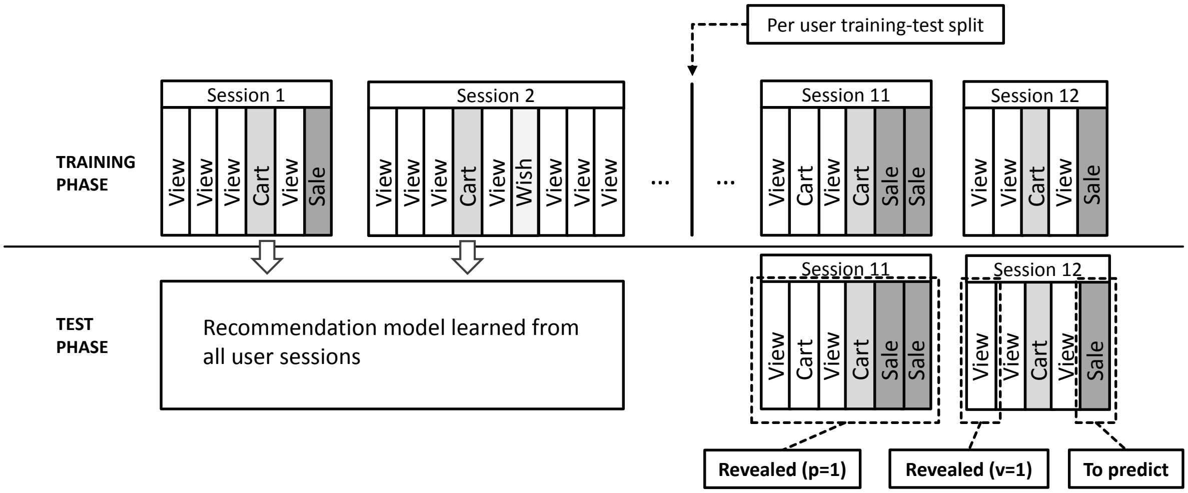
Per user training-test split
TRAINING PHASE
Session 1
View View View Cart View Sale
Session 2
View View View Cart View Wish View View View
...
...
Session 11
View Cart View Cart Sale Sale
Session 12
View View Cart View Sale
TEST PHASE
Recommendation model learned from all user sessions
Session 11
View Cart View Cart Sale Sale
Session 12
View View Cart View Sale
Revealed (p=1)
Revealed (v=1)
To predict

Figure 2: Protocol sketch: training phase and evaluation of the 12th session of a user.

Recommendation and evaluation.

1. Predict the next user action, e.g., purchase, for each session in the test set that contains at least one action of the desired type.

2. To allow a given algorithm to adapt its strategy to short-term goals, reveal a set of recent actions (the user's context) in addition to the training data. Here, two different types of information are used.

   - Parameter v determines up to how many *views* of the *currently evaluated* session – e.g., session 12 in Figure 2 – are revealed. This parameter helps us to assess how many clicks it takes for an adaptive recommendation strategy to guess the user's shopping goal.
   - Parameter p determines up to how many *previous* sessions – with respect to the currently evaluated one – are revealed. By changing the parameter a potential recent interest drift of the visitor during the last few sessions can be determined.

3. Use hit rates or rank-based measures to assess the quality of the predictions.

Discussion.

The general question in offline experimental designs is whether the chosen evaluation procedure and the performance metrics represent a good approximation of the *true quality* of an information system. Recent studies for example indicate that algorithms that achieve lower RMSE values are not necessarily favorable when we look at the specific business or application goals to be achieved (see, e.g., [9, 15, 18]). Overall, we are confident that the proposed form of predicting user actions is a realistic way to asses the effectiveness and accuracy of different algorithms. First, our protocol is similar to one protocol used by our research partner Zalando for offline performance evaluations[3]. In addition, recent RS competitions held by industrial sponsors, e.g., the *Tmall* Prize competition[4] and the 2015 ACM RecSys Challenge, use protocols that share several similarities with our protocol. The main difference to these existing protocols is that our approach is more generic in terms of compatible algorithms and that the simulation of short-term information is more parameterizable.

3. EXPERIMENTAL SETUP

In the following sections, we propose a number of adaptation strategies and report the results of a series of experiments which we conducted using two datasets: a larger one provided by the online fashion retailer Zalando, and a smaller one from the Tmall competition.

3.1 Details of the Zalando dataset

The "raw" dataset from Zalando comprises nearly 1 million purchases, 1.6 million cart actions and about 20 million item view events in about 170,000 sessions. The data was sampled from the shop's web log in a way that no conclusions about the visitors or the business numbers can be drawn. There are 800,000 anonymized user IDs in the dataset; however, more than 500,000 of the visitors have never made a purchase but only viewed items. The product catalog is huge and comprises over 150,000 different items.

Data sparsity represents a major problem in our application scenario. The average number of purchases per user, for example, is about 3.5 when only counting those users which have ever made a purchase. Each item ever purchased was sold on average about 6 times. Finally, a major fraction of the users visited the shop only once.

Personalization based on past behavior is therefore only reasonable for the subset of "heavy" users. Following a common practice in research, we not only used the full dataset but created subsets with higher data density by applying constraints on the minimum number of purchases per user and per item. In Table 1 we show the resulting characteristics of the data subsamples called *Sparse*, *Medium*, and

[3] Zalando is a major European online retailer of fashion products, see `http://www.zalando.com`

[4] `http://102.alibaba.com/competition/addDiscovery/index.html`

Table 1: Characteristics of Zalando evaluation data sets with different density constraints. The datasets are still sparse; note that users in typical movie datasets have rated at least 20 items.

	Sparse	Medium	Dense
Users	121,018	38,447	1,869
Items	40,526	19,061	2,208
Purchases	680,787	344,684	43,079
Views	9,807,282	3,929,813	117,734
Min. purchases/user	3	5	10
Min. purchases/item	3	5	10

Dense. In our evaluation, we only retained view and purchase actions but not the cart actions; we will discuss this decision later on.

3.2 Algorithms

We differentiate between (i) non-contextualized *baseline strategies* used to learn the long-term user models and (ii) *contextualization strategies* that rely on the additionally revealed user actions to adapt the recommendations to the short-term shopping goals.

Non-contextualized baseline strategies

Any existing recommendation algorithm capable of building a model from the given implicit rating data can be used as a baseline. The task of the baseline strategy is to compute a ranked list of items for a given user based on the training data. We chose the following algorithms in the experiments to generate long-term models.

BPR: Matrix factorization (MF) and learning-to-rank techniques represent the most successful classes of methods to build highly accurate recommender systems in the recent literature, especially for implicit feedback domains. Therefore, and since only implicit user feedback is available, we use BPR (Bayesian Personalized Ranking) [25] in combination with an MF model of 100 features as a state-of-the-art baseline in our experiments. The model was learned in 100 training steps with learning rate $\alpha = 0.05$ and regularization parameters $\lambda_W = 0.0025$, $\lambda_{H^+} = 0.0025$, $\lambda_{H^-} = 0.00025$. The optimal parameters for BPR and the next algorithm (FctMch) were manually fine-tuned.

FctMch: Factorization Machines [24] combine feature engineering and factorization models and can be applied for general prediction tasks. Markov Chain Monte Carlo optimization (MCMC) was used in the experiments with the parameters stddev = 0.3, steps = 50.

PopRank: Depending on the evaluation setting, simple popularity-based approaches can represent a quite hard baseline [10]. We implemented an unpersonalized baseline strategy which ranks the items based on the number of times they have been viewed or purchased in the training set.

Random: A method that recommends random items to users. We include this baseline to assess the effects when only short-term techniques are applied (and the resulting lists are filled up with random elements).

We do not recommend items to users that they already purchased in the past. We also do not differentiate between item views and purchases in the training phase, since contextualization is our main focus. In more elaborate schemes, a graded relevance feedback technique could be used as in [20] or [30] and more weight could, for example, be given to purchase actions.

Contextualization strategies

The following approaches are inspired by those of Amazon.com and similar shops and rely on the additional information about the user's recent navigation behavior (*context*), which the evaluation protocol reveals to them. In one of the proposed schemes, we also use additional content information. The contextualization strategies are of low computational complexity and therefore suitable for real-time adaptation of the recommendations. The general strategy of all techniques is to refine the results returned by a *baseline recommender* – which reflects the long-term model – with the help of the current context. This can be considered as a form of contextual post-filtering.

CoOccur: In this method, the recommendable items are ranked by the *conditional probability* of co-occurring with the items in the user's context (which corresponds to association rules of size two and the "Customers who viewed ..." strategy shown in Figure 1). Items for which no scores can be computed are appended to the list in the order in which they were ranked by the baseline method.

CoOccur-Filter: The method again combines co-occurrence scores with the item ranking of the baseline method. The recommendation list again starts with items for which co-occurrence scores can be computed. However, the ordering of the baseline method is used for ranking the items. Again, the list is then filled up with the remaining items using the order of the baseline method.

FeatureMatching (FM): This method re-ranks the items returned by a baseline recommender based on *content features*. Specifically, a short-term content-based user profile is created that contains the brand and category information of items that the user has recently viewed. Each recommendable item is compared with this short-term profile and re-ranked with a weight factor based on the number of overlapping feature values.

RecentlyViewed (RV): This technique places the recently viewed items on the top of the recommendation list and appends the remaining items. The internal ranking of both parts of the list is based on the baseline method's score. This strategy was designed in analogy to the last set of recommendations shown in Figure 1 used on Amazon.com.

3.3 Evaluation measures

Since the data only contains unary (positive) feedback, we use the protocol variant of [10] to determine the *recall* by measuring the relative position of the recommended items in a given set of items. Each target item t is combined with k random items unknown to the user. The resulting list of $k + 1$ items is then ranked by the algorithm. A "hit" occurs when t was in the top n items. The recall for each ranking problem is therefore either 0 or 1. Precision can be computed as $1/(k + 1) \cdot \mathit{recall}$ and is thus proportional to the recall. We report the results obtained using $k = 100$ and Recall@10.

We additionally measured the MRR , and the results were in line with the results for Recall@10. Varying the protocol parameters k and n led to different absolute results but did not change the overall ranking of the algorithms.

Table 2: Recall for the *Dense* Zalando data set

	v=0, p=2	v=2, p=2	v=5, p=2	v=10, p=2	v=5, p=0
BPR	0.40				
FctMch	0.20				
PopRank	0.21				
Random	0.09				
CoOccur +					
BPR	0.38	0.47	0.49	0.52	0.48
FctMch	0.38	0.46	0.48	0.52	0.42
PopRank	0.37	0.45	0.48	0.51	0.41
Random	0.31	0.44	0.48	0.50	0.40
CoOccur-Filter +					
BPR	0.39	0.47	0.48	0.50	0.48
FctMch	0.31	0.37	0.39	0.41	0.40
PopRank	0.29	0.36	0.38	0.39	0.38
Random	0.23	0.34	0.35	0.37	0.36
FeatureMatching +					
BPR	0.41	0.65	0.71	0.76	0.71
FctMch	0.37	0.61	0.68	0.73	0.63
PopRank	0.38	0.62	0.68	0.72	0.64
Random	0.31	0.60	0.66	0.73	0.61
RecentlyViewed +					
BPR	0.40	0.55	0.64	0.72	0.63
FctMch	0.36	0.53	0.61	0.70	0.51
PopRank	0.36	0.53	0.62	0.71	0.52
Random	0.27	0.47	0.57	0.67	0.47
RecentlyViewed + FeatureMatching +					
BPR	0.41	**0.66**	**0.73**	**0.79**	**0.71**
FctMch	**0.42**	0.65	0.72	**0.79**	0.63
PopRank	0.40	0.65	0.72	**0.79**	0.64
Random	0.32	0.63	0.70	0.78	0.62

4. RESULTS

We report the results of different configurations, where the maximum number of revealed views v and previous sessions p was varied to determine the importance of the recent actions and sessions. The setting $v = 0$, $p = 0$ corresponds to the non-contextualized evaluation shown in the second row of Table 2. We randomly selected 80% of the users and repeated the measurements 5 times using different samples. For each user, the recommendation task was to predict the purchases of the last session in which a purchase was observed. Table 2 shows the results on the *Dense* dataset for the baseline strategies and the effects when adding the contextualization strategies. The standard deviations across the different measurements were between 0.003 and 0.015 for the recall, i.e., the results are quite stable.

Observations for baseline strategies.

Regarding the baseline strategies, the BPR learning-to-rank technique for implicit feedback unsurprisingly – see also [16] – leads to the best results for recall (and all other measures) as shown in the first data rows of Table 2. Beating the popularity-based baseline is generally hard in this specific setup as discussed in [10], in particular when only implicit feedback is available, for which techniques like FctMch are not optimized[5]. While using FctMch alone is not better than PopRank in this setup, we will see in the following that FctMch can be a better suited baseline strategy when contextualization is applied.

General observations regarding contextualization.

All contextualization strategies lead to better recall (and MRR) values than when using the baselines alone in case the navigation actions of the current session ($v \geqslant 2$) are taken into account. The differences are statistically significant ($p < 0.01$). The accuracy of all techniques also consistently increases when more views of the current session are revealed, i.e., all techniques are able to adapt their recommendations to the current short-term goals based on the navigation behavior. Revealing more previous sessions (e.g., $p = 0$ vs. $p = 2$ with $v = 5$) has a much lower impact on the performance, especially for BPR[6]. Combining these two findings indicates not only that the most recent actions reflect the user's short-term shopping goals, but also that these goals can vary quickly from session to session.

Using short-term content-based profiles.

The content-enhanced FeatureMatching method works very well in terms of recall with any of the baseline techniques. This indicates that many users arrive at the web site with a specific shopping goal and focus their navigation on items of a certain category in which they finally make a purchase. In addition, brand loyalty seems to be a relevant domain-specific aspect.

Users who viewed ... also viewed ....

The co-occurrence based and non-personalized "standard" technique of modern shops leads to comparably high recall values, even if the lists are combined (continued) with random elements and only little is known about the current session (e.g., $v = 2$). Combinations with other baseline methods lead to further statistically significant improvements ($p < 0.01$) for all cases except for BPR with $v = 0$.

This observation indicates that co-occurrence based short-term models can significantly contribute to the overall accuracy of a system. Combining these models with stronger baselines helps to further increase the accuracy, in particular when nothing is known about what the user was interested in during previous sessions ($v = 5$, $p = 0$).

The CoOccur-Filter leads to similar results when BPR is used as a baseline but exhibits lower absolute recall values when the baseline is weaker.

Recently viewed items.

Recommending what users have recently viewed leads to very good results with respect to recall, independent of the chosen baseline strategy. Note that in our application scenario a purchase action for an item is in most cases preceded by a view action in the current or one of the previous sessions. This explains the comparably high values for recall because it is not unlikely that the user viewed the purchased item at the beginning of a session or has slept on his decision since the last session.

Reminding users of items they have recently looked at generally seems to be a reasonable strategy. However, our rank measures unfortunately do not tell us if focusing on such items supports the business strategy of a company in the best possible way. Recommending only items that the visitor already knows will, e.g., not help to stimulate them to purchase items from other parts of the product catalog. At the same time, visitors might find such recommendations

[5] An evaluation using the traditional item-to-item nearest-neighbor scheme led to even worse results, which we omit here for space reasons.

[6] Results when using $p = 1$ (not shown) are almost identical to $p = 2$.

Table 3: Recall for the *Medium*, *Sparse* and *"Raw"* (complete) Zalando datasets

	v=0, p=2	v=2, p=2	v=5, p=2	v=10, p=2	v=5, p=0
Medium: RecentlyViewed + FeatureMatching +					
BPR	**0.53**	**0.69**	**0.73**	**0.78**	**0.72**
FctMch	0.52	0.68	0.72	0.77	0.73
PopRank	0.51	0.68	0.72	0.77	0.65
Random	0.36	0.59	0.65	0.72	0.55
Sparse: RecentlyViewed + FeatureMatching +					
BPR	**0.59**	**0.75**	**0.79**	**0.83**	**0.77**
FctMch	0.58	0.73	0.77	0.82	0.73
PopRank	0.58	0.73	0.78	0.82	0.72
Random	0.36	0.63	0.69	0.76	0.60
Raw dataset: RecentlyViewed + FeatureMatching +					
BPR	**0.64**	**0.74**	**0.77**	**0.81**	**0.74**
FctMch	0.63	**0.74**	**0.77**	**0.81**	0.72
PopRank	0.63	**0.74**	**0.77**	**0.81**	0.71
Random	0.36	0.59	0.64	0.70	0.54

Table 4: Recall for the *Tmall* data set

	v=0, p=2	v=2, p=2	v=5, p=2	v=10, p=2	v=5, p=0
BPR	**0.49**				
FctMch	0.21				
PopRank	0.24				
Random	0.10				
CoOccur +					
BPR	**0.45**	0.43	0.45	0.46	0.42
FctMch	0.43	0.44	0.44	0.46	0.27
PopRank	0.41	0.40	0.43	0.43	0.25
Random	0.39	0.39	0.43	0.42	0.21
CoOccur-Filter +					
BPR	**0.45**	0.45	0.45	0.46	0.42
FctMch	0.37	0.35	0.37	0.37	0.27
PopRank	0.36	0.32	0.34	0.33	0.23
Random	0.31	0.27	0.30	0.30	0.20
RecentlyViewed +					
BPR	0.40	**0.78**	**0.88**	**0.92**	**0.88**
FctMch	0.38	0.77	0.87	0.92	0.80
PopRank	0.36	0.75	0.85	0.91	0.80
Random	0.33	0.72	0.83	0.91	0.75

to be of limited value. Still, Amazon.com, for example, also recommends items under the "recently viewed" label that were viewed in the current session.

Finally, combining the recency-based approach with the content-based technique and the strongest baseline BPR leads to the best overall results as shown in the last rows of Table 2. Compared to RecentlyViewed and FeatureMatching individually, the improvements obtained with the hybrid of both strategies can be significant if enough context information is available (e.g., $p = 2$ and $v \geqslant 2$). Otherwise, the combination is about as good as FeatureMatching alone.

Results for lower-density datasets.

Table 3 shows results for the best performing strategy on the larger, low-density datasets. On the *raw* dataset we only made predictions for users that had at least 5 purchases in the test set. Overall, the same trends can be observed as with the dataset of higher density. Again, the combination of the different contextualization techniques and using BPR as a baseline leads to the best results. Although comparing the absolute recall values across the different datasets should be done with care, we can, to some surprise, see that the absolute values are similar or even better for the lower-density datasets. The reasons for this can be (a) that the contextualization strategies are responsible for a major fraction of the hits and (b) that some of the baseline methods profit from a much larger training data set.

Validation on an additional dataset.

We repeated the experiments on another real-world dataset of the recommender competition held by the Chinese online retailer *Tmall*[7]. The published dataset is smaller but has similar characteristics as the Zalando dataset and comprises about 175,000 time-stamped view actions and 7,000 purchase actions organized in sessions. There were about 750 users, who on average purchased around 8 items of the about 2,000 different products. It contains, however, no content information about the items.

Therefore, we could only use the RecentlyViewed and CoOccur strategies. The results corroborate most of our previous observations, see Table 4. BPR generally leads to higher and statistically significant recall and MRR values than the popularity-based and FctMch approaches. The non-contextualized recall for BPR was about 0.49 and 0.24 for PopRank. Focusing on recently viewed items expectedly leads to strong improvements. The recall with BPR as a baseline increased to 0.78 and to 0.75 when using PopRank ($p = 2$, $v = 2$). The CoOccur contextualization strategy, however, only leads to significant improvements when the popularity-based scheme is used. CoOccur with the BPR baseline performs worse than using BPR without contextualization. Furthermore, the item-to-item method again did not reach the performance level of the popularity-based method in this setup. These effects are most probably caused by the small size of the dataset.

5. DISCUSSION

Implications. Our results support the assumption that taking the short-term interests of visitors into account and combining them with long-term preference models can help to significantly increase the recommendation accuracy. Specifically, even comparably simple, "real-time"-enabled techniques based on content-similarity, item co-occurrence or recent user behavior can be helpful, even in cases in which only limited information about the current shopping session is revealed. At the same time, the choice of a good baseline technique can be essential and our analysis has revealed that recent methods like BPR lead to good results in the explored domains. Optimizing for accuracy measures through offline experiments as done in the research literature is therefore important. In practice, however, these recommendations should be paired with complementary techniques.

Another observation was that the consideration of domain-specific aspects can be crucial. In particular brand loyalty seems to be a common phenomenon as well as the tendency of users to make repeated purchases from a restricted set of product categories. This tendency to purchase "more of the same" was also observed in the real-world study in [15].

Open questions. From a methodological perspective, we see our work as a step towards more realistic and complementary research designs for recommender systems. However, while our protocol allows us to assess the role of short-

[7] `http://www.tmall.com`

term interests to some extent using an offline experimental design, some questions remain to be explored in future works. One specific aspect to consider is, how certain recent user actions should be considered in the recommendation process. Should items be recommended that the user has already viewed before? Should we remind the users of an item that they have placed in the cart some time ago and never purchased? In which specific ways should we incorporate other types of user actions? These decisions often depend on the application domain and cannot easily be generalized or answered through offline experiments.

A more general issue is that relying on established measures like recall and MRR alone to predict the effectiveness of an RS can be misleading as these measures do not take alternate objectives and business goals into account. Recommending only recently viewed items can, for example, result in comparably high hit rates but perhaps not in additional purchases. Such an approach is also not suited for directing users to additional item categories of the online shop or non-mainstream products as shown in [11] or [32]. Traditional accuracy measures should therefore be paired and contrasted with other possible quality aspects like diversity, serendipity or domain-specific measures to obtain a more realistic picture of the potential effectiveness of a method. Additionally, given that the different recommendation and contextualization strategies can lead to quite different recommendation lists corresponding, e.g., to short-term or long-term interests, a multi-list approach could be advisable to clearly distinguish the purpose of the recommendations. Research in this direction is unfortunately very limited so far.

Dataset limitations. The used dataset from Zalando only contains log data from a comparably limited time frame and only certain types of information about the items. In practice, however, additional factors like visitor demographics or seasonal aspects could be relevant for the success of the recommendation system. Furthermore, since our data was collected from a real-world web shop, the behavior of the users might be at least to some extent biased by unknown external factors. Nonetheless, since we could validate our findings on an additional dataset, we are confident that the effects of such unknown factors, if they exist, must be very limited.

6. RELATED WORK

Implicit ratings. Fueled by the existence of publicly available data sets, most of today's RS research is based on explicit rating information [17]. There are, however, a number of recent approaches that focus on processing implicit unary or binary relevance feedback, e.g., [14] or [26]. In our work, we use Rendle et al.'s BPR method [25] as a baseline technique in particular because it has characteristics of recent learning-to-rank methods.

Context-aware and time-aware recommendations (CARS). The goal of CARS is to incorporate information about the user's recent situation – such as short-term shopping goals – or other environmental conditions into the recommendation process. As discussed in [6], the connection between contextual factors and item selection can be hard to assess. According to the classification from [1] our methods fall in the category of "post-filtering" approaches. Context-enriched benchmark datasets are rare and time-stamp information is often the only additional source of context. Existing works that rely on time-stamp information in web log data include [5] and [19]. For a recent overview on various forms to evaluate such time-aware RS, see [7]. In our work, the log entries are sorted in chronological order but no explicit time-stamps are available. Therefore, we can only reason about possible effects related to the relative recency of the events. With the availability of explicit time-stamps, additional behavioral patterns could potentially be detected in the data using the above-mentioned methods.

Short-term interests. Information about the user's short-term interests is typically not explicitly given and has to be estimated based on the recent actions as done in our work. While short-term interests seem to play a minor role in RS research – some exceptions are [22] or [27] – there exists a number of recent works in the information retrieval field. Especially for news recommendation, short-term interests are a current research topic. Similar to our approach, the news recommender in [21] adapts the results of a collaborative filtering approach to the user's current interests with the content-based information of the recent search behavior. Some approaches, e.g., [2] or [3], use clustering to identify common navigation patterns in the log data and apply CF or association rule mining to match the recent history of a target user. In our work, a combination of short- and long-term interests is used to generate recommendation. Alternative combination methods are discussed in [4] and [23]. The domain of fashion products is explicitly addressed in [29] with a scenario-based modeling technique. The recent approach in [31] also focuses on fashion products and aims to identify the theme of a user's session by using factored Markov decision processes (fMDPs). In [12], short-term goals are modeled by a multi-armed bandit algorithm that detects a shift in the user's interest based on implicit feedback signals. In contrast to our work, these two approaches focus only on the short-term user goals whereas our method combines long-term user model with short-term interests.

Evaluation aspects. A number of factors determine the success of an RS in practice, e.g., the quality perceived by users [8, 9] or the correspondence of the recommendations with the application goals [16]. In fact, some studies suggest that methods that are optimized for high predictive accuracy on historical data do not always work best with regard to the desired effects on the users [15, 18]. [28] frames this evaluation challenge as a multi-objective decision problem where user requirements, business models, and technical constraints should all be taken into account in parallel. Our evaluations so far are limited to the usual accuracy metrics. In order to obtain a better assessment of the actual effectiveness, a multi-dimensional analysis has to be done provided that suitable data for such an evaluation is available.

7. SUMMARY

The goal of this work was to analyze the importance and quantify the effects of issuing recommendations according to long- and short-term shopping goals in e-commerce sites. We evaluated different short-term recommendation strategies which can be found on modern e-commerce sites and combined them with state-of-the-art techniques for long-term user profiling. Our experiments were based on a time-based and session-based evaluation protocol to show that short-term adaptations can be crucial to be able to make accurate recommendations according to short-term shopping goals. Furthermore, the results indicate that combining optimized long-term models with short-term strategies leads to the best overall results, i.e., long-term user models are not only suitable for generating non-contextualized recommendations on a shop's landing page but encode valuable knowledge that can be exploited for matching immediate short-term shopping goals. Our ongoing works include the analysis of additional characteristics of the recommendation lists – like diversity, catalog coverage, or popularity biases – and the development of additional algorithmic approaches for short-term adaptations.

8. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Context-aware recommender systems. In *Recommender Systems Handbook*, pages 217–253. 2011.

[2] S. R. Aghabozorgi and T. Y. Wah. Recommender systems: Incremental clustering on web log data. In *Proc. ICIS '09*, pages 812–818, 2009.

[3] Y. AlMurtadha, N. B. Sulaiman, N. Mustapha, N. I. Udzir, and Z. Muda. ARS: Web page recommendation system for anonymous users based on web usage mining. In *Proc. ECS '10*, pages 115–120, 2010.

[4] S. Anand and B. Mobasher. Contextual recommendation. In *From Web to Social Web*, volume 4737 of *LNCS*, pages 142–160. 2007.

[5] L. Baltrunas and X. Amatriain. Towards time-dependant recommendation based on implicit feedback. In *Proc. CARS WS at RecSys '09*, 2009.

[6] L. Baltrunas, B. Ludwig, and F. Ricci. Context relevance assessment for recommender systems. In *Proc. IUI '11*, pages 287–290, 2011.

[7] P. G. Campos, F. Díez, and I. Cantador. Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. *UMUAI*, 24(1-2):67–119, 2014.

[8] P. Cremonesi, F. Garzotto, S. Negro, A. Papadopoulos, and R. Turrin. Looking for "good" recommendations: A comparative evaluation of recommender systems. In *Proc. Interact '11*, pages 152–168, 2011.

[9] P. Cremonesi, F. Garzotto, and R. Turrin. Investigating the persuasion potential of recommender systems from a quality perspective: An empirical study. *ACM TIST*, 2(2):11:1–11:41, 2012.

[10] P. Cremonesi, Y. Koren, and R. Turrin. Performance of algorithms on top-n recommendation tasks. In *Proc. RecSys '10*, pages 39–46, 2010.

[11] M. B. Dias, D. Locher, M. Li, W. El-Deredy, and P. J. Lisboa. The value of personalised recommender systems to e-business: A case study. In *Proc. RecSys '08*, pages 291–294, 2008.

[12] N. Hariri, B. Mobasher, and R. Burke. Context adaptation in interactive recommender systems. In *Proc. RecSys '14*, pages 41–48, 2014.

[13] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM TOIS*, 22(1):5–53, 2004.

[14] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *Proc. ICDM '08*, pages 263–272, 2008.

[15] D. Jannach and K. Hegelich. A case study on the effectiveness of recommendations in the mobile internet. In *Proc. RecSys '09*, pages 205–208, 2009.

[16] D. Jannach, L. Lerche, F. Gedikli, and G. Bonnin. What recommenders recommend - an analysis of accuracy, popularity, and sales diversity effects. In *Proc. UMAP '13*, pages 25–37, 2013.

[17] D. Jannach, M. Zanker, M. Ge, and M. Gröning. Recommender systems in computer science and information systems - a landscape of research. In *Proc. EC-Web '12*, pages 76–87, 2012.

[18] E. Kirshenbaum, G. Forman, and M. Dugan. A live comparison of methods for personalized article recommendation at Forbes.com. In *Proc. ECML/PKDD '12*, pages 51–66, 2012.

[19] Y. Koren. Collaborative filtering with temporal dynamics. In *Proc. KDD '09*, pages 447–456, 2009.

[20] L. Lerche and D. Jannach. Using graded implicit feedback for bayesian personalized ranking. In *Proc. RecSys '14*, pages 353–356, 2014.

[21] J. Liu, P. Dolan, and E. R. Pedersen. Personalized news recommendation based on click behavior. In *Proc. IUI '10*, pages 31–40, 2010.

[22] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Using sequential and non-sequential patterns in predictive web usage mining tasks. In *Proc. ICDM '02*, pages 669–672, 2002.

[23] Q. N. Nguyen and F. Ricci. Long-term and session-specific user preferences in a mobile recommender system. In *Proc. IUI '08*, pages 381–384, 2008.

[24] S. Rendle. Factorization machines with libFM. *ACM Transactions on Intelligent Systems Technology*, 3(3):57:1–57:22, 2012.

[25] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *Proc. UAI '09*, pages 452–461, 2009.

[26] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proc.WWW '10*, pages 811–820. ACM, 2010.

[27] F. Ricci, A. Venturini, D. Cavada, N. Mirzadeh, D. Blaas, and M. Nones. Product recommendation with interactive query management and twofold similarity. In *Proc. ICCBR '03*, pages 479–493, 2003.

[28] A. Said, D. Tikk, K. Stumpf, Y. Shi, M. Larson, and P. Cremonesi. Recommender systems evaluation: A 3D benchmark. In *Proc. RUE WS at RecSys '12*, pages 21–23, 2012.

[29] E. Shen, H. Lieberman, and F. Lam. What am I gonna wear?: Scenario-oriented recommendation. In *Proc. IUI '07*, pages 365–368, 2007.

[30] Y. Shi, A. Karatzoglou, L. Baltrunas, M. Larson, and A. Hanjalic. xCLiMF: optimizing expected reciprocal rank for data with multiple levels of relevance. In *Proc. RecSys '13*, pages 431–434, 2013.

[31] M. Tavakol and U. Brefeld. Factored MDPs for detecting topics of user sessions. In *Proc. RecSys '14*, pages 33–40, 2014.

[32] M. Zanker, M. Bricman, S. Gordea, D. Jannach, and M. Jessenitschnig. Persuasive online-selling in quality & taste domains. In *Proc. EC-Web '06*, pages 51–60, 2006.