Re-ranking Recommendations Based on Predicted Short-term Interests – A Protocol and First Experiment

Dietmar Jannach and **Lukas Lerche** and **Matthäus Gdaniec**
Technische Universität Dortmund
Joseph-von-Fraunhofer-Straße 23
44227 Dortmund
Germany

Abstract

The recommendation of additional shopping items that are potentially interesting for the customer has become a standard feature of modern online stores. In academia, research on recommender systems (RS) is mostly centered around approaches that rely on explicit item ratings and long-term user profiles. In practical environments, however, such rating information is often very sparse and for a large fraction of the users very little is known about their preferences. Furthermore, in particular when the shop offers products from a variety of categories, the decision of what should be recommended can strongly depend on the user's current short-term interests and the navigational context.

In this paper, we report the results of an initial experimental analysis evaluating the predictive accuracy of different contextualized and non-contextualized recommendation strategies and discuss the question of appropriate experimental designs for such types of evaluations. To that purpose, we introduce a parameterizable protocol that supports session-specific accuracy measurements. Our analysis, which was based on log data obtained from a large online retailer for clothing and lifestyle products, shows that even a comparably simple contextual post-processing approach based on product features can leverage short-term user interests to increase the accuracy of the recommendations.

Introduction

The automated recommendation of additional items of interest during a customer's shopping session is a pervasive feature of most modern online stores. In many cases, such recommendation lists are varied depending on the navigational situation of the user.

The need for adapting the recommendation strategy to the current user's shopping goal and short-term interests is particularly obvious when the online store features a variety of different product categories (e.g., books, electronics, and groceries). However, even for a single-product shopping platform, a context-dependent adaptation based on short-term interests might be appropriate, see (Ricci et al. 2003) or (Jannach and Hegelich 2009). In general, we can observe an increasing interest in context-aware recommender systems (RS) in recent years. The consideration of the user's current browsing context in the recommendation process, however, only plays a minor role in today's mainstream RS research, even though there exist a number of works, e.g., in areas such as predictive Web usage mining, which use the user's current navigation path for intelligent Web site adaptation, see, e.g., (Mobasher et al. 2002a). One major reason for the lack of studies in that direction lies in the limited availability of public and comparable reference data sets.

Another aspect to be considered when developing real-world recommendation systems is the fact that in many domains the size of the product catalog is huge and at the same time the amount of explicit item ratings is extremely low or that no explicit ratings are available at all. Therefore, the recommendation and personalization process has to be based solely on implicit customer feedback such as purchase and shopping cart actions or Web log data on item views. While there exist a number of recommendation algorithms that work on implicit rating information – including recent ones such as Bayesian Personalized Ranking (BPR) (Rendle et al. 2009) – more research, for example on the interpretation of different types of implicit feedback, is required.

The research reported in this paper is based on such a real-world problem setting where the goal is to provide item recommendations for a large online store for clothing and lifestyle products. Instead of explicit ratings for items, the data set made available by our industrial partner Zalando[1] contains information about the past purchases of a customer as well as session log information such as item views or cart actions and a very limited amount of information about item features.

Following the above discussion, the intuition is that taking into account the user's session-specific short-term interests should help to predict more accurately what the customer will finally purchase. As the commonly used evaluation protocols cannot be directly applied to measure the effects of this contextualization, we propose to use a parameterizable protocol that supports session-specific accuracy measurements. To test our hypothesis, we implemented a number of comparably simple recommendation strategies which post-process the recommendations of an underlying recommendation algorithm depending on the current session. Our first results indicate that already quite simple strategies can uti-

[1] http://www.zalando.de/

lize the information about short-term interests to measurably increase the accuracy of the recommendations.

Overall, we see our work as a further step toward alternative approaches for the evaluation of recommendation algorithms, which can be applied to a larger class of realistic problem settings.

Data set and evaluation protocol

Data set characteristics and sub-sampling

The data set used in the analysis consists of log entries from our partner's e-commerce site. The entries have been anonymized and were artificially distorted[2]. Each of the about 9 million time-ordered log entries corresponds to one of four different user actions on an item: "view", "put in cart", "purchase" and "add to wish list". These actions are grouped into shopping *sessions* with unique IDs. The users themselves are not identified as logged-in users; instead, their identity is approximated through a browser cookie. Thus, it is not possible to exactly distinguish individual users who share a computer. For each item, basic category information is available, e.g., if the item is for men, women or kids or if the item is a shirt or a pair of shoes.

The actual purchase data is extremely sparse. The data set contains about 380,000 purchase transactions from about 100,000 users. Furthermore, 60,000 additional users have viewed items but never purchased anything. The catalog of purchased items including item versions in different colors is over 85,000. The resulting sparsity level is about $4.4 \cdot 10^{-4}$. The majority of the log entries consist of item views (about 8 million entries). There are about 650,000 cart actions which is two times more than there are actual purchases. Finally, about 90,000 items have been placed in wish lists.

The log entries are the result of about 570,000 user sessions. On average, there are about 16 user actions per session and most of them (about 14) are non-purchase actions such as item views. The percentage of sessions which resulted in a sales transaction is about 22 percent[3].

In our experimental evaluations our goal was to predict purchase actions for the current user session. In order to analyze the impact of data availability on the recommendation accuracy, we created three different subsets of the original data. For each subset, we applied different density constraints on the minimum number of purchase transactions per user and per item. At the same time, we tried to keep the overall number of purchases constant across the subsets.

Table 1 summarizes the characteristics of the data sets used in our experiments. For each data set, different density constraints were applied and a corresponding set of randomly chosen users and items was selected. Since we use a special type of time-based criterion to split training and test data as described in the next section, we created five subsamples with similar characteristics in order to be able to repeat the experiments and factor out random effects. Correspondingly, the table shows the average characteristics of each sub-sample.

	Sparse	Medium	Dense
Users	5,000	3,400	1,400
Items	12,600	4,200	2,300
Purchases	18,600	21,000	18,300
Views	125,400	87,800	29,800
Avg. sessions/user	3.5	4.1	4.3
Avg. session length	8.3	8.0	8.0
Pop. distr. (Gini)	0.457	0.415	0.334
Min. purchases/user	1	3	5
Min. purchases/item	1	3	5

Table 1: Characteristics of data sets used in the experiments

The three subsets are relatively small when compared with the original dataset. The reason for this is that our aim was to have a comparable number of purchases in each sample and applying even modest density constraints as shown in the table led to a strong reduction of the data sets.

Evaluation protocol and metrics

In our work, we focus on the accuracy of the recommendations produced by a recommender system and for the moment neglect other possible aspects to be evaluated such as the diversity or novelty of the recommendations. The standard protocol for evaluating the predictive accuracy of an RS when only implicit rating information is given is to split the data in training and test sets, generate a top-N recommendation list for each user and measure precision, recall or some ranking-based metric in several cross-validation runs.

Measuring precision and recall Given the large item catalog and the small number of purchases per user, we rely on variants of the precision and recall metrics as proposed in (Cremonesi, Koren, and Turrin 2010). Instead of taking the top-N list from the ranked list of all unseen items and counting the number of "hits" (actual purchases), each purchased item in the test set is evaluated independently by combining it with k other catalog items which the user has not purchased[4]. The task of the recommender is then to rank these $k+1$ elements. From this ranked list, we take the top N elements and check if the target element is contained in this list. The recall for each item is therefore either 0 or 1. Precision can correspondingly be computed as $1/L \cdot recall$, where L is the recommendation list length. We repeat the procedure for all users and all test items and then calculate the overall average as the final recall value.

We use this procedure for different reasons. First, the number of purchases per user is very low and for most users,

[2]This makes it impossible to infer customer data or business figures of our partner.

[3]The observed user behavior might to some extent be biased by the recommendation system, which already exists on the platform. This might in turn have an impact on the observed performance of different recommendation algorithms. However, since our goal is to measure the impact of contextualization on top of existing strategies, we consider this possible bias to be less problematic.

[4]Choosing a different number of random items k in our experiments only had an impact on the absolute recall values but not on the ranking of the algorithms. In our experiments, we use $k = 100$ since this value was large enough to highlight the differences in the results.

only a small number of items (e.g., 2 or 3) will remain in the test set, meaning that the calculation of top-5 or top-10 lists might not be appropriate. Furthermore, there are different ways in which precision and recall can be measured. Either we only count elements for which we know the ground truth or we leave all items in the ranked list. The first option cannot be chosen as we only have "unary" ratings and there cannot be any false positives. Using the second option, we will end up with very tiny and hard-to-compare precision values given the huge item catalog. For a further discussion of potential problems when directly applying common information retrieval measures such as precision and recall, see, e.g., (Herlocker et al. 2004).

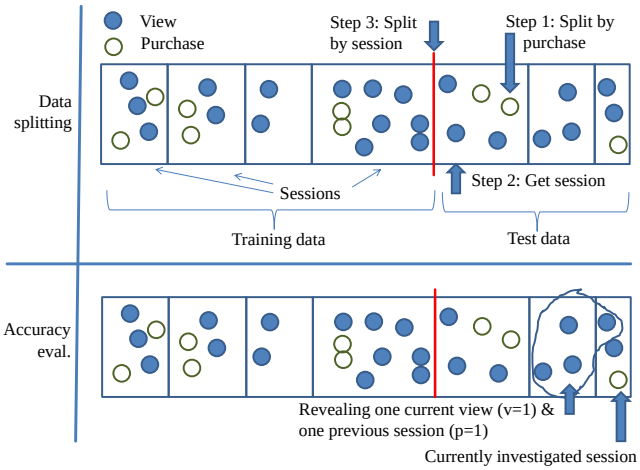

Figure 1: Data splitting ans session-wise evaluation

Creating training and test splits. The main hypothesis of our work is that is favorable to adapt the recommendations according to the user's short-term interests in the current and possibly some previous sessions. We therefore propose the following parameterizable experimental procedure, which we believe is more realistic for our problem situation and at the same time is not too specific so that it can be applied in other comparable problem settings, too.

1. As a first step, we split the purchase transactions into a preliminary training and test set (e.g., 90% and 10% respectively) based on the time of the purchase. The test set for each user has to contain at least one purchase.
2. Next, we pick the first purchase entry in the test set – the order of entries is still time-based – and retrieve the corresponding session ID.
3. This session ID is then used to do the final training and test split. Every event (view, purchase, etc.) that happened in this session or a later session, will be part of the test data, including further purchase transactions of these sessions. All other log entries represent the training data.

Figure 1 visualizes this procedure. When splitting the data that way, it may happen that no data remains in the training set. This in particular happens if there is only one single session in which the user made a purchase and this happened in his or her first (and perhaps only) visit of the web shop. In that case, a recommender cannot build a personal profile but rather has to rely on other strategies and for example recommend only popular items.

Contextualized recommendations and evaluation The last step in the protocol is the actual generation of the recommendation list and the measurement of recall as described above. Instead of taking the training data and generating one single recommendation list, we generate such a list for every session in the test set that contains at least one purchase transaction and measure the recall for every purchase in that session. Thus, in the example in Figure 1, no recommendation list would be generated for the second to last session since it contains only views.

The evaluation protocol has two parameters:

1. The amount of knowledge provided to the recommender about the current session (Parameter v). As we assume that the current user's goal determines what should be recommended, we can vary what we "reveal" to the recommender. In our experiments, we for example revealed information about the first $v = 2, 5$ and 10 item views of the session and also tested situations where only information about the previous sessions was used ($v = 0$ and $p > 0$, see below).
2. The number of previous sessions whose view actions are made visible to the recommender (Parameter p). One of our assumptions is that there might be situations, where the customer wants to continue his previous shopping session, in which she or he has viewed some items but made no purchase. With the parameter p, we vary the number of previous user sessions that are revealed to the recommender.

If we assume that $v = 1$ and $p = 1$ in the example in Figure 1, the prediction for the last session can be based – in addition to the training data – on the item views in the second to last session and the first view in the last session.

Note that whenever we use the term "context" in this paper, we refer to the revealed short-term interests of an user consisting of the v views from the current session and the views from p previous sessions for that specific user. Therefore, we call a recommendation strategy or recommender "contextualized" when it makes use of this additional information about the user's short-term interests.

In our experiments, we have not included "cart" and "wish list" actions neither in the training nor recommendation process. These types of user actions are strong indicators for an increased customer interest in certain products and would definitely help us to improve the prediction accuracy, in particular as purchase transactions are preceded by cart actions and views. However, we believe that recommending items that the user has already put in the shopping cart would be not particularly helpful for the user or even raise user doubts about the quality of the recommendations in general. One option would be to implement a time-based method, which re-adds an item to the set of recommendable items after a number of sessions, if the user has not purchased it in the meantime. Such an evaluation is however beyond the scope of our current work.

Finally, we are also not dealing with repeated purchases of the same product at the moment.A particularity of the domain is that customers sometimes order several variants of a product (e.g., in different sizes), of which they return later on all but one. Information about item returns is however not available in our data set.

Recommendation strategies

In order to test our hypothesis that even comparably simple approaches of contextualized filtering or item re-ordering based on short-term interests can lead to higher prediction accuracy than using non-contextualized approaches, we have tested a number of strategies on our data set.

Non-contextualized Baseline Strategies

- POPRANK: Popularity-based approaches can represent a comparably hard baseline (Cremonesi, Koren, and Turrin 2010). We implemented an unpersonalized baseline strategy that ranks the items based on the number of times they have been viewed or purchased in the training set. Our current baseline scheme does not differentiate between these two types of implicit feedback so far; an analysis if different weighting schemes lead to better results is part of our ongoing work.
- BPR: Approaches based on matrix factorization (MF) and learning-to-rank techniques represent the most successful classes of methods to build highly accurate recommender systems in recent literature. Therefore, and since only implicit customer feedback is available, we use BPR (Bayesian Personalized Ranking) (Rendle et al. 2009) in combination with the MF learning model as a state-of-the-art baseline in our experiments. Again, both views and purchases were used as implicit feedback[5]. A recent analysis in (Jannach et al. 2013) also showed the superiority of the approach compared to other (MF) techniques and simple popularity-based approaches in particular when precision is measured as described in the previous section.
- CONTENTPOP: This is another baseline strategy that combines a content-based approach with popularity information. The user profiles for the content-based part were created as follows. For each item property, e.g., brand or color, we calculated the distribution of values per user (importance weight) based on the shop actions. An example user profile could look like this: [User: Alice [color: 61% blue, 27% black, ..] [brand: 34% Nike, ...]]. The similarity between items and users for a set of properties P is calculated as $sim(u,i) = \sum_{p \in P} weight(u,i,p)/|P|$, where $weight(u,i,p)$ returns the user's importance weight for the particular property value of item i. If, for example, the item to rank is black, the function would return 27% for Alice using the example profile above. In the recommendation step, the score of an item for a given user is determined by weighting the similarity values with the popularity values from POPRANK as follows: $score(u,i) = r_i \cdot sim(u,i)$, where r_i is the value returned by POPRANK[6].

Contextualized approaches

The following strategies take the user's short-term interests into account by using their actions in the current and some previous sessions as described in the previous section.

- COOCCUR: Recommending products based on their co-occurrence in shopping carts is a classical, often non-personalized approach to build RS in e-commerce. In our experiments, we included a corresponding technique, which can be used to generate item-dependent recommendations of the popular style "users who viewed this item also viewed ...". Specifically, instead of using a fully-fledged and computationally intensive association rule mining approach (Sarwar et al. 2000), we limited ourselves to simpler co-occurrence patterns of size two in the training set and calculated the conditional probabilities that one item is viewed or purchased in the same session given another one. To recommend items in the context of a given session of the test set, a top-N list of items with the highest probability values is generated given the items that have been viewed in the current user context. In the future we are planning to combine COOCCUR or association rules with non-contextualized techniques, in particular BPR.
- FEATUREMATCHING (FM): For this approach, we used the recommendation lists generated by the best-performing non-contextualized algorithm BPR as well as POPRANK as a starting point and reordered the recommendation lists based on their feature overlap with items that have been viewed in the current context[7]. We therefore computed a "short-term user profile" in which we recorded the observed feature values. For the "brand" feature, for example, the short-term profile would contain the values *Nike* and *Puma* if the user has viewed only items of these manufacturers in his current context. Each recommendable item in the given recommendation list is then compared with the short-term profile and obtains a score based on the number of overlapping features. As we considered four different item features, each item can contain a score between 0 and 4. However, restricting to just two features (category and brand) yielded the best results. The given recommendation list was then re-ordered according to this score. For items with identical scores, the original ranking was retained.

  We also tried some simple variations of this strategy which combine the matching score and the BPR score and re-order the recommended items accordingly. The underlying idea was to avoid that originally low-ranked items are placed at the top of the list only based on the overlap score. However, these variations did not perform as good as the plain FEATUREMATCHING strategy and we therefore do not report the results here.

[5]We also made experiments with the FUNK-SVD approach, which however led to much poorer results.

[6]The content-based approach alone did not perform well.

[7]This corresponds to cascading, post-filtering approaches according to the classification of (Adomavicius and Tuzhilin 2011).

Results

We evaluated the different algorithms on data set subsamples as shown in Table 1 using the particular technique to measure recall ($k = 100$) as described in the previous sections. Furthermore, we varied the number of revealed item views v and previous sessions p to analyze to which extent adding more information about the current session context can help to improve the recommendations. The results obtained when using recommendation lists of size 10 are shown in Tables 2, 3 and 4.

	v=0, p=2	v=2, p=2	v=5, p=2	v=10, p=2	v=5, p=0
POPRANK	0.13				
CONTENTPOP	0.14				
BPR	0.50				
COOCCUR	0.28	0.32	0.36	0.38	0.27
POPRANK+FM	0.33	0.65	0.75	0.83	0.70
BPR+FM	0.60	0.73	0.80	**0.86**	0.78

Table 2: Recall results for the sparse (1-1) data set

	v=0, p=2	v=2, p=2	v=5, p=2	v=10, p=2	v=5, p=0
POPRANK	0.14				
CONTENTPOP	0.16				
BPR	0.57				
COOCCUR	0.29	0.38	0.43	0.46	0.35
POPRANK + FM	0.34	0.67	0.78	0.83	0.73
BPR+FM	0.64	0.77	0.84	**0.88**	0.82

Table 3: Recall results for the medium (3-3) data set

	v=0, p=2	v=2, p=2	v=5, p=2	v=10, p=2	v=5, p=0
POPRANK	0.08				
CONTENTPOP	0.10				
BPR	0.47				
COOCCUR	0.26	0.32	0.36	0.39	0.29
POPRANK + FM	0.30	0.64	0.73	0.81	0.68
BPR+FM	0.57	0.71	0.79	**0.84**	0.74

Table 4: Recall results for the dense (5-5) data set

Note that the absolute values for recall cannot be compared across the different data sets in Tables 2, 3 and 4 as the data sets have different characteristics. The medium data set, for example, contains the highest number of purchase transactions, which are the target of our measurement of recall. Table 1 furthermore shows that the Gini index for the sparse data set is the highest among all data sets. The Gini index is used here to measure the concentration of transactions on certain items as described in (Zhang 2010). When the index is higher, the "long tail" of unpopular items is longer. At the same time, it becomes easier for popularity-biased methods (including BPR) to properly rank items. The dense data set, finally, is comparably small which leads to lower absolute recall values, because e.g., the COOCCUR strategy cannot generate many rules.

Non-contextualized techniques. The first three rows show the results for recall for the non-contextualized techniques. Using the simple popularity-based approach is the weakest method according to this metric and we can observe that combining popularity information with the small amount of available content information can already improve the performance. BPR, as expected, clearly outperforms both basic strategies.

Contextualized techniques. The fourth row in the tables shows the recall values for the COOCCUR method, which uses co-occurrence patterns in the training data and the current session context to select the applicable rules. The results show that despite the high data sparsity and the huge number of catalog items, the simple method can at least outperform the popularity-based approach, but not the BPR method.

The final rows of the tables show the results of the different contextualized post-processing strategies. POPRANK + FM combines popularity based ranking with the FM reordering strategy. We can observe that this simple contextualization strategy starts to perform much better than the uncontextualized BPR method when there are at least $v = 2$ views from the current session revealed. Combining BPR with FM consistently leads to the best results across all data sets and experiment configurations. Again, remember that the FM strategy can be seen as an ad-hoc approach that in addition can only rely on coarse category information, which means that more sophisticated algorithms should easily lead to even better results.

When comparing the results for different values of v, we can observe that revealing more of the current session as expected makes it easier for the recommender to predict what will actually be purchased. The obtained results of course have to be analyzed with care as purchased items are typically also viewed in a session. The interesting aspect in our view, however, is that already a very small amount of information about the current session (and the previous session) helps to increase the accuracy. We also ran experiments where we did not reveal item views of the current session ($v = 0$) but only view actions of some previous sessions. Looking at the first and second result column with fixed $p = 2$ shows – especially for POPRANK – that revealing views from the current sessions strongly increases the recall. Compared to the third and fifth result column where the number of revealed views is fixed at $v = 5$, the addition of previous sessions does not increase the recall as much.

To estimate the value of utilizing context and content information, we finally compared the contextualized approaches with a non-contextualized variation of BPR + FM that uses the user's whole history from the training data – instead of recent actions – as "context". Interestingly, the results for this algorithm were about on par or even sightly better than when using a broad context ($v = 10$, $p = 2$). We see this as an indication that our re-ordering strategy in fact works well in this domain even when no contextualization based on short-term interests is done. The development of optimized algorithms for this data set and domain was how-

ever not in the focus of this paper, in which we are more interested in possible accuracy improvements based on context information.

Finally, we also measured the mean reciprocal rank (MRR) (Voorhees 1999) for all purchases in the test set using the session-specific recommendation lists. We omit the results due to space constraints as they follow the same overall trend as the recall. As expected however, the BPR-based strategies led to higher MRR-values as they are optimized for ranking.

Related work

The work presented in this paper is related to a number of topics in past RS research, ranging from the usage of implicit feedback and shopping-basket analysis, over sparse-data situations to context-awareness, hybridization and evaluation approaches.

With respect to the available customer feedback, most of today's research in RS is based on explicit rating information, which is in particular fueled by the existence of corresponding publicly available data sets (Jannach et al. 2012). There are, however, a number of recent approaches that focus on algorithms that can process implicit unary or binary relevance feedback, among them the works by (Koren 2008), (Hu, Koren, and Volinsky 2008), (Pan et al. 2008), or the BPR-method by (Rendle et al. 2009). In our work, we used the recent BPR method as a baseline technique and only included the available "positive" feedback from item views and purchases. Both our approach as well as many previous techniques consider all types of feedback to be equal. In fact, an item view might be less relevant than a purchase and a longer viewing time might be a stronger indicator for the user's interest. In our ongoing work, we therefore aim to evaluate if considering different feedback types in the learning process, as done in a simple form, e.g., in (Jannach and Hegelich 2009), can help us to further improve the recommendation accuracy.

Context-aware recommender systems (CARS) attracted increased attention in RS research in the last few years. CARS usually generate recommendations using additional and "externally" provided information such as the location or the time of the year. However, there is a lack of available benchmark data sets, with time-stamp information often being the only source of information (Campos, Díez, and Cantador 2013). The survey gives a recent overview of time-aware RS and shows that even in environments with time information being the only context factor, there is no standard evaluation design yet. For recent research in CARS see, e.g., (Koren 2009) for an algorithmic approach to exploit time-stamp information in RS and (Baltrunas and Ricci 2009) for a way to incorporate generic context information in standard CF. Our work, in contrast, proposes a comparably simple method of incorporating the *immediate* short-term interests of the user and manipulating the results obtained from an un-contextualized model. The short-time interest can also be considered as a contextual factor, however, in most cases it is not directly available in the data. Instead, we assume that such interests can be derived from the user's recent actions. In some sense, our approach is similar to the one of (Ricci et al. 2003) who use the user's current query as a short-term interest profile and combine the query-based approach with collaborative features. Instead of filtering assumedly non-relevant items, we however focus more on re-ranking. According to the classification from (Adomavicius and Tuzhilin 2011), the method of this paper falls in the category of "post-filtering" approaches, in which recommendation lists are post-processed based on context information. In RS research, only a limited number of approaches such as (Ricci et al. 2003) or (Mobasher et al. 2002b) can be found that explicitly deal with short-term interests and are based, e.g., on constraints or navigation patterns. In the field of IR, however, a number of recent works exist that try to immediately adapt the search results based on the user's recent search behavior ((Bennett et al. 2012), (Liao et al. 2012), (White, Bennett, and Dumais 2010)).

The field of evaluating recommendation algorithms based on historical data is mostly dominated by IR measures such as precision and recall or error metrics often used in machine learning such as the RMSE. Other metrics that focus on evaluating ranked results have also been introduced in the past, for example the half-life measure (Breese, Heckerman, and Kadie 1998) and the NDCG (Järvelin and Kekäläinen 2002). In our work, we rely on a particular variant of determining the precision and recall metric which is appropriate for the given situation with implicit feedback, sparse data and a large item catalog. The comparison of the different algorithms based on metrics that also take the specific position of the recommended item in the list into account, is part of our ongoing work.

Summary and Conclusion

In this work, we have explored the usefulness of contextualized recommendations based on real-world web log data. We have first proposed a new test protocol that aims to model the session-centric user behavior which can be found in many e-commerce platforms. We then examined the performance of baseline and state-of-the art recommendation strategies and compared them with a post-processing technique that utilizes the user's context and short-term behavior.

Our results show that ranking optimization based on BPR and implicit feedback performs quite well compared to popularity- or content-based baseline algorithms. Using a comparably simple contextualized feature-matching strategy by incorporating the short-term user-behavior, we were able to further improve the accuracy of the top-N lists generated by BPR (and other strategies). Our measurements furthermore show that small amounts of additional information about short-term interests can be sufficient. More advanced techniques to exploit contextual information will obviously perform better than our ad-hoc item reordering strategy. Overall, however, we see our observations so far as a strong indicator that taking the user's current context can be crucial for the quality of recommendations in real-world applications.

The basis of our evaluation is a new test protocol which we see as another step towards more realistic offline experimental designs for recommender systems. We are aware that

our evaluation protocol in some sense is unfair as the contextualized recommenders have slightly more knowledge than the un-contextualized ones. The amount of additional information about item views is however comparably small. Another limitation of the approach is that our contextualized techniques might recommend items which the user has already seen in the current or last few sessions. For the user, such recommendations might appear redundant and of limited novelty. A deeper analysis is therefore required in that direction. Remember however, that the goal of this work was not to outperform methods such as BPR but to show that small amounts of extra information can help to further improve the performance of the underlying techniques.

Overall, web log data can represent a highly relevant and possibly the only information source for building RS for e-commerce in practice. Therefore, more research is required on how to handle the different types of customer feedback. Furthermore, we assume that in practice the availability of additional information about the users' exact identification, demographics, more detailed item information, exact navigation paths or search terms as well as temporal and seasonal aspects should further help to increase the prediction accuracy of real-world recommenders, in particular when the rating information is sparse.

References

Adomavicius, G., and Tuzhilin, A. 2011. Context-aware recommender systems. In Ricci, F.; Rokach, L.; Shapira, B.; and Kantor, P. B., eds., *Recommender Systems Handbook*. Springer. 217–253.

Baltrunas, L., and Ricci, F. 2009. Context-based splitting of item ratings in collaborative filtering. In *Proc. ACM RecSys 2009*, 245–248.

Bennett, P. N.; White, R. W.; Chu, W.; Dumais, S. T.; Bailey, P.; Borisyuk, F.; and Cui, X. 2012. Modeling the impact of short- and long-term behavior on search personalization. In *Proc. SIGIR '12*, 185–194.

Breese, J. S.; Heckerman, D.; and Kadie, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proc. UAI '98*, 43–52.

Campos, P. G.; Díez, F.; and Cantador, I. 2013. Time-aware recommender systems: A comprehensive survey and analysis of existing evaluation protocols. *UMUAI* forthcoming.

Cremonesi, P.; Koren, Y.; and Turrin, R. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Proc. ACM RecSys 2010*, 39–46.

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

Hu, Y.; Koren, Y.; and Volinsky, C. 2008. Collaborative filtering for implicit feedback datasets. In *Proc. ICDM 2008*, 263–272.

Jannach, D., and Hegelich, K. 2009. A case study on the effectiveness of recommendations in the mobile internet. In *Proc. ACM RecSys 2009*, 205–208.

Jannach, D.; Zanker, M.; Ge, M.; and Gröning, M. 2012. Recommender systems in computer science and information systems - a landscape of research. In *Proc. EC-WEB 2012*, 76–87.

Jannach, D.; Lerche, L.; Gedikli, F.; and Bonnin, G. 2013. What recommenders recommend - an analysis of accuracy, popularity, and sales diversity effects. In *Proc. UMAP 2013*.

Järvelin, K., and Kekäläinen, J. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.* 20(4):422–446.

Koren, Y. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proc. KDD 2008*, 426–434.

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

Liao, Z.; Song, Y.; He, L.-w.; and Huang, Y. 2012. Evaluating the effectiveness of search task trails. In *Proc. WWW '12*, 489–498.

Mobasher, B.; Dai, H.; Luo, T.; and Nakagawa, M. 2002a. Using sequential and non-sequential patterns in predictive web usage mining tasks. In *Proc. ICDM 2002*, 669–672.

Mobasher, B.; Dai, H.; Luo, T.; and Nakagawa, M. 2002b. Using sequential and non-sequential patterns in predictive web usage mining tasks. In *Proc. ICDM '02*, 669–. Washington, DC, USA: IEEE Computer Society.

Pan, R.; Zhou, Y.; Cao, B.; Liu, N. N.; Lukose, R.; Scholz, M.; and Yang, Q. 2008. One-class collaborative filtering. In *Proc. ICDM 2008*, 502–511.

Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proc. UAI 2009*, 452–461.

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

Sarwar, B.; Karypis, G.; Konstan, J.; and Riedl, J. 2000. Analysis of recommendation algorithms for e-commerce. In *Proc. EC 2000*, 158–167.

Voorhees, E. M. 1999. Trec-8 question answering track report. In *Proc. TREC-8*, 77–82.

White, R. W.; Bennett, P. N.; and Dumais, S. T. 2010. Predicting short-term interests using activity-based search context. In *Proc. CIKM '10*, 1009–1018.

Zhang, M. 2010. *Enhancing the diversity of collaborative filtering recommender systems*. PhD Thesis. Univ. College Dublin.