Bayesian Personalized Ranking for Novelty Enhancement

Jacek Wasilewski
Insight Centre for Data Analytics
University College Dublin
Dublin, Ireland
jacek.wasilewski@insight-centre.org

Neil Hurley
Insight Centre for Data Analytics
University College Dublin
Dublin, Ireland
neil.hurley@insight-centre.org

ABSTRACT

Novelty enhancement of recommendations is typically achieved through a post-filtering process applied on a candidate set of items. While it is an effective method, its performance heavily depends on the quality of a baseline algorithm, and many of the state-of-the-art algorithms generate recommendations that are relatively similar to what the user has interacted with in the past. In this paper we explore the use of sampling as a means of novelty enhancement in the Bayesian Personalized Ranking objective. We evaluate the proposed extensions on the MovieLens 20M dataset, and show that the proposed method can be successfully used instead of two-step reranking, as it offers comparable and better accuracy/novelty tradeoffs, and more unique recommendations.

ACM Reference Format:

Jacek Wasilewski and Neil Hurley. 2019. Bayesian Personalized Ranking for Novelty Enhancement. In *27th Conference on User Modeling, Adaptation and Personalization (UMAP '19), June 9–12, 2019, Larnaca, Cyprus.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3320435.3320468

1 INTRODUCTION

In the era of information and products overload, the task of browsing systems to find items of interest to users is becoming more difficult. Recommender systems play an important role in such scenarios as they filter a large set of selections into a much smaller set of items that a user is likely to be interested in. They do this by utilising implicit or explicit user feedback on items recorded by the system. While historically recommender systems were more often tasked with rating prediction, nowadays the task of ranking items seems to be more relevant, and a number of methods optimised for the ranking task have been proposed.

While it is important to serve accurate recommendations, there are other utilities beyond accuracy that have been identified as desired properties of a system. Novelty is one of them, and there are different notions of novelty. One notion expresses whether recommendations made to users are made of item types that the users are aware of. This is connected to the filter bubble problem that many of standard algorithms suffer from, where recommendations closely follow past interactions with the system, resulting in recommendations being not particularly engaging. By enhancing novelty of recommendations, it is possible to improve users' experience by widening the range of possible item types recommended to the user, previously not explored by the users.

UMAP '19, June 9–12, 2019, Larnaca, Cyprus

ACM ISBN 978-1-4503-6021-0/19/06…$15.00
https://doi.org/10.1145/3320435.3320468

A common way for novelty enhancement is to generate candidate recommendations and to rerank them in a post-filtering process, such that accuracy and novelty are balanced. However, the performance of this approach heavily depends on the quality of candidate recommendations — final recommendations will be as novel as the most novel subset of the candidate items — and state-of-the-art algorithms tend to recommend items similar to those consumed by the user in the past. As an alternative, algorithms can be modified to directly consider how similar or dissimilar items are while making recommendations.

In this paper, we modify the Bayesian Personalised Ranking (BPR) algorithm to make it aware of content relationships that exist between items. We achieve this through sampling of the training data, based on these relationships. The evaluation shows that the proposed solution offers novel and diverse recommendations at the same time, with similar performance as two-step post-filtering solutions. Additionally we show that the integrated approach offers recommendations much more varied than the baseline or the reranked baseline.

The rest of the paper is structured as follows. In the next section we give an overview of Bayesian Personalized Ranking. In section 3 we introduce a sampling scheme to account for item relationships. We describe the evaluation protocol and discuss the obtained results in section 4. We compare the proposed method with related existing solutions in section 5, and we finally conclude in section 6.

2 BAYESIAN PERSONALISED RANKING

Bayesian Personalised Ranking (BPR) [10] is one of many algorithms proposed to produce ranked recommendations. It infers rankings from implicit user feedback, and learns a matrix factorisation model (as one of its variations). It requires pairwise item preferences from which the model is learned. In systems with implicit user feedback, only positive user-item interactions are available, and the non-observed interactions are a mixture of users being not interested in certain items — negative feedback — or users being unaware of items — missing values. In this case, the method assumes that a user prefers positive interactions over non-observed interactions.

User-specific pairwise item preferences are triples of the form (u, i, j) which express that user u prefers item i over j. Given the set of available items $\mathcal{I}$, for each user $u \in \mathcal{U}$ we denote her/his positive interactions as $\mathcal{I}_u^+$, and by $(u, i) \in \mathcal{R}$ all user-item pairs of positive interactions. We derive the collection of preference relations, $D_S : \mathcal{U} \times \mathcal{I} \times \mathcal{I}$, i.e. the training data, in the following way:

$$D_S = \{(u,i,j)|(u,i) \in \mathcal{R} \wedge j \in \mathcal{I} \setminus \mathcal{I}_u^+\}\,.$$

Given the derived training set D_S, the optimisation criterion for the BPR algorithm is defined as

$$\text{BPR-Opt} = \arg\min_{\Theta} \sum_{(u,i,j)\in D_S} -\ln \sigma(\hat{r}_{uij}) + \lambda_\Theta ||\Theta||^2\,,$$

where Θ represents model parameters on which the $\hat{r}_{uij}$ prediction depends, λ_Θ are model regularisation parameters. The $\hat{r}_{uij}$ expresses the preference of user u between items i and j, which can be seen as a difference of predicted utilities: $\hat{r}_{uij} = \hat{r}_{ui} - \hat{r}_{uj}$. The σ function is the logistic sigmoid function that transforms the predictions: $\sigma(x) = \frac{1}{1+e^{-x}}$. The objective is being minimised, which happens when the difference between items i and j is large.

The matrix factorisation model is used to drive the objective, thus Θ corresponds to low-rank matrices $\mathrm{P} : |\mathcal{U}| \times k$ and $\mathrm{Q} : |\mathcal{I}| \times k$ that approximate the utility matrix of all user-item interactions, where k is the dimensionality of the approximation. Matrices P and Q represent, respectively, latent feature vectors describing users and items, and the $\hat{r}_{ui}$ prediction can be formulated as $\hat{r}_{ui} = \mathbf{p}_u^T \mathbf{q}_i$. Final recommendations are made by calculating $\hat{r}_{ui}$ for all user-item pairs and ordering them per user in a descending manner.

The Stochastic Gradient Descent (SGD) optimisation algorithm has been proposed to run the optimisation of the objective, based on uniform bootstrap sampling with replacement, however in other work, Rendle and Freudenthaler [9] pointed out that the training data is skewed towards popular positive items which dominate the gradient, making it harder to learn the model. To address the skewness of the training data, Rendle and Freudenthaler proposed an adaptive oversampling technique, where for each positive interaction, a negative item is sampled w.r.t. the estimated ranking for its context. A similar idea has been proposed in Weston et al. [15]. We use a simplified version of this, by selecting negative items such that they are close to violating the ranking implicit in the pattern of user interactions. This can be done by selecting negative items whose predicted score is high.

3 CONSTRAINING ITEMS SAMPLING

There are two main components of the BPR algorithm—derivation of the item preferences and the learning objective. Here we focus on the first one, the sampling process, to enhance the method and promote novelty.

A number of novelty notions exist, with two most commonly considered being: a) the long tail novelty [3, 8], where novelty is based on items popularity; and b) personalised unexpectedness [1], based on some distance between recommended items and items that we know the user is aware of:

$$\mathrm{PD}(\mathrm{R}_u) = \frac{1}{|\mathrm{R}_u|} \sum_{i\in \mathrm{R}_u} \mathrm{dist}(i, E_u)\,, \qquad (1)$$

where E_u represents a set of obvious items that the user would expect. Based on the assumption that items in the user profile, $\mathcal{I}_u$, are known and not unexpected by the user, the distance $\mathrm{dist}(i, E_u)$ can be further decomposed into $\mathrm{dist}(i, E_u) = \sum_{j\in \mathcal{I}_u} \mathrm{dist}(i, j)$ where $\mathrm{dist}(i, j)$ can be defined through e.g. item features. The above definition of novelty relies on pairwise dissimilarity, and there a natural way to incorporate such pairwise dissimilarities into the the BPR method.

The training data, D_S, that the BPR uses, has been constructed from positive interactions recorded by the system paired with unseen interactions. This approach ignores any relationship that exists between a pair of items. Item similarity/dissimilarity is one such relationship, used to express novelty of recommendations, as we see in Equation 1, and it is possible to extend the definition of the D_S set to make it aware of such relationships, by constraining the sampling. Given a measure $\mathrm{dist}(i, j)$ that expresses dissimilarity between items i and j, we can constrain the above set of pairs as follows:

$$D_S^{\text{dist}} = \{(u,i,j)|(u,i) \in \mathcal{R} \wedge j \in I \setminus \mathcal{I}_u^+ \wedge \mathrm{dist}(i,j) < \tau\}\,.$$

In the above definition, pairs of items whose dissimilarity is greater than the threshold τ are not selected. While this might be counter-intuitive, let's consider an example showing how the BPR learning objective operates on such pairs. Assuming that $\mathrm{dist}(i, j)$ takes a value on the range $[0, 1]$, we select threshold $\tau = 0.8$. This threshold splits the space of items into a similar/different, in addition to a positive/unseen categorisation. If two items have a dissimilarity larger than or equal to τ, we exclude that pair from the training data on the assumption that two dissimilar items are not considered as contrary items. Note that the BPR-Opt pushes away item pairs that are defined as contrary — that is, all item pairs in the dataset D_S or D_S^{dist}. Dissimilar items can be seen as two 'positive' items, thus the items that we want to separate in the learning process are the items that we consider similar.

In our definition of D_S^{dist}, we treat a pair of known items and unknown but dissimilar items, in the same way as a pair of two known and liked items. In reality, some of the unknown dissimilar items may be relevant to the user but that will not be the case for all items. Actually if we were randomly selecting pairs of items, and we knew the ground truth for all items, it is much easier to find a pair of dissimilar items that are not relevant, than a pair of relevant dissimilar items. If we base the process of learning user preferences purely on D_S^{dist}, we actually would learn disturbed user-item relations, which may have a negative effect on the accuracy of the model. Instead, we propose to sample from both datasets, the original D_S, and the one constrained by distance, D_S^{dist}. By doing that, we incorporate some additional signals on the dissimilar pairs of items, while respecting that these are not really two positive interactions. To control the strength of this effect, we propose to sample from both sets of triples, with probability β of taking a sample from D_S^{dist}.

In Figure 1 we present the learning procedure of BPR where distance-biased sampling is applied. The pseudocode shows how β is used to control the source of sampling — a random number r is selected uniformly on a 0-1 range, and based on that number, the dataset is selected.

4 EVALUATION

4.1 Dataset

To compare the effectiveness of novelty enhancement in the BPR setting, we perform evaluation on the MovieLens 20M (ML-20M) dataset[5]. The biggest MovieLens dataset consists of about 20M

function LEARNDISTBPR(D_S, D_S^{dist}, Θ, α, β, λ_Θ, λ)
  initialise Θ
  repeat
    r = random(0,1)
    draw $(u, i) \in \mathcal{R}$ uniformly
    if $r \leq \beta$ **then**
      draw j from $\mathcal{I} \setminus \mathcal{I}_u^+ \propto s(u, j)$ s.t. dist$(i, j) < \tau$
    else
      draw j from $\mathcal{I} \setminus \mathcal{I}_u^+ \propto s(u, j)$
    end if
    $\Theta \leftarrow \Theta - \alpha \left(\frac{-e^{-\hat{r}_{uij}}}{1+e^{-\hat{r}_{uij}}} \frac{\partial}{\partial \Theta} \hat{r}_{uij} + \lambda_\Theta \Theta \right)$
  until convergence
  return Θ
end function

Figure 1: BPR algorithm with distance-biased sampling.

ratings on a 0.5 to 5 scale, with a step-size of 0.5, from 138K users on 28K movies. Movies are enriched by 20 genres, however interactions for items without any genre information have been removed from the dataset. As the considered algorithm was designed for implicit datasets, we treat ratings of 5 as implicit positive feedback, and take the score of 1. The rest of the ratings are removed. The dataset has been split into training and test sets, with an 80:20 split.

4.2 Evaluation Protocol

In order to assess the considered method, we generate and evaluate ranked lists R_u of top $N = 20$ items for each user in the test set. Before testing novelty enhancements, the BPR algorithm has been tuned for λ_Θ to find the best baseline solution in terms of precision. The factorisation dimensionality has been set to $k = 20$, training is run for 100 iterations, the learning rate is $\alpha = 0.01$, and sample size in each iteration is equal to the size of the training data. As BPR depends on random sampling, we repeat the training/evaluation process 5 times and present the average performance.

We evaluate the proposed method (denoted by BPR_{dist}), and the impact of its parameters by varying $\beta \in [0, 1]$ and $\tau \in [0, 1]$. We compare the performance with the novelty reranker applied on the original BPR (denoted as BPR+PD). We generate a set C of 40 candidate items for each user, then we iteratively construct the reranked list by greedily selecting at each iteration item i that satisfies:

$$i^* = \arg \max_{i \in \mathrm{C} \setminus \mathrm{R}_u} (1 - \lambda) s(u, i) + \lambda \frac{1}{|\mathcal{I}_u|} \sum_{j \in \mathcal{I}_u} \text{dist}(i, j),$$

and updating $\mathrm{R}_u \leftarrow \mathrm{R}_u \cup \{i\}$ until $|\mathrm{R}_u| = N$. Parameter $\lambda \in [0, 1]$ controls the tradeoff between accuracy and novelty, and $s(u, i)$ is the score given by the baseline recommender.

We measure the performance in terms of the following metrics: precision, nDCG, EPD and EILD. A good overview of metrics can be found in [12]. Precision represents the fraction of relevant items in the results, nDCG measures relevance w.r.t. to an item's position, and ideal ranking list. The Expected Profile Distance (EPD) is an extension of Equation 1, which measures the average pairwise dissimilarity between items in a recommendation set, and previously consumed items, but can also take into account rank and relevance of recommended items. The EILD measures the average pairwise dissimilarity of items in a recommendation set, also with the possibility to consider rank and relevance. We apply logarithmic rank discount, however we use relevance-unaware versions of these metrics. To express item relationships, we use cosine dissimilarity between item genre profiles.

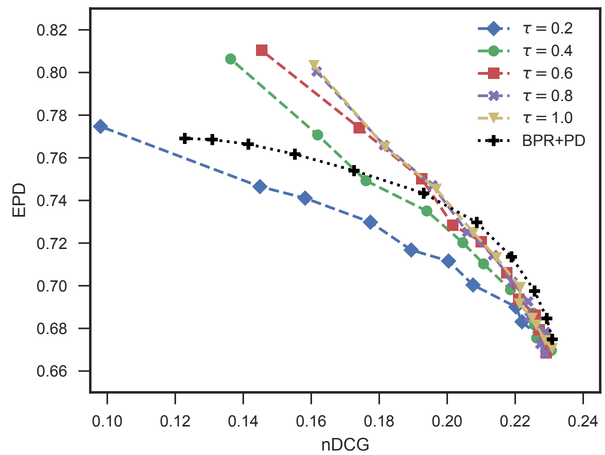

Figure 2: Tradeoffs between accuracy (nDCG) and novelty (EPD). For BPR_{dist}, various τ and β settings are presented, where for BPR+PD, λ is varied.

4.3 Results

Impact of method parameters. The distance-biased sampling method, BPR_{dist}, is controlled by two parameters, τ and β. To measure how these impact accuracy and novelty of recommendations, in Figure 2 we present the tradeoffs between accuracy (expressed through the nDCG metric) and novelty of different combinations of these two parameters — for each threshold we vary the β parameter. For clarity we plot the results only for thresholds $\tau \in \{0.2, 0.4, 0.6, 0.8, 1.0\}$ which mark the general trend, and other threshold settings follow these. Parameter β controls the fraction of the dataset that comes from the dataset constrained by item distances. The higher the value of the parameter, the more the method is trained on the constrained dataset D_S^{dist}, the higher novelty performance we could expect, which the results confirm.

The second parameter, τ, is used to split item pairs into those that we consider different and those that we consider similar. By exposing the BPR algorithm to pairs of similar items, we want to demote these items and push their representations apart, which results in recommendations made of more novel items. The results show that by picking higher thresholds, we obtain solutions that offer better novelty performance for the same accuracy, because more items are considered similar.

Comparison with novelty reranking. In Figure 2 performance of the BPR+PD reranker is shown, for different values of the blending parameter λ. It can be seen that for $\lambda < 0.5$ the rerankers offer slightly better accuracy/novelty tradeoff, however BPR_{dist} is really close to them. From that point onward, the BPR_{dist} outperforms BPR+PD, offering generally higher novelty scores, and better novelty performance for the nDCG scores obtained by BPR+PD. The above is generally true if a reasonable parameters of BPR_{dist} are

	λ	τ	β	**Prec.**	**nDCG**	**EPD**	**EILD**
BPR	-	-	-	<u>0.1165</u>	<u>0.2309</u>	0.6705	0.6399
+PD	0.5	-	-	0.1064	0.1931	$^{\triangle}$0.7434	$^{\blacktriangle}$<u>0.7353</u>
BPR_{dist}	-	0.6	0.8	0.1016	0.1925	$^{\blacktriangle}$<u>0.7502</u>	$^{\triangle}$0.7238

Table 1: Comparison of novelty enhancement methods. We have set a desirable performance, and we report one best setting where nDCG reaches the level of approximately 0.19**. For each metric, improvements over the baseline are denoted with $^{\triangle}$, best improvements with $^{\blacktriangle}$, and overall best are underlined. All differences with respect to the baseline are statistically significant (Wilcoxon $p < 0.05$).**

selected, $\tau \geq 0.4$. Lower threshold values do not distinguish items well.

In Table 1 we compare other metrics that we have measured, for one setting of the considered novelty enhancement methods, and the original BPR algorithm. We have selected settings that obtain the value of nDCG of approximately 0.19. For both methods, this also corresponds to precision of around 0.105. Similar nDCG and lower precision may mean that BPR+PD holds more relevant items higher in the ranked list. We will look into the qualitative analysis of recommendations later.

With respect to the BPR, both methods improve novelty expressed through the EPD metric, however BPR_{dist} seems to offer better tradeoff, with the score of 0.7502 compared to 0.7434 of BPR+PD. These correspond to improvements of 11-12% over the baseline.

Diversity of recommendations. Diversity of items is another distance-based utility of recommendations, and commonly improvements in novelty improve diversity and vice versa. We have looked into this aspect as well, and in Table 1 we report the EILD metric of diversity. We can observe that both methods boost diversity, with BPR+PD showing slightly better capabilities — improving by 15% over the baseline, where BPR_{dist} improves diversity by 13%. For the BPR_{dist} method improving diversity is expected as its objective is generally trying to separate similar items. In the case of BPR+PD, this is rather a side effect of the reranking.

Comparison of recommendations. Finally, we would like to compare the recommendations produced by all the methods, by comparing items that have been recommended to each of the users. We do that for the recommendations generated by settings reported in Table 1. We simply count the items that each pair of recommendations share, which follows the Inter-System Diversity metric of Bellogín et al. [2]. Results show that the reranked recommendations share with the baseline BPR, on average, 15 out of 20 items, where BPR_{dist} shares only 10 items with the baseline. Such huge overlap of items in the BPR+PD method could also explain slightly better tradeoffs we have discussed earlier, because for lower λ values even fewer items are replaced in the recommendations (and these are closer to the list end), making it easier to hold accuracy.

If in addition to previously considered settings, we also looked into settings that offered the most in terms of novelty of recommendations — $\lambda = 1.0$ for BPR+PD; $\tau = 0.6$ and $\beta = 1.0$ for BPR_{dist} — we can find that BPR+PD shares also only 10 items with the baseline, however as we have seen in Figure 2, this happens at much higher cost in terms of accuracy. The most 'novel' setting of BPR_{dist} shares only 7 items with the baseline. If we compare these different settings of the same methods with each other, we can find that BPR+PD recommendations share 15 items on average, where BPR_{dist} 10 items on average. All of the above tells us that BPR+PD makes recommendations much more similar (in terms of items used) to the original BPR method and that these lists vary less than those produced by BPR_{dist}. Together with the fact that the novelty performance of BPR_{dist} (depending on the choice of parameters) is comparable or better, we claim that the BPR with distance-constrained sampling process is better than the reranking approach, and it is a one-step approach.

5 RELATED WORK

Many improvements have been proposed to the BPR method since it was introduced. In [4], sampling has been modified to take into account items' popularity. Rendle and Freudenthaler [9] proposed a sampler that is context-dependent and over-samples informative pairs to speed up convergence. Loni et al. [7] modified the sampler to utilise different types of user feedback. Similarly, Lerche and Jannach [6] extended the BPR to deal with graded preference relations coming from implicit feedback, claiming that e.g. purchase of an item is a stronger feedback than a view. Additional pairwise preferences are derived from such information, and incorporated into the optimisation phase by choosing a proportion of samples coming from these additional preferences. We applied a similar pattern to mix constrained sampling and original sampling.

Wasilewski and Hurley [13, 14] have proposed methods to optimise other beyond-accuracy utility, diversity, also in one step, instead of post-filtering reranking. The stream of research presented in this paper falls into the same category of algorithms, however achieved through data sampling rather than reformulation of the objective. Also for diversity, Su et al. [11] has proposed a set-oriented formulation of BPR. Instead of training the model based on item pairs, whole sets of items are considered. Sets in the training collection are constrained by set relevance and set diversity. In this paper we apply a similar constraining, however on item pairs rather than item sets.

6 CONCLUSIONS

In this paper we looked into the problem of novelty enhancement of the Bayesian Personalized Ranking method. We proposed to constrain the sampling process of the algorithm, by utilising the content relationships between items. By doing so, we managed to improve novelty of recommendations produced by the method, without a post-filtering step proposed in the state-of-the-art. We showed that it is not only possible to improve novelty while training the BPR model, but also that the tradeoff between accuracy and novelty is comparable or better than post-filtering methods, and that final recommendations are more unique when compared to the recommendations produced by the BPR.

ACKNOWLEDGMENTS

This project has been funded by Science Foundation Ireland under Grant No. SFI/12/RC/2289.

REFERENCES

[1] Panagiotis Adamopoulos and Alexander Tuzhilin. 2014. On Unexpectedness in Recommender Systems: Or How to Better Expect the Unexpected. *ACM Trans. Intell. Syst. Technol.* 5, 4, Article 54 (Dec. 2014), 32 pages. https://doi.org/10.1145/2559952

[2] Alejandro Bellogín, Iván Cantador, and Pablo Castells. 2013. A Comparative Study of Heterogeneous Item Recommendations in Social Systems. *Information Sciences* 221 (2013), 142 – 169. https://doi.org/10.1016/j.ins.2012.09.039

[3] Òscar Celma and Perfecto Herrera. 2008. A New Approach to Evaluating Novel Recommendations. In *Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys '08)*. ACM, New York, NY, USA, 179–186. https://doi.org/10.1145/1454008.1454038

[4] Zeno Gantner, Lucas Drumond, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2011. Personalized Ranking for Non-uniformly Sampled Items. In *Proceedings of the 2011 International Conference on KDD Cup 2011 - Volume 18 (KDD-CUP'11)*. JMLR.org, 231–247.

[5] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (Dec. 2015), 19 pages. https://doi.org/10.1145/2827872

[6] Lukas Lerche and Dietmar Jannach. 2014. Using Graded Implicit Feedback for Bayesian Personalized Ranking. In *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys '14)*. ACM, New York, NY, USA, 353–356. https://doi.org/10.1145/2645710.2645759

[7] Babak Loni, Roberto Pagano, Martha Larson, and Alan Hanjalic. 2016. Bayesian Personalized Ranking with Multi-Channel User Feedback. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys '16)*. ACM, New York, NY, USA, 361–364. https://doi.org/10.1145/2959100.2959163

[8] Yoon-Joo Park and Alexander Tuzhilin. 2008. The Long Tail of Recommender Systems and How to Leverage It. In *Proceedings of the 2008 ACM Conference on Recommender Systems (RecSys '08)*. ACM, New York, NY, USA, 11–18. https://doi.org/10.1145/1454008.1454012

[9] Steffen Rendle and Christoph Freudenthaler. 2014. Improving Pairwise Learning for Item Recommendation from Implicit Feedback. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining (WSDM '14)*. ACM, New York, NY, USA, 273–282. https://doi.org/10.1145/2556195.2556248

[10] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI '09)*. AUAI Press, Arlington, Virginia, United States, 452–461.

[11] Ruilong Su, Li'Ang Yin, Kailong Chen, and Yong Yu. 2013. Set-oriented Personalized Ranking for Diversified Top-n Recommendation. In *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13)*. ACM, New York, NY, USA, 415–418. https://doi.org/10.1145/2507157.2507207

[12] Saúl Vargas. 2015. *Novelty and Diversity Evaluation and Enhancement in Recommender Systems.* Ph.D. Dissertation. Universidad Autónoma de Madrid.

[13] Jacek Wasilewski and Neil Hurley. 2016. Incorporating Diversity in a Learning to Rank Recommender System. In *Proceedings of the Twenty-Ninth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2016, Key Largo, Florida, May 16-18, 2016.* 572–578.

[14] Jacek Wasilewski and Neil Hurley. 2018. Intent-aware Item-based Collaborative Filtering for Personalised Diversification. In *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization (UMAP '18)*. ACM, New York, NY, USA, 81–89. https://doi.org/10.1145/3209219.3209234

[15] Jason Weston, Samy Bengio, and Nicolas Usunier. 2011. WSABIE: Scaling Up to Large Vocabulary Image Annotation. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence - Volume Volume Three (IJCAI'11)*. AAAI Press, 2764–2770. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-460