Learning to Recommend Diverse Items over Implicit Feedback on PANDOR

Sumit Sidana
Univ. Grenoble Alpes, CNRS/LIG
Grenoble, France
sumit.sidana@univ-grenoble-alpes.fr

Charlotte Laclau
Univ. Grenoble Alpes, CNRS/LIG
Grenoble, France
charlotte.laclau@
univ-grenoble-alpes.fr

Massih-Reza Amini
Univ. Grenoble Alpes, CNRS/LIG
Grenoble, France
Massih-Reza.Amini@
univ-grenoble-alpes.fr

ABSTRACT

In this paper, we present a novel and publicly available dataset for online recommendation provided by Purch[1]. The dataset records the clicks generated by users of one of Purch's high-tech website over the ads they have been shown for one month. In addition, the dataset contains contextual information about offers such as offer titles and keywords, as well as the anonymized content of the page on which offers were displayed. Then, besides a detailed description of the dataset, we evaluate the performance of six popular baselines and propose a simple yet effective strategy on how to overcome the existing challenges inherent to implicit feedback and popularity bias introduced while designing an efficient and scalable recommendation algorithm. More specifically, we propose to demonstrate the importance of introducing diversity based on an appropriate representation of items in Recommender Systems, when the available feedback is strongly biased.

CCS CONCEPTS

• **Information systems → Recommender systems**;

ACM Reference Format:
Sumit Sidana, Charlotte Laclau, and Massih-Reza Amini. 2018. Learning to Recommend Diverse Items over Implicit Feedback on PANDOR. In *Twelfth ACM Conference on Recommender Systems (RecSys '18), October 2–7, 2018, Vancouver, BC, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3240323.3240400

1 INTRODUCTION

Nowadays, there are many learning based approaches for optimizing the performance of advertising campaigns. Most of these methods are designed to be generic and adaptable for any type of advertiser on Internet, and allow to operate on different marketing axes, including commercial performance. A competitive model has precise campaign objectives defined according to quantitative criteria based on either financial (profitability), media (traffic) or commercial (conversion, registration, purchase) goals. These objectives can be achieved through a fine user targeting and sophisticated algorithms facilitating the decision on the ads which should be displayed to a given user or when the decision to stop presenting a given ad to the user should be made. This fine ad targeting is primarily based on the collection and processing of the browsing history of the users, which can be traced using web cookies.

Contribution. In this paper, we introduce Pandor (Purch dAta for oNline recommeDation and cOld-staRt), a novel collection that gathers one month of user's traffic collected from an high-tech website. The website is an online publication owned by Purch Group which provides articles, news, price comparisons, videos and reviews on computer hardware and high technology. This dataset contains one month of implicit feedback in the form of clicks, given by more than 1.5 million users over 3,000 offers that have been displayed to them while surfing on thos website. We also provide contextual information, such as offer title and keywords as well as anonymous page content and the url on which offers are displayed. Furthermore, we present experimental results over Pandor obtained by state-of-the-art approaches in two different settings with respect to the set of items used for prediction. Additionally, we demonstrate how Pandor can be of a great interest for developing novel algorithms incorporating diversity in Recommender Systems (RS, henceforth), where the feedback provided are implicit. Indeed, although, the goal of a RS is to have fewer flops on the top [7] of the recommended list, inducing more diversity in this recommended list ensures that user may prefer to interact with at least some items in contrast to the situation where we introduce just monotonous relevant items. In addition, the recent work of [1] shows that diversity can be used in order to control the popularity bias in such type of data, also known as the problem of long tail i.e. a situation where a large majority of items have only very few ratings or clicks, either because they are new or just unpopular. The purpose of sharing such data set is to encourage research in recommender systems which scale to commercial sizes and which develop approaches to handle popularity bias. There is dearth of such data sets which are available to recommender system community.

Related datasets. The feedback given by a user can be of different nature, and it has evolved over time from explicit feedback, given in the form of ratings on a numerical scale, to mostly implicit feedback inferred from user's behavior, such as clicking on items, bookmarking a page or listening to a song. Implicit feedback presents several challenging characteristics such as the scarcity of negative feedback, i.e., only positive observations, clicks, for instance, are available. In addition, a user listening to a song, browsing through a web page, or clicking on a product does not necessarily mean that he or she likes the corresponding item, and it is, therefore, impossible

[1] http://www.purch.com/

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
RecSys '18, October 2–7, 2018, Vancouver, BC, Canada
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5901-6/18/10…$15.00
https://doi.org/10.1145/3240323.3240400

to measure the degree of preference from such interactions. For all these reasons, research on implicit feedback has gained increasing attention in very recent years [4] through competitions organized by some of the major industrial actors, like Criteo [2], Outbrain[3], XING[4], Spotify [5], as well as international conferences [10]. It is in this spirit of promoting research on Recommender Systems that scale to commercial size and address common implicit feedback related issues that we propose to publicly release Pandor.

2 PANDOR: A NOVEL DATASET FOR RS

This Section presents in detail, a novel and publicly available dataset for online recommendation provided by Purch. The dataset, referred to as Pandor, records the behavior of users of a high-tech website during one month. Pandor gathers implicit feedback in the form of both impressions and clicks, given by users who have interacted with Purch online ads displayed on web articles.

2.1 Collection of the data

The dataset is designed to provide useful information in order to create and develop effective algorithms for online advertising. Information is collected when a user, browsing through Purch's websites, is shown an ad (either a standard ad, or on the basis of user's context, for example, the content of the page that the user is browsing). In this context, 1 PageView and N OfferViews (N being the number of displayed offers) are generated. Then, if the user clicks on one of the offers that is shown to him, 1 ClickId is generated.

The dataset contains implicit feedback (offer views, clicks) of the users that have interacted with Purch's ads (see Table 1 for details where we list the features we use to train our baselines). It should be noted that the dataset which we are going to make public also contains contextual information about offers such as keywords, titles, attributes and url of the page (and its anonymized text) on which offer was displayed. As some of the baselines (refer section 3) do not use contextual information, and to keep the comparison fair, we do not use them in the baselines we compare on Pandor. However, baselines and our approach can be easily adapted to make use of all the contextual information we provide with this dataset. For privacy reasons, the UserID was anonymized. For each feedback (positive and negative), the Timestamp is recorded.

Table 1: Description of train_set, test_set and Ratings files in Pandor.

File name	Format	Features
Ratings	csv	utcDate, userId, offerViewId, offerId, wasClicked
train_set	csv	UserId, OfferId, Feedback (1 or -1), Timestamp
test_set	csv	UserId, OfferId, Feedback (1 or -1), Timestamp

Finally, we also provide the train set and the test set used in the next section. All these files and additional details about the features can be found online.[6]

[2]https://www.kaggle.com/c/criteo-display-ad-challenge
[3]https://www.kaggle.com/c/outbrain-click-prediction
[4]http://2016.recsyschallenge.com
[5]http://www.recsyschallenge.com/2018/
[6]http://ama.liglab.fr/pandor/.

2.2 Features of Pandor

Some statistics are provided in Tables 2 and 3, highlighting the complexity of the proposed data, both in terms of sparsity and size. As outlined in Table 2, the datasets gather the actions of close to 2M users over 3.7K products. Among the 18M interactions observed, only 225K resulted in a positive feedback, i.e., click. Furthermore, one can observe that the maximum number of clicks done by one user is 106 while the average number of clicks is below 0.118 (see Table 3).

Table 2: Overall Dataset Statistics.

# of users	# of unique offers	# of offers shown	# of clicks
1,918,968	3,755	18,094,817	225,579

Table 3: Overall Dataset Aggregate Statistics.

Maximum # of offers shown to 1 user	2,389
Average # of Offers Shown to 1 user	9.43
Maximum # of clicks done by 1 user	106
Minimum # of clicks done by 1 user	0
Average # of clicks done by 1 user	0.118
Average # of clicks done by 1 user (if user did at least one click)	1.431

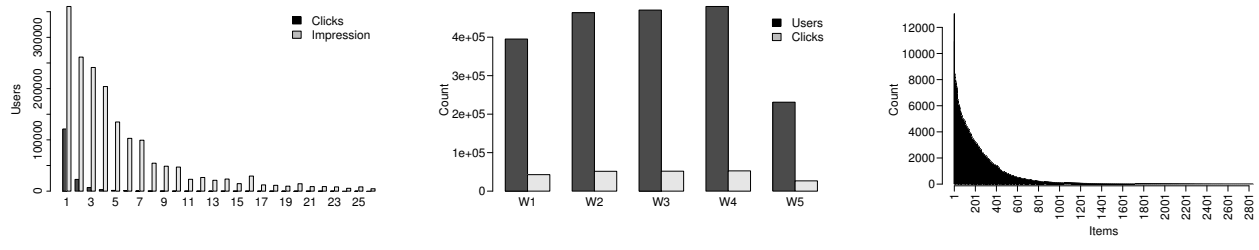
From Figure 1(a), one can observe that the number of users fall sharply as the number of clicks rises. In addition, the majority of users were shown one offer (i.e. impressions), while, the number of users that were shown 2 to 7 offers are quite balanced. Figure 1(b) depicts how the number of users and the number of clicks vary during the month the dataset was collected. We can see that both numbers remain stable over the weeks. Finally, an important specificity of the dataset is that, at the time it was extracted, the actual recommendation system in production was mainly based on the popularity of the items, meaning that the ads displayed to any particular user were mostly related to the most clicked or sold products. Another part of recommendations system is based on LDA-based user profile similarity. As a result, the coverage of items is extremely low and the dataset presents what is referred to as the long-tail phenomenon or the popularity bias in the literature [2, 6](see Figure 1(c)).

3 EXPERIMENTAL RESULTS

In this Section, we evaluate the performance of our models and demonstrate how incorporating diversity in the objective function can help to overcome the bias induced by popularity and CTR.

3.1 Set-up

Compared approaches. First, we compare the performance of different state-of-the art approaches that do not take into account the diversity for recommendation. The tested methods include two non-machine learning approaches and five machine-learning based approaches which were developed to deal with highly sparse data and implicit feedback. Popularity and Random, consists of recommending the same list of the most popular items to all users, and a list of random items to each user, respectively. We also use Rank-ALS [12], a ranking formulation of Matrix Factorization; Bayesian Personalized Ranking (BPR) [9], a pairwise ranking approach; Factorization Machine (FM) [8], a hybrid model between SVM and matrix factorization that relies on a new feature representation



(a) (b) (c)

Figure 1: (a) Number of clicks and number of offer views vs. number of users; (b) Number of clicks and number of users who did at least one click per week; (c) Long tail item : number of time each item is recommended

and has proven very efficient on sparse implicit feedback; and LightFM [5], that relies on learning the embedding of users and items with the Skip-gram model while optimizing a ranking loss. We also demonstrate the performance of `RecNet`[11], a neural network based recommendation framework with a 2-way pair-wise ranking loss for learning recommendations.

Evaluation Protocol. We filter out users without a single click; the dataset contains 1,767,589 interactions from 119,536 unique users on 2,840 unique items. In addition, we sort all interactions according to time, then take the first 70% interactions for training the models, and the remaining 30% for testing. Finally, we consider two settings w.r.t. to the set of items selected for the prediction.

(1) Item recommendation only relies on past interacted offers, that is, we only consider for a given user, the items that the user interacted with in the training phase. By interacted, we mean the user was either shown the offer or user clicked on the offer. While this is probably the most popular setting in the literature, it is also the less realistic one, as in an real online setting one has to consider all the available items when making prediction.
(2) The RS considers the full set of items as possible candidate for the prediction.

For the first setting, the average number of interacted items per user is 20.653 , i.e. the prediction is done over 20.653 items on average, while for the second one, the prediction is over 2840 items. The accuracy of the ranking list of items is evaluated by the Mean Average Precision (MAP) obtained for the set of top k= 1, 5 and 10 items. Then, following [13], we use the EILD (expected intra-list diversity) to measure diversity. High value of EILD indicates high diversity, and we report this metric at k=10. We proposed to define the distance between items as the distance between their embeddings. We give more details about this choice in Section 3.3.

3.2 Traditional Results

The results of comparing all methods on Pandor, in both settings are summarized in Tables 4 and 5. One can see that on interacted items, LightFM significantly outperforms all competing approaches and achieved reasonable performance for this task. However, looking at the results of the second setting, the compared approaches give very low performance, and BPR-MF and RecNet give slightly better performance than LightFM and FM. Figure 2 provides a deeper analysis of these results for FM and LightFM, which are supposed to be particularly efficient for this type of data. This figure shows the rank of items as a function of their click-through rate (CTR) i.e. the ratio of clicks to impressions of an item, for FM, LightFM and Popularity. We can make two observations: (1) FM's recommendation is driven by items with the highest CTR (in the top 1%); (2) LightFM behaves like Popularity, recommending only the most clicked items.

Next, we demonstrate how incorporating diversity using item embeddings, in Rank-ALS and RecNet, can enhance these results.

Table 4: MAP@k obtained for all compared approaches on interacted items on Pandor. The best results are in bold.

	MAP@1	MAP@5	MAP@10	EILD@10
Random	0.110	0.133	0.138	0.230
Popularity	0.174	0.203	0.208	0.174
FM (SGD)	0.191	0.223	0.228	**0.257**
BPR-MF	0.157	0.183	0.176	0.230
LightFM	**0.345**	**0.399**	**0.409**	0.148
RecNet	0.247	0.295	0.300	0.177
Rank-ALS	0.190	0.197	0.198	0.056

Table 5: MAP@k obtained for all compared approaches on all items on Pandor.

	MAP@1	MAP@5	MAP@10	EILD@10
Random	0.0	0.001	0.001	0.559
Popularity	0.002	0.007	0.009	0.575
FM (SGD)	0.003	0.003	0.004	0.534
BPR-MF	**0.004**	**0.008**	**0.010**	0.552
LightFM	0.001	0.002	0.005	0.299
RecNet	**0.004**	0.006	0.008	**0.615**
Rank-ALS	0.0	0.001	0.002	0.612

3.3 Diversity Results

Hereafter, we propose to explore the ability of diversity in RS to overcome the strong bias induced by popular items, or items with high CTR. Also, we focus only on the setting in which we test on all items as most approaches fail to provide good results on such setting. To this end, we propose to evaluate two approaches. The

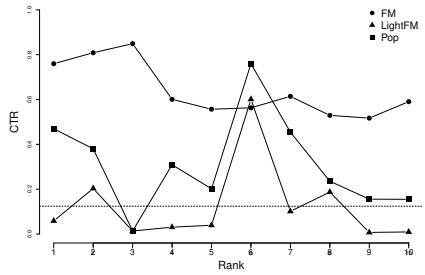


Figure 2: Rank of recommended items as a function of their CTR. Here the results are for the setting where all items are considered for making prediction. The dot line represents the average CTR of all items.

Table 6: Results of RecNet coupled with diversity. HM denotes the harmonic mean of MAP and EILD.

Metric Maximized	β	MAP@10	EILD
MAP@10	0.0001	0.014	0.604
EILD	-0.2	0.001	0.645
HM(MAP@10,EILD)	0.0001	0.014	0.604
HM(MAP@10,EILD) while maximizing diversity	-0.01	0.010	0.617

first one was initially proposed by [13] and consider the objective function of Rank-ALS [12] augmented with a regularization term that consists of the intra-list diversity (ILD) measure. Then, without loss of generality, we propose to build upon RecNet [11]. The diversity regularizers, we add here for RankALS or RecNet, can be used with any loss function. In [13], the authors used the movies' genre to compute distances between two items. However, on many occasions item metadata is not available. To overcome this problem of absence of item metadata, we propose to compute item embeddings as meta data [3]. Here, we would like to stress on the fact that computing embeddings with the Item2Vec [3] technique to measure diversity is a fresh departure from previous works on this topic; indeed, in our case, item diversity is not related to the characteristics of the items themselves, such as the genre, or the category, but rather to the diversity of the sequence of items displayed to users. This means that our goal is to, somehow, force the RS algorithm to display various diverse sequences of items to each user. We compute item embeddings, with Gensims based Skip-Gram implementation of Word2Vec (adapted to Item2Vec). We set the dimension to 20 and consider 3 as the context window.

RecNet with diversity. For RecNet, we propose to minimize the objective function of RecNet and to incorporate diversity within the list of items recommended to each user through a penalty term based on the Kullback-Leibler (KL). To this end, we propose to measure the dissimilarities between each pair of items $i \in \mathcal{S}_u$ (where $\mathcal{S}_u^k$ denotes the list of items and k its size) of the loss function associated to this new problem can be written as

$$\mathcal{L}_{RecNet}(f, U, V, \mathcal{S}) + \beta \frac{1}{|U|} \sum_{u \in U} \left(\frac{1}{k(k-1)} \right) \sum_{i, i' \in \mathcal{S}_u^k} KL(\mathbf{V}_i^{\ell_1} || \mathbf{V}_{i'}^{\ell_1}),$$

where $\mathbf{V}_i^{\ell_1}$ (resp. $\mathbf{V}_{i'}^{\ell_1}$) is the ℓ_1-normalized embedding associated with item i (resp. i'); β is the diversity inducing regularization parameter whose role is to induce more or less diversity in the final list of recommended items. Positive values of β imply minimizing diversity and vice versa. We cross-validate the value of β on a validation set built from the original training set.

Table 7: Results of RankALS coupled with diversity. HM denotes the harmonic mean of MAP and EILD

Metric Maximized	*regularizer*	MAP@10	EILD
MAP@10	PLapDQ-min	0.078	0.586
EILD	No-Regularizer	0.002	0.612
HM(MAP@10,EILD)	PLapDQ-min	0.078	0.586
HM(MAP@10,EILD) while maximizing diversity	DQ-max	0.070	0.595

Table 8: By Introducing diversity we are able to increase both relevance of the items and diversity of items

	Before Diversity		After Diversity	
	MAP@10	EILD@10	MAP@10	EILD@10
RecNet	0.008	0.615	0.010	0.617
RankALS	0.002	0.612	0.070	0.595

RankALS with diversity. In RankALS [13], a diversity regularization term is added, thus taking into account diversity in a single step learning, as we propose for RecNet. From the EILD metric, the authors derived various forms for the regularization term, all based on a distance matrix between items using some available characteristics. In this work, we compute the distance between items embeddings as described previously.

Results. Best results are summarized in Tables 6, 7 and 8. Overall, one can observe that in both cases, adding diversity based on embeddings, results in significant boost of the RS performance in terms of MAP, and allows Rank-ALS and RecNet to outperform BPR-MF (which was found to be the strongest baseline in this setting). For RecNet, one can also note that by taking negative β, we are actually able to improve MAP and EILD computed in Table 5. This observation stresses the fact that by introducing more diversity in recommendations on data sets such as Pandor, which were built by popularity biased algorithms, we are actually able to improve the relevance of recommended offers. For Rank-ALS, the gap in terms of MAP between the versions with and without diversity is even more important.

4 CONCLUSION

In this paper, we presented Pandor, a novel dataset in order to encourage future research on recommendation systems using implicit feedback. It is designed to analyze wide range of recommender approaches which make use of contextual information about offers as it contains meta-information about offers and pages on which these offers were displayed. For comprehensiveness, a description of the data and some statistics were presented. We also conducted experiments and compared strong baseline approaches, where we observed that, LightFM and BPR-MF are the best approaches when prediction is done on interacted offers and all offers respectively. Finally, we demonstrated that introducing diversity, computed based on the embedding based representation of items, can greatly improve the results and should be investigated more in this context.

ACKNOWLEDGEMENT

We thank FEDER for having financed in part of the Calypso FUI project.

REFERENCES

[1] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. 2017. Controlling Popularity Bias in Learning-to-Rank Recommendation. In *Proceedings of RecSys*. ACM, New York, NY, USA, 42–46.

[2] Chris Anderson. 2006. *The Long Tail: Why the Future of Business Is Selling Less of More*. Hyperion.

[3] Oren Barkan and Noam Koenigstein. 2016. ITEM2VEC: Neural item embedding for collaborative filtering. In *26th IEEE International Workshop on Machine Learning for Signal Processing, MLSP , Vietri sul Mare, Salerno, Italy*. IEEE, 1–6.

[4] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2016. Fast Matrix Factorization for Online Recommendation with Implicit Feedback. In *Proceedings of SIGIR*. ACM, New York, NY, USA, 549–558.

[5] Maciej Kula. 2015. Metadata Embeddings for User and Item Cold-start Recommendations. In *Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with RecSys*. CEUR-WS.org, 14–21.

[6] Yoon-Joo Park and Alexander Tuzhilin. 2008. The Long Tail of Recommender Systems and How to Leverage It. In *Proceedings of RecSys*. ACM, New York, NY, USA, 11–18.

[7] Bibek Paudel, Thilo Haas, and Abraham Bernstein. 2017. Fewer Flops at the Top: Accuracy, Diversity, and Regularization in Two-Class Collaborative Filtering. In *Proceedings of the Eleventh ACM Conference on Recommender Systems (RecSys '17)*. ACM, 1–6.

[8] Steffen Rendle. 2010. Factorization Machines. In *Proceedings of ICDM*. IEEE Computer Society, 995–1000.

[9] 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.

[10] Sumit Sidana, Charlotte Laclau, Massih-Reza Amini, Gilles Vandelle, and André Bois-Crettez. 2017. KASANDR: A Large-Scale Dataset with Implicit Feedback for Recommendation. In *Proceedings of SIGIR*. ACM, 1245–1248.

[11] Sumit Sidana, Mikhail Trofimov, Oleg Horodnitskii, Charlotte Laclau, Yury Maximov, and Massih-Reza Amini. 2017. Representation Learning and Pairwise Ranking for Implicit Feedback in Recommendation Systems. *CoRR* abs/1705.00105 (2017).

[12] Gábor Takács and Domonkos Tikk. 2012. Alternating least squares for personalized ranking. In *Proceedings of RecSys*. ACM, 83–90.

[13] Jacek Wasilewski and Neil Hurley. 2016. Incorporating Diversity in a Learning to Rank Recommender System. In *Proceedings of FLAIRS*. AAAI Press, 572–578.