Exploring Recommendations Under User-Controlled Data Filtering

Hongyi Wen
Cornell Tech, Cornell University
hw557@cornell.edu

Longqi Yang
Cornell Tech, Cornell University
ly283@cornell.edu

Michael Sobolev
Cornell Tech, Cornell University
michael.sobolev@cornell.edu

Deborah Estrin
Cornell Tech, Cornell University
destrin@cornell.edu

ABSTRACT

Traditionally, recommendation systems are built on the assumption that each service provider has full access to all user data generated on its platform. However, with increasing data privacy concerns and personal data protection regulation, service providers such as Google, Twitter, and Facebook are enabling their users to revisit, erase, and rectify their historical profiles. Future recommendation systems need to be robust to such profile modifications and user-controlled data filtering. In this paper, we explore how recommendation performance may be affected by time-sensitive user data filtering, that is, users choosing to share only recent "N days" of data. Using the MovieLens dataset as a testbed, we evaluated three widely used collaborative filtering algorithms. Our experiments demonstrate that filtering out historical user data does not significantly affect the overall recommendation performance, but its impact on individual users may vary. These findings challenge the common belief that more data is essential to better performance, and suggest a potential win–win solution for services and end users.

CCS CONCEPTS

• **Information systems → Collaborative filtering**;

KEYWORDS

Recommendation; Evaluation; Privacy; Collaborative filtering

ACM Reference Format:
Hongyi Wen, Longqi Yang, Michael Sobolev, and Deborah Estrin. 2018. Exploring Recommendations Under User-Controlled Data Filtering. 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/10.1145/3240323.3240399

1 INTRODUCTION

The classical personalization framework assumes that central services have absolute control over user-generated data. Under such a framework, recommender systems are built on the complete view of users' historical profiles. However, with increasing data privacy concerns and the emergence of personal data protection regulations, users are being granted privileges to selectively share personal data. For example, Google's MyActivity [3] enables users to review their account activities and delete specific items from the history. Similarly, some users choose to delete all of their Facebook data. Since the EU General Data Protection Regulation (GDPR) [1] took effect in May 2018, services have been required to provide greater transparency regarding data usage and grant users with controls over their own data. User-controlled data filtering will become a common practice in the future.

The paradigm shift from central to distributed data control poses challenges to the design of recommendation algorithms, as they need to be robust to ad hoc user data manipulation. In this paper, we take an initial step in exploring how the performance of different recommendation algorithms may be affected by *time-sensitive user data filtering*, that is, users choosing to share only recent "N days" of data. Such a mechanism has been widely deployed by many service providers [2].

Specifically, through an experiment using the MovieLens dataset and three widely used recommendation algorithms, we show how recommendation performance is affected by (a) the percentage of users who filter their data and (b) the time span of the shared data (i.e., the most recent N days). In addition, we investigate how the changes in performance are attributed to different groups of users (e.g., users who filter their data, users who share complete data, and new users who have no historical data). Our main findings include:

- **F1**: Population-level recommendation performance is not affected, as long as users keep at least 60 days' worth of data.
- **F2**: Incorporating complete user interaction records results in sub-optimal recommendation performance.
- **F3**: Data filtering has little impact on recommendations for cold-start users and users who share their complete history. Users who choose to filter may be negatively affected if N is small.

Our findings suggest a potential win–win solution for services and end users: It is possible to achieve competitive or even better performance while granting time-based user data control.

2 RELATED WORK

Our work is inspired by two lines of research: privacy-aware recommendation and temporal collaborative filtering.

In regard to privacy-aware recommendation, Canny et al. [6] first proposed a peer-to-peer collaborative filtering system where users

RecSys '18, October 2–7, 2018, Vancouver, BC, Canada

ACM ISBN 978-1-4503-5901-6/18/10…$15.00
https://doi.org/10.1145/3240323.3240399

have full control over their log data. Berkovsky et al.[5] investigated the trade-off between privacy and accuracy in such a decentralized setting. The authors studied how data modification techniques such as obfuscation could protect user privacy without hampering accuracy. Similar techniques were also evaluated in [14, 18]. Privacy-aware collaborative filtering has also been explored in centralized recommendation systems. Polat et al. [15] leveraged randomized perturbation to disguise user data. However, user-controlled data filtering techniques, such as removal of out-dated records based on users' time preferences, have not yet been investigated in either setting. To the best of our knowledge, our work is the first to explore this problem.

In terms of temporal collaborative filtering, previous work leveraged the time factor to improve recommender performance. For example, Sugiyama et al. [17] designed an adaptive web search engine by exploiting long-term and short-term user profiles. The authors used browser history from the previous N days to construct persistent interests and browser history for the current day to capture ephemeral aspects; Ding et.al. [7] used a time-weighting scheme to down-weight older user profiles; Baltrunas et al. [4] ensembled overlapping sub-profiles that represent users in different time contexts; Koren et al. [11] developed temporal dynamics modeling into factorization models and improved the quality of predictions. However, previous temporal recommendation models assumed access to complete user records. Our work investigates how user-driven data filtering may affect recommender performance.

3 EXPERIMENTS

The experiments were designed to understand *how different time-based user data filtering affects recommendation performance.* We considered **two control variables**: P, the percentage of users choosing to filter, and N, the time span of the data shared by each user. For example, $P = 0.25$ and $N = 30$ refers to the condition that 25% of users share their most recent 30 days of data, while the remaining 75% of users share their complete profiles.

3.1 Experimental setup

The goal of the experiments was to compare the performance of recommendation algorithms trained on different settings (P, N). To achieve that goal, we leveraged the MovieLens 20M dataset [8], along with three recommendation algorithms and two evaluation metrics. Next, we discuss these experimental details.

3.1.1 Dataset and evaluation protocol. The MovieLens dataset[1] contains movie ratings from 01/09/95 to 03/31/15. We used a testing-before-training evaluation paradigm [12]. At each time point t, where $t \in [01/01/15, 03/19/15]$, the recommendation algorithms were first trained on *partial* data, which consisted of records within the time interval $[t - N, t]$ for users who filtered, and $[01/01/14, t]$ for users who shared complete data. Then the models were tested on the ratings from the time interval $[t, t + 7]$, as shown in Figure 1. To conduct hyperparameter selection, for each user we randomly held out a positive item from the training set to for validation.

3.1.2 Algorithms and metrics. We considered implicit-feedback recommendations [10] (i.e., movies rated by a user were treated as "positive items", not-yet-rated movies were treated as "negative items") and investigated three algorithms: Probabilistic Matrix Factorization [13] (**PMF**), Bayesian Personalized Ranking [16] (**BPR**), and Collaborative Metric Learning with Uniform Weights [9] (**U-CML**). BPR and PMF were chosen because they are classical algorithms used in many recommendation systems. U-CML was selected to represent one of the state-of-the-art recommendation solutions. Each algorithm was evaluated against Hit Ratio (HR@10) and Normalized Discounted Cumulative Gain (NDCG@10).

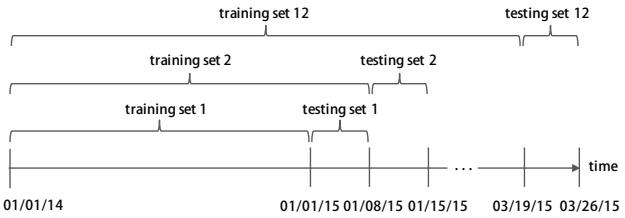

Figure 1: Recommendation algorithms were evaluated using the testing-before-training mechanism. For baseline (no data filtering), at each time point $t \in [01/01/15, 03/19/15]$, an algorithm was trained on data from the time interval $[01/01/14, t]$ and tested on data from $[t, t + 7]$. The final recommendation performance was averaged over all t.

3.1.3 Implementations. We implemented the algorithms using the OpenRec framework [19]. The code we used for the experiments is available on github [2]. Specifically, we set the user and item embedding sizes to 50 and trained all the recommenders on up to 10,000 iterations. During validation, we selected the optimal number of training iterations and L2-regularization parameter ($\alpha \in \{0.1, 0.01, 0.001, 0.0001\}$). For each algorithm and a given (P, N) setting, the validation was conducted only once, for $t = 01/01/15$, and the selected optimal model parameters were used for all t. We considered settings (P, N) for $P \in \{0.25, 0.5, 0.75, 1.0\}$ and $N \in \{1, 7, 14, 30, 60, 90, 180\}$.

3.2 Results

F1: Population-level performance. We first investigated *how user-controlled data filtering may affect recommendation performance at the population level.* Figure 2 shows the relative performance improvement under different settings (P, N) compared to the *no-filter* baseline (i.e., $P = 0$). We observe two trends: (1) A larger P has a higher impact on the population-level performance. For example, the degradation is less than 5% for HR@10 and 6.4% for NDCG@10 when $P = 0.25$. (2) The performance generally improves as N increases, and sometimes even outperforms the baseline. An interesting observation is that when $P = 1.0$, the best performance on HR@10 and NDCG@10 is achieved for N with $14 \leq N \leq 180$, suggesting that recent user profiles allow recommenders to capture up-to-date user preferences. To sum up, *population-level recommendation performance is not necessarily affected by user data filtering as long as $N \geq 60$ on this dataset.*

[1] http://files.grouplens.org/datasets/movielens/ml-20m-README.html

[2] https://github.com/whongyi/datafilter-recsys

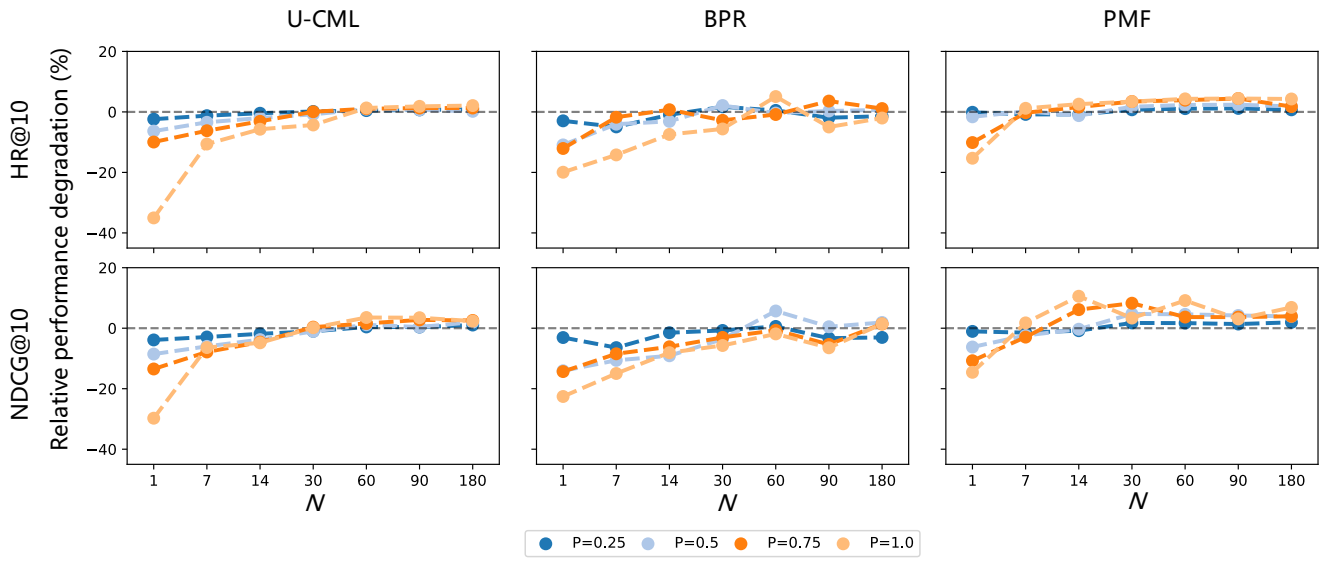

Figure 2: Relative performance degradation compared to the *no-filter* baseline under different settings (P, N). The dashed lines represent the baseline. Positive percentages indicate performance improvements, while negative percentages indicate degradation. We observe an overall improvement in performance as N increases.

Start Date	HR@10			NDCG@10		
	U-CML	BPR	PMF	U-CML	BPR	PMF
2014/01	0.6373	**0.6172**	**0.6020**	**0.3846**	**0.3697**	**0.3652**
2013/07	**0.6437**	0.6077	0.5924	0.3796	0.3582	0.3535
2013/01	0.6416	0.6121	0.5816	0.3798	0.3474	0.3176
2012/07	0.6339	0.6009	0.5382	0.3705	0.3399	0.3098
2012/01	0.6298	0.5986	0.5361	0.3688	0.3368	0.3082
2011/07	0.6179	0.5875	0.5293	0.3560	0.3342	0.3005
2011/01	0.6094	0.5813	0.5254	0.3535	0.3370	0.3017
2010/07	0.5955	0.5491	0.5118	0.3436	0.3237	0.2948
2010/01	0.5819	0.5462	0.5048	0.3303	0.3052	0.2879

Table 1: Performance of the three recommenders on user records over time intervals of different length. Recommenders trained on shorter time intervals outperformed the ones trained on longer time intervals.

F2: Effect of length of history incorporated by recommender. As stated earlier, recommenders trained on recent user data may outperform those that are trained on a one-year history. A natural question to ask is *whether including more data would improve the one-year-trained recommender baseline.* To answer this question, we trained a recommender on complete user records for time intervals of varying length: from 1 year to 5 years in multiples of 6 months. The results on performance are presented in Table 1. We observe a significant improvement in performance in terms of HR@10 and NDCG@10 when we trained the recommendation models on a one-year history, which demonstrates that the baseline used in the previous experiment is strong, and that *more data does not necessarily result in better performance.*

F3: Disaggregated performance. Lastly, we examined the disaggregated recommendation performance for three groups of users. The user group distributions are presented in Figure 3.

- G_1: Users whose records were affected after applying the filter. This group is a subset of the users who chose to filter their data.
- G_2: Users whose records were unmodified, regardless of whether they chose to filter. (The time span of a user's data may be less than N.)
- G_3: Users who had no historical record (cold-start users).

We present the decomposed performance in Figure 4. For users from G_1, the recommendation performance is negatively affected when N is small. This is because these users shared too few data points for algorithms to build accurate profiles. However, as N increases, the recommenders become more accurate and approach the performance for users who shared their complete usage history (G_2). The convergence rate varies for different algorithms. For example, PMF converges faster at $N = 60$ than U-CML and BPR at $N = 180$. These findings demonstrate that *it is possible for users to filter out their historical data without suffering from accuracy degradation, as long as more than 2 months' worth of data are shared.*

For users from G_2, the recommendation performance is not significantly affected under most settings (P, N), which suggests that users who share their complete records won't be affected by the fact that others choose to filter data.

Finally, for users from G_3, the impact of user data filtering is algorithm dependent. For example, a minor change in performance is observed for PMF and U-CML, but for BPR the recommender may be positively or negative affected for different N.

4 CONCLUSION AND FUTURE WORK

We explored how recommendation algorithm performance may be affected by time-based user data filtering. Our experimental results suggest that filtering out-of-date data records can be a win–win solution for service providers and end users to protect user privacy. Recommendation algorithms do not need complete historical

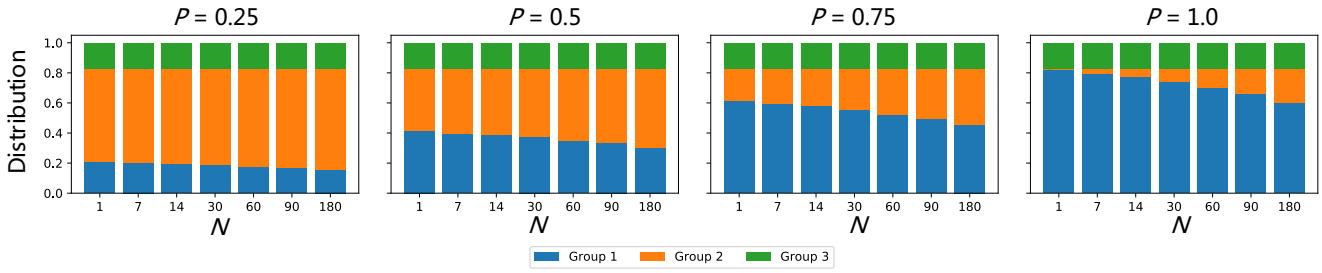

Figure 3: Distributions for the three user groups in the testing set under different settings (P, N). As N increases, the size of G_1 decreases, as some users may have records for only the most recent N days.

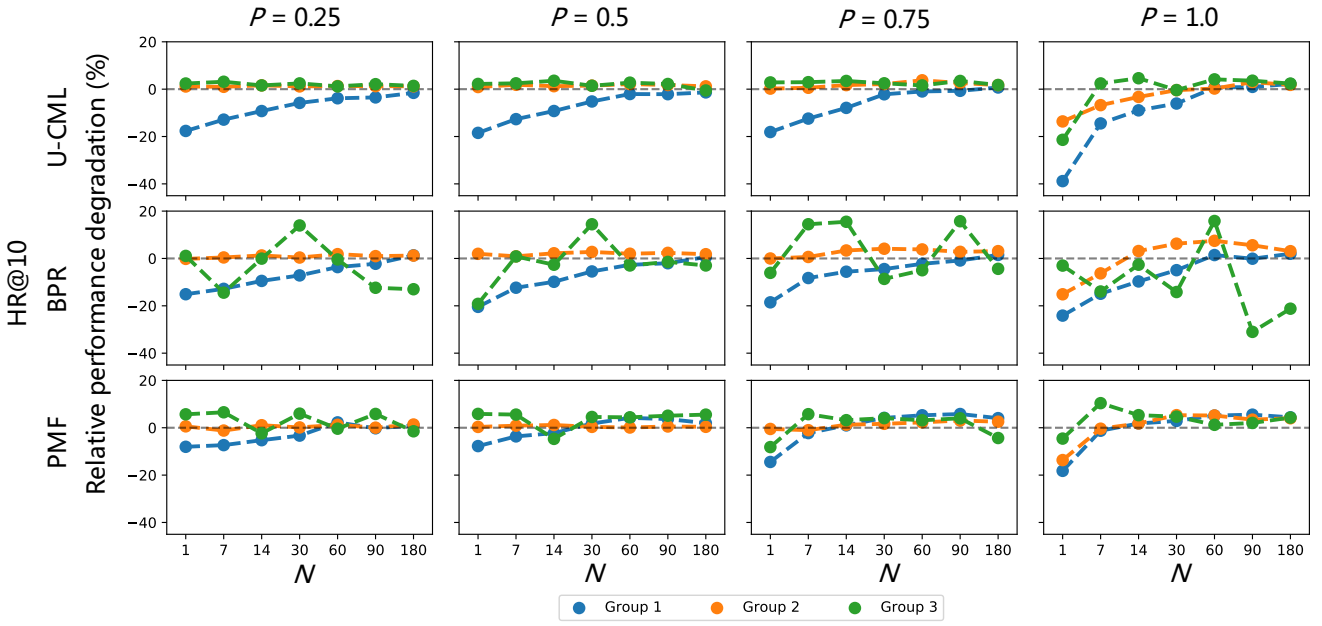

Figure 4: Relative performance degradation on the three groups of users compared to the *no-filter* baseline. G_1 and G_3 are affected more than G_2. U-CML and PMF perform more robustly than BPR, especially for cold-start users. Note that when $P = 1.0$, G_2 consists of users who had an interaction history of less than N days. (We also tested on NDCG@10 and found similar trends.)

records to perform well. In fact, incorporating outdated data is likely to degrade performance because users' preferences change over time. There are several limitations in our experiments, which we plan to address in future work.

- The recommendation algorithms we explored are preliminary. More sophisticated solutions, such as temporal models and models that incorporate auxiliary and contextual information, could be explored, raising additional research questions as to how users are going to selectively share their data beyond clicks and ratings.
- We assumed that every user who chooses to filter has the same data-sharing behavior (i.e., that they all share data from the same time period). However, in the real world, data-sharing patterns may be much more complex, for example, with different values of N across users and temporally changing privacy settings. Aside from time-based data filtering, many other data control mechanisms (e.g., sharing of data based on content, tags, and location) could be explored. These techniques have the potential to provide finer-grained user control on personal data.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their insightful comments and suggestions. This research was funded by the National Science Foundation (#1700832), Oath (the Connected Experiences Laboratory at Cornell Tech), and Google Faculty Research Awards. The work was further supported by the small data lab at Cornell Tech, which receives funding from NSF, NIH, RWJF, UnitedHealth Group, Google, and Adobe.

REFERENCES

[1] 2018. *General Data Protection Regulation.* https://www.eugdpr.org/the-regulation.html

[2] 2018. *Google Analytics.* https://support.google.com/analytics/answer/7667196

[3] 2018. *Google MyActivity.* https://myactivity.google.com/myactivity

[4] Linas Baltrunas and Xavier Amatriain. 2009. Towards time-dependant recommendation based on implicit feedback. In *Workshop on context-aware recommender systems (CARS '09)*.

[5] Shlomo Berkovsky, Yaniv Eytani, Tsvi Kuflik, and Francesco Ricci. 2007. Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In *Proceedings of the 2007 ACM conference on Recommender systems (RecSys '07)*. ACM, 9–16.

[6] John Canny. 2002. Collaborative filtering with privacy. In *Proceedings IEEE Symposium on Security and Privacy*. IEEE, 45–57.

[7] Yi Ding and Xue Li. 2005. Time weight collaborative filtering. In *Proceedings of the 14th ACM international conference on Information and knowledge management (CIKM '05)*. 485–492.

[8] 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

[9] Cheng-Kang Hsieh, Longqi Yang, Yin Cui, Tsung-Yi Lin, Serge Belongie, and Deborah Estrin. 2017. Collaborative metric learning. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*. 193–201.

[10] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *Eighth IEEE International Conference on Data Mining (ICDM'08)*. IEEE, 263–272.

[11] Yehuda Koren. 2009. Collaborative Filtering with Temporal Dynamics. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*. ACM, New York, NY, USA, 447–456. https://doi.org/10.1145/1557019.1557072

[12] Martha Larson, Alessandro Zito, Babak Loni, and Paolo Cremonesi. RecSys'17 Workshop. Towards Minimal Necessary Data: The Case for Analyzing Training Data Requirements of Recommender Algorithms. In *Workshop on Responsible Recommendation at RecSys'17*. https://doi.org/10.18122/B2VX12

[13] Andriy Mnih and Ruslan R Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in neural information processing systems (NIPS '08)*. 1257–1264.

[14] Rupa Parameswaran and Douglas M Blough. 2007. Privacy preserving collaborative filtering using data obfuscation. In *IEEE International Conference on Granular Computing. (GRC'07)*. IEEE, 380–386.

[15] Huseyin Polat and Wenliang Du. 2003. Privacy-preserving collaborative filtering using randomized perturbation techniques. In *Third IEEE International Conference on Data Mining (ICDM'03)*. IEEE, 625–628.

[16] 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, 452–461.

[17] Kazunari Sugiyama, Kenji Hatano, and Masatoshi Yoshikawa. 2004. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of the 13th international conference on World Wide Web (WWW '04)*. ACM, 675–684.

[18] Udi Weinsberg, Smriti Bhagat, Stratis Ioannidis, and Nina Taft. 2012. BlurMe: Inferring and obfuscating user gender based on ratings. In *Proceedings of the sixth ACM conference on Recommender systems (RecSys'12)*. ACM, 195–202.

[19] Longqi Yang, Eugene Bagdasaryan, Joshua Gruenstein, Cheng-Kang Hsieh, and Deborah Estrin. 2018. OpenRec: A Modular Framework for Extensible and Adaptable Recommendation Algorithms. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*. ACM, New York, NY, USA, 664–672. https://doi.org/10.1145/3159652.3159681