Using Recommendations To Balance Demand and Supply in Two-Sided Marketplaces

Deddy Jobson
Mercari Inc.
Tokyo, Japan
deddy@mercari.com

Abstract

We look into the application of recommender systems in two-sided markets. Current recommender systems are consumer-centric in the sense that their primary objective is to help consumers find relevant items. In other words, they help manage a high supply in the marketplace. We propose using recommender systems to solve the opposite problem: high demand. We show how recommendations can be made to help items find relevant consumers. Our hypothesis is that these item-centric recommendations can motivate producers to produce more, thus increasing supply and balancing out demand. We perform preliminary experiments to show the tradeoff between these item-centric and user-centric recommendations and construct a lever that one can use to decide whether to help demand or supply depending on the business goals. We discuss limitations and outline plans for future work.

Introduction

Recommender systems are a popular way to help users keep up with the enormous quantity of content available online. They help channel the attention of the customer to a subset of available content to maximize customer satisfaction. In other words, they help meet the demand of consumers. While they do indirectly help the creators of content by improving the quality of the visibility of the content, their primary objective is still consumer-focused.

Consider the case of a CtoC marketplace where anyone can buy or sell items. To sell an item, one must first ”list” an item (put it up for sale). This creates a two-sided market where listers (the producers) list things for buyers (the consumers) to buy.

If there are too many buyers but not enough listers, many buyers will be unable to find what they seek, leading to a negative experience. In the same way, if there are too many listers and not enough buyers, many listers will not get their items sold, leading to a negative experience for them. Maintaining the balance between the number of buyers and sellers is vital to help the business grow. Depending on many extraneous factors, the number of buyers and listers varies over time. For example, during holiday periods, increased demand for buying items tends to occur. During periods of economic downturn, people may want to sell items. Consequently, the goals of the business can change depending on the balance between the number of buyers and listers.

When there are too many listers on the platform, the best strategy would be to increase the number of buyers. Recommender systems already serve this need by finding relevant items for potential buyers, thus improving the buyer experience and eventually increasing the number of buyers. On the other hand, when there are too few listers, it is important to help get listed items sold so as to improve the lister experience and encourage more listings. However, current recommender systems are only buyer-focused. We would like to enhance the lister's experience as the primary goal at times when the current number of listers is too low, and improving lister motivation is our top priority.

In this extended abstract, we show how we can use recommendations to better improve the visibility of items for listers. We do so by changing the way items are assigned to users.

Our Proposal

Item-centric Recommendations

We propose a simple reranking method that can be used to make recommendations more focused on listings. We standardize the predicted probabilities for each item by the mean and standard deviation of the probabilities for that item. We then provide top N recommendations like always to users.

The proposed recommendation strategy prioritizes items that benefit most from being recommended to a user. For example, consider a simple case where we have two users and two items with relevance scores as in Table 1. The goal is to give a single recommendation to each user. Before normalizing, the more relevant item gets recommended to both users, providing maximum benefit to the users. After standardizing, we see that the originally less relevant item gets recommended to the user since the user is estimated to receive the item more favourably than other users. Therefore, the new ranking is more item-centric because it helps promote even items with a low baseline relevance.

More generally, let there be N_j users for each item j and M_i items for each user i, with the originally estimated rele-

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Table 1: Toy Example for Top 1 Item-Centric Recommendations (consisting of sub-tables 2 and 3): As per the originally estimated relevance scores, Item 1 will be recommended to both users. After performing item-centric standardization of scores, Item 2 takes precedence over Item 1 for User 1.

Table 2: Originally predicted relevance scores

	Item 1	Item 2
User 1	**5**	3
User 2	**5**	1

Table 3: Item-centric standardized relevance scores

	Item 1	Item 2
User 1	0	**1**
User 2	**0**	-1

vance score for user i and item j being s_{ij},

$$\mu_j^{item} = \frac{\sum_i s_{ij}}{N_j}$$

$$\sigma_j^{item} = \sqrt{\frac{\sum_i (s_{ij} - \mu_j^{item})^2}{N_j}}$$

$$\hat{s}_{ij}^{item} = \frac{s_{ij} - \mu_j^{item}}{\sigma_j^{item} + \epsilon}$$

ϵ is a small positive constant for numerical stability.

Constructing the Lever

In order to be able to decide the extent to which we want the recommendations to be item-centric, we need first to create user-centric recommendations. With the same terminology as before,

$$\mu_i^{user} = \frac{\sum_j s_{ij}}{M_i}$$

$$\sigma_i^{user} = \sqrt{\frac{\sum_j (s_{ij} - \mu_i^{user})^2}{M_i}}$$

$$\hat{s}_{ij}^{user} = \frac{s_{ij} - \mu_i^{user}}{\sigma_i^{user} + \epsilon}$$

Note that the rankings from $\hat{s}_{ij}^{user}$ do not differ from that of s_{ij}; we simply normalize. We next use a parameter λ, bounded between 0 and 1, to arrive at an intermediate point between the user-centric and item-centric relevance scores.

$$\hat{s}_{ij}(\lambda) = (1 - \lambda)\hat{s}_{ij}^{user} + \lambda \hat{s}_{ij}^{item}$$

Based on the priority of the business, we can set the value of λ to get recommendations that favour either the producers (high value of λ) or the consumers (low value of λ).

Experiments

We use the Movielens dataset (1 Million)(Harper and Konstan 2015) for our experiments. To avoid the cold-start problem, we iteratively remove users and items from the dataset until we have no users and items with less than 10 ratings and 10 raters respectively. We end up with 6040 users, 3233 items, and 998,269 user-item pairs with ratings in total. The ratings range from 1 to 5. We split the data into train and test datasets randomly with a ratio of 3:1.

We use the SVD algorithm from scikit-surprise(Hug 2020), an open-source Python package for our experiments, even though any recommendation algorithm that predicts relevance scores for user-item pairs can be used in its place.

We use the Negative Discounted Cumulative Gain at 20 (NDCG@20)(Järvelin and Kekäläinen 2002) score to assess the relevance of the top 20 recommendations. We measure both user-centric NDCG@20, the most commonly used calculation method, and the item-centric NDCG@20, which averages the NDCG for each item based on the ranking of the relevance of the users it gets recommended to. When estimating the NDCG, we consider two approaches. We first evaluate the NDCG for all items in the test set irrespective of whether or not the user rated the item, setting the ratings of unrated items to 0, as is common practice(Hu, Koren, and Volinsky 2008; He et al. 2017). However, this approach suffers from popularity bias(Pradel, Usunier, and Gallinari 2012). We therefore also measure the NDCG for just the rated items of each user to assess the quality of the rankings from the known data. Since the unrated data are missing not at random(Schnabel et al. 2016), we expect this approach too to be biased(Steck 2013), albeit in the opposite direction.

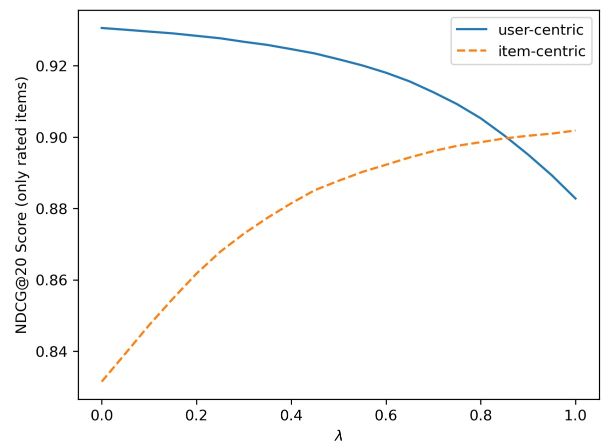

Figure 1: Variants of the NDCG score for only-rated items in the test set (this method suffers from selection bias)

As shown in Figure 1, there is a tradeoff between the user-centric and item-centric NDCG scores when evaluated on just the rated items, depending on the value of the lever chosen. The larger the value of the lever, the more item-centric the ranking, so the more producer-focused the recommendations will be. We do not see it as such in Figure 2, where we evaluate the NDCG score on all items, rated and unrated. We suspect the performance to drop the larger the value of λ because recommendations become more ”different” from the recommendations that generated the data we use for evaluation(Castells and Moffat 2022).

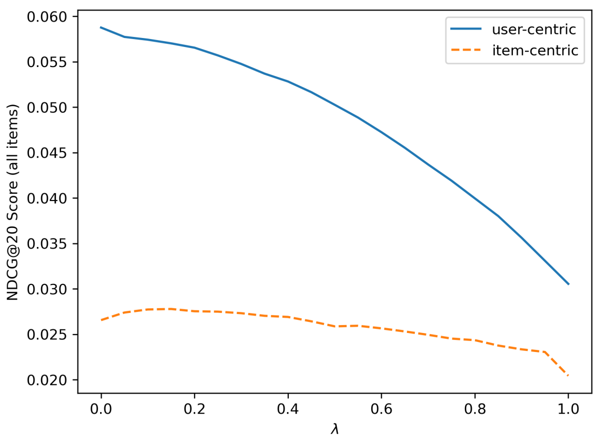

Figure 2: Variants of the NDCG score for rated and unrated items in the test set (suffers from popularity bias)

Related Work

Recommender systems have found widespread adoption in two-sided marketplaces(Su 2020; Goswami, Hedayati, and Mohapatra 2014). Most of the current implementations are consumer-focused in that they are designed to improve the well-being of the consumers and not the producers. While producers too indirectly benefit from recommendation systems on average, not all producers benefit equally; some producers may not get as much exposure and would be better off without recommendation systems. Consequently, attempts have been made to use fair recommendations(Li et al. 2022) to better support producers in the two-sided marketplace. Fair recommendations have been successfully used to correct biases that crop up in ML systems in general, like racial bias(Salman et al. 2020), gender bias(Geyik, Ambler, and Kenthapadi 2019), etc. Biswas et al. (2021) propose FairRec and FairRecPlus to guarantee a minimum amount of exposure to each producer in the two-sided marketplace.

While fair recommendation algorithms improve the welfare of the producers of two-sided markets better than regular recommendation algorithms and encourage individual producers to produce more, their motivating goal differs from this paper's since we focus on balancing the demand and supply of the marketplace as a whole. As such, our methods and evaluation criteria differ from the literature on fair recommendations. That being said, ideas from fair recommendations can be incorporated to shift the focus of recommender systems towards producers.

Limitations and Future Work

One hurdle faced by research on item-centric recommendations is that, since it invariably involves promoting less popular items, it suffers from evaluation by metrics that reward recommenders that favour items users have observed. Looking into ways to deal with missing feedback, perhaps with synthetic datasets(Wang et al. 2019; Saito et al. 2020), or using metrics that consider diversity(Parapar and Radlinski 2021) can help us better evaluate the effectiveness of our approach. Furthermore, we would also like to measure how adjusting the lever corresponds to changes in downstream business KPIs to help guide business decisions.

Ethical Implications

Recommendation systems are one of the primary means through which users engage with e-commerce marketplaces (search being another major channel). They have a strong say in the success or failure of small business that rely on the e-commerce platform, and so care must be taken to ensure they are fair to producers as well as to consumers. In this paper, however, we come across another form of fairness which is yet to be properly developed; producers vs consumers. Rather than ensuring fairness within producers or within consumers(Naghiaei, Rahmani, and Deldjoo 2022), we see that we can use the lever to instead ensure fairness between producers and consumers is maintained. Exploring the optimal setting of the lever to ensure fairness is an interesting avenue for future research.

Conclusion

In this extended abstract, we use recommender systems to help balance demand and supply in two-sided marketplaces. We do so by constructing a lever that one can use to shift the nature of the recommendations towards helping producers or consumers accordingly based on the business objective. We believe our approach can extend the capabilities of recommender systems to improve the long-term welfare of all participants of the two-sided marketplace.

Acknowledgement

I would like to thank Mercari Inc. for supporting my research and also my colleague Ueno Emi, whose insightful conversation sparked the initial concept that eventually evolved into the research presented in this extended abstract.

References

Biswas, A.; Patro, G. K.; Ganguly, N.; Gummadi, K. P.; and Chakraborty, A. 2021. Toward Fair Recommendation in Two-sided Platforms. *ACM Transactions on the Web*, 16(2): 8:1–8:34.

Castells, P.; and Moffat, A. 2022. Offline Recommender System Evaluation: Challenges and New Directions. *AI Magazine*, 43(2): 225–238. Number: 2.

Geyik, S. C.; Ambler, S.; and Kenthapadi, K. 2019. Fairness-Aware Ranking in Search & Recommendation Systems with Application to LinkedIn Talent Search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, 2221–2231. New York, NY, USA: Association for Computing Machinery. ISBN 978-1-4503-6201-6.

Goswami, A.; Hedayati, F.; and Mohapatra, P. 2014. Recommendation Systems for Markets with Two Sided Preferences. In *2014 13th International Conference on Machine Learning and Applications*, 282–287.

Harper, F. M.; and Konstan, J. A. 2015. The MovieLens Datasets: History and Context. *ACM Transactions on Interactive Intelligent Systems*, 5(4): 19:1–19:19.

He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural Collaborative Filtering. ArXiv:1708.05031 [cs].

Hu, Y.; Koren, Y.; and Volinsky, C. 2008. Collaborative Filtering for Implicit Feedback Datasets. In *2008 Eighth IEEE International Conference on Data Mining*, 263–272. ISSN: 2374-8486.

Hug, N. 2020. Surprise: A Python library for recommender systems. *Journal of Open Source Software*, 5(52): 2174.

Järvelin, K.; and Kekäläinen, J. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4): 422–446.

Li, Y.; Chen, H.; Xu, S.; Ge, Y.; Tan, J.; Liu, S.; and Zhang, Y. 2022. Fairness in Recommendation: A Survey.

Naghiaei, M.; Rahmani, H. A.; and Deldjoo, Y. 2022. CPFair: Personalized Consumer and Producer Fairness Re-ranking for Recommender Systems. ArXiv:2204.08085 [cs].

Parapar, J.; and Radlinski, F. 2021. Towards Unified Metrics for Accuracy and Diversity for Recommender Systems. In *Fifteenth ACM Conference on Recommender Systems*, 75–84. Amsterdam Netherlands: ACM. ISBN 978-1-4503-8458-2.

Pradel, B.; Usunier, N.; and Gallinari, P. 2012. Ranking with non-random missing ratings: influence of popularity and positivity on evaluation metrics. In *Proceedings of the sixth ACM conference on Recommender systems*, RecSys '12, 147–154. New York, NY, USA: Association for Computing Machinery. ISBN 978-1-4503-1270-7.

Saito, Y.; Yaginuma, S.; Nishino, Y.; Sakata, H.; and Nakata, K. 2020. Unbiased Recommender Learning from Missing-Not-At-Random Implicit Feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, WSDM '20, 501–509. New York, NY, USA: Association for Computing Machinery. ISBN 978-1-4503-6822-3.

Salman, O.; Gauch, S.; Alqahtani, M.; Ibrahim, M.; and Alsaffar, R. 2020. Incorporating Diversity in Academic Expert Recommendation.

Schnabel, T.; Swaminathan, A.; Singh, A.; Chandak, N.; and Joachims, T. 2016. Recommendations as treatments: debiasing learning and evaluation. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, 1670–1679. New York, NY, USA: JMLR.org.

Steck, H. 2013. Evaluation of recommendations: rating-prediction and ranking. In *Proceedings of the 7th ACM conference on Recommender systems*, 213–220. Hong Kong China: ACM. ISBN 978-1-4503-2409-0.

Su, Y.-H. 2020. Rankings for Two-Sided Market Platforms.

Wang, X.; Zhang, R.; Sun, Y.; and Qi, J. 2019. Doubly Robust Joint Learning for Recommendation on Data Missing Not at Random. In *Proceedings of the 36th International Conference on Machine Learning*, 6638–6647. PMLR. ISSN: 2640-3498.