RecDelta: An Interactive Dashboard on Top-k Recommendation for Cross-model Evaluation

Anonymous

Abstract

In this demonstration, we present RecDelta, an interactive tool on the top-k recommendation for cross-model evaluation. RecDelta is a web-based information system where people can compare the performance of various recommendation algorithms and their recommended items visually. In the proposed system, we visualize the distribution of δ *scores* between compared algorithms—a distance metric measuring the intersection between recommended lists. With such visualization, people can quickly identify users for whom the recommended items generated by different algorithms diverge or vice versa; then, one can further select the desired user, and the relationship between recommended items and historical behavior for this user is presented. RecDelta can benefit both academics and practitioners by enhancing the model explainability as they can work around recommendation algorithms with their newly gained insights.

Introduction

Recommender systems are ever prevalent. With the help of machine learning, platforms such as Youtube, Amazon, or Netflix can provide personalized experiences (Nie, Zhang, and Liu 2014; Smith and Linden 2017; Gomez-Uribe and Hunt 2015). For recommender systems to succeed, the algorithms used should make spot-on recommendations for each of the users. Mostly, academics leverage aggregated evaluation metrics such as precision, recall, and normalized discounted cumulative gain (NDCG) for comparing different recommendation algorithms. In addition to such an evaluation on offline datasets, practitioners usually include eyeball evaluation before entering the next phase, i.e., the A/B testing. Yet the unique nature of recommender systems, creating personalized recommendations for drastically different users, makes the algorithm evaluation process challenging. When recommendations are closely tied up to personal preferences, high aggregated evaluation scores might not be well reflected in the actual quality of recommendations (McNee, Riedl, and Konstan 2006). Moreover, when multiple algorithms share similar scores, it is always challenging to determine which algorithms to use for recommendation in practice. In this case, practitioners need to comb through the user's history and recommendations made by each algorithm and decide which model to adopt or how to combine the results from multiple algorithms. However, due to the nature of embedding-based or deep learning-based models that generally lack model explainability, analyzing the relationships among different algorithms becomes an extremely challenging problem as the user and item embeddings from different algorithms are located in different embedding space and thus hard to be compared (Himabindu, Padmanabhan, and Pujari 2018; Rendle et al. 2009; He et al. 2020).

Therefore, evaluating recommender systems beyond typical evaluation approaches has long been a critical question for both researchers and machine learning practitioners. Some prior arts propose unconventional evaluation methods from different perspectives, such as (Schröder, Thiele, and Lehner 2011; Cremonesi et al. 2011). However, most previous literature still focuses on quantitative evaluation only, and to the best of our knowledge, there exists no visualization tool aiming at comparing multiple recommendation algorithms both quantitatively and qualitatively.

We present RecDelta,[1] a prototyping system aiming to compare the performance of various recommendation algorithms and their recommended items visually. Instead of comparing different algorithms in the unexplainable embedding spaces, we turn to analyzing the recommended lists generated from different algorithms. Precisely, RecDelta visualizes the distribution of δ *scores* between compared algorithms—a distance metric measuring the intersection between recommended lists, by which people can quickly identify users for whom the recommended items generated by different algorithms diverge or vice versa. After that, one can further select the desired user, and the relationship among recommended items from different algorithms and the relationship between recommended items and historical behavior for the selected user are illustrated by Venn diagrams and heap maps, respectively. Due to our system's dynamic and interactive characteristics, RecDelta enables us to efficiently locate the similarities and differences among recommended lists and thus enhance the model explainability. Furthermore, RecDelta provides valuable insights for constructing a better recommender system, including designing new algorithms, assembling the results from multiple models, and tuning strategies for different sets of models.

[1] RecDelta is live at https://demopaper.herokuapp.com/

System Description

In this work, we build a tool featuring different model evaluation methods to simultaneously compare the recommended items generated from multiple recommendation algorithms in visual analytic workflows. To showcase our idea, we conduct the experiments and the visualization on MovieLens Latest 100K (Harper and Konstan 2015).

Comparison for recommended lists for all users

In this study, we use δ score defined in (Fagin, Kumar, and Sivakumar 2003) to measure the difference between the two recommended lists. Let ℓ_1 and ℓ_2 denote two top-k lists and $\ell_1(i)$ ($\ell_2(i)$) denotes the restriction of the top-k list ℓ_1 (ℓ_2, respectively) to the first i items. Given item sets $\mathcal{I}_{\ell_1(i)}$ and $\mathcal{I}_{\ell_2(i)}$ for $i = 1, \cdots, k$, each of which includes the items from $\ell_1(i)$ and $\ell_2(i)$, respectively, we have

$$\delta(\ell_i, \ell_j) = \frac{1}{k} \sum_{i=1}^{k} \left| \mathcal{I}_{\ell_1(i)} \Delta \mathcal{I}_{\ell_2(i)} \right| / (2i), \tag{1}$$

where the operator Δ is symmetric and $X \Delta Y := (X \setminus Y) \cup (Y \setminus X)$. For each user, δ describes the similarity between two top-k lists generated by two algorithms; the higher the score (ranging from 0 to 1) is, the lower the similarity is.

In our system, we illustrate the distribution of δ of all users (see Figure 1), where each block represents a user and a bar is composed of multiple users. Note that as the δ score is derived from a pair of lists, for the case that more than two algorithms are compared (i.e., $n > 2$ algorithms), for each user, there are C_2^n δ scores for all pairs of algorithms. Therefore, the x-axis denotes the mean of the δ scores for each user, the color on which reflects the variance of the δ scores. (Note that the blue color denotes a larger variance, while the light green one is for a smaller variance.) Such a visualization provides a macroscopic perspective when inspecting the differences among results from different recommendation algorithms. For example, one can easily observe that if the distribution plot is skewed to the left, the recommendations made by selected algorithms are generally dissimilar.

Comparison for recommended lists for a user

On the distribution chart, with the change to the "select user mode," one can select a block representing a specific user from the data set. After the selection, the system shows the Venn diagram of recommended lists generated by the algorithms for the selected user, by which people can further examine the overlaps among the recommended items and some meta information of the selected items (see Figure 2). In addition to the Venn diagrams, we use the heat maps to represent the relationships between recommended items and the user's historical behavior, for which two modes are considered in our systems. First, the user's history is sorted by the timestamp, where each block represents the cosine similarity between each top-k recommended item and one of items from user's history. Second, if the user's history is categorized by genres, each block represents the average of cosine similarities between each top-k recommended item and all historical items belonging to a specific genre.

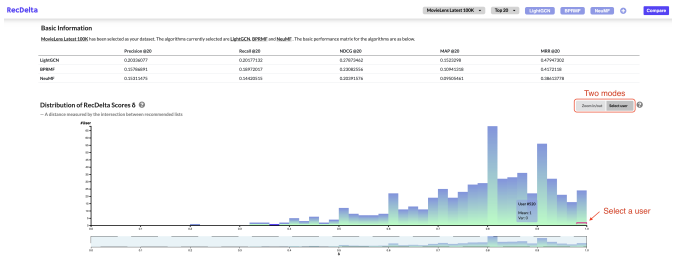

Figure 1: Distribution chart of the δ scores.

.

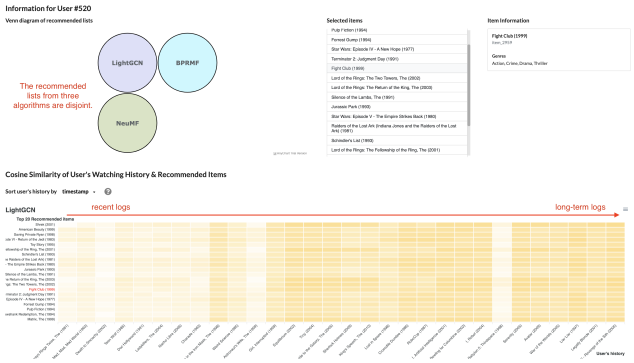

Figure 2: The Venn diagram and the heat maps.

.

Case Study

The δ scores reflect the levels of differences of top-k recommendations made by different algorithms; a high δ indicates that the top-k recommendations of the two algorithms are dissimilar. For example, when three algorithms, LightGCN, BPRMF, and NeuMF, are selected to be compared, the average δ score for user #520 is 1, the highest δ score, indicating that the recommendations made by the three algorithms are disjoint, and this observation can be supported by the non-overlapping Venn diagram shown in the upper panel of Figure 2. On the other hand, we can also tell from the heatmaps that each algorithm focuses on different aspects. For user #520, the heatmaps show that LightGCN tends to capture user's long-term behaviors (see the lower panel of Figure 2), while BPRMF weights the user's recent behavior more; as for NeuMF, it seems like there is no particular distribution in its heat map. Such visualization helps enhance model explainability and provides valuable insights for constructing a better recommender system, including designing new algorithms, assembling the results from multiple models, and tuning strategies for different sets of models. Moreover, a low δ score for user #574, 0.333, demonstrates how RecDelta can assist machine learning researchers to locate scenarios where algorithms are making rather similar recommendations. We can easily notice the algorithms share a large number of the same set of top-k items from the Venn diagram. In sum, with RecDelta, machine learning researchers and practitioners could better investigate the results generated by various algorithms at one and the same time from both macroscopic and microcosmic perspectives.

References

Cremonesi, P.; Garzotto, F.; Negro, S.; Papadopoulos, A.; and Turrin, R. 2011. Comparative Evaluation of Recommender System Quality. 1927–1932.

Fagin, R.; Kumar, R.; and Sivakumar, D. 2003. Comparing Top k Lists. *SIAM Journal on Discrete Mathematics*, 17(1): 134–160.

Gomez-Uribe, C. A.; and Hunt, N. 2015. The Netflix Recommender System: Algorithms, Business Value, and Innovation. *ACM Transactions on Management Information Systems*, 6(4): 1–19.

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

He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; and Wang, M. 2020. Lightgcn: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 639–648.

Himabindu, T. V.; Padmanabhan, V.; and Pujari, A. K. 2018. Conformal Matrix Factorization Based Recommender System. *Information Sciences*, 467: 685–707.

McNee, S. M.; Riedl, J.; and Konstan, J. A. 2006. Being Accurate Is Not Enough: How Accuracy Metrics Have Hurt Recommender Systems. In *Proceedings of the conference for Human-Computer Interactio Extended Abstracts on Human Factors in Computing Systems*, 1097–1101.

Nie, B.; Zhang, H.; and Liu, Y. 2014. Social Interaction Based Video Recommendation: Recommending YouTube Videos to Facebook Users. In *Proceedings of the 2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 97–102.

Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2009. BPR: Bayesian Personalized Ranking From implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, 452–461.

Schröder, G.; Thiele, M.; and Lehner, W. 2011. Setting Goals and Choosing Metrics for Recommender System Evaluations. In *Proceedings of the UCERSTI2 Workshop at the 5th ACM Conference on Recommender Systems*, 78–85.

Smith, B.; and Linden, G. 2017. Two Decades of Recommender Systems at Amazon.com. *IEEE Internet Computing*, 21(3): 12–18.