

# An Ensemble Hypergraph Learning framework for Recommendation

Alireza Gharahighehi<sup>1,2</sup>, Celine Vens<sup>1,2</sup>, and Konstantinos Pliakos<sup>3</sup>

<sup>1</sup> Itec, imec research group at KU Leuven, Kortrijk, Belgium

<sup>2</sup> KU Leuven, Campus KULAK, Department of Public Health and Primary Care, Kortrijk, Belgium

<sup>3</sup> KU Leuven, Department of Management, Strategy and Innovation, Leuven, Belgium

{alireza.gharahighehi,celine.vens,konstantinos.pliakos}@kuleuven.be

**Abstract.** Recommender systems are designed to predict user preferences over collections of items. These systems process users' previous interactions to decide which items should be ranked higher to satisfy their desires. An ensemble recommender system can achieve great recommendation performance by effectively combining the decisions generated by individual models. In this paper, we propose a novel ensemble recommender system that combines predictions made by different models into a unified hypergraph ranking framework. This is the first time that hypergraph ranking has been employed to model an ensemble of recommender systems. Hypergraphs are generalizations of graphs where multiple vertices can be connected via hyperedges, efficiently modeling high-order relations. We perform experiments using four datasets from the fields of movie, music and news media recommendation. The obtained results show that the ensemble hypergraph ranking method generates more accurate recommendations compared to the individual models and a weighted hybrid approach.

**Keywords:** Recommender systems · Hypergraph learning · Ensemble methods.

## 1 Introduction

Nowadays, people use digital services more and more to fulfill their needs. The owners of these services monitor users' behavior and utilize users' interactions with provided items, such as movies, songs, commercial products, to predict users' preferences. This enables the personalization of digital services and the rise of effective recommender systems (RSs) which learn from users' preferences and provide them with accurate recommendations. Generally, there are two main categories in RSs: content-based filtering and collaborative filtering approaches. Content-based RSs use the features that describe the items for computing similarities between the items and the user interaction profile. Next, they recommend items that are more similar to this user profile. Upon a recommendation query

for a target user, these RSs do not consider the interactions of the other users in generating the recommendation list. In contrast to that, collaborative filtering approaches infer the users' preferences by processing the collaborative information between users or items. In many applications, collaborative filtering RSs generate more accurate [1] and less obvious [15] recommendations compared to content-based approaches.

Each type of RS processes the information based on different assumptions to decide which items should be ranked higher among many available ones. For instance, memory-based collaborative filtering approaches (user-based and item-based) assume that users (items) with similar interactions have similar interests. Therefore, these approaches form neighborhoods to generate recommendations. Model-based collaborative filtering approaches assume that users and items can be represented in a common feature space and they use different learning methods to learn these latent features. While these approaches might vary in prediction power, they convey relevant information from different perspectives, following practically different learning strategies for the same recommendation task. Ensemble methods include multiple learning methods and integrate their predictive power into a single system, achieving superior predictive performance to individual models. Examples of ensembles in machine learning are bagging and boosting. In recommendation tasks a hybrid RS can be applied to exploit several data sources or the prediction power of different RSs to generate more relevant recommendations. An ensemble RS is a hybrid model that employs the ranking lists of multiple RSs to decide which items should be recommended to each user [2].

In this paper we propose an ensemble hypergraph learning framework for recommendation. This way we integrate the predictive power of several models into a unified RS powered by hypergraph ranking. Unlike regular graphs, where edges connect pairs of nodes, in hypergraphs multiple nodes can be connected via hyperedges. These higher order relations in hyperedges empower hypergraphs to cast more reliable information in the model [21]. Furthermore, hypergraph learning can inherently model the complex relations between different types of entities in a unified framework. It is therefore a deliberate choice for the construction of an ensemble of individual RSs driven by different types of information. Moreover, as was shown in [12], hypergraph ranking-based methods can mitigate popularity bias, enhance fairness and coverage as well as act as innate multi-stakeholder RSs. The main contribution of this paper is to construct a hypergraph as an ensemble framework for recommendation tasks. Despite its capability to stack multiple connections in a unified model, to the best of our knowledge hypergraphs have not been employed to form ensembles of RSs.

The structure of this paper is as follows: Studies about applications of hypergraph learning in RSs are presented in Section 2. Next, in Section 3, we show how a unified hypergraph can be formed as an RS (Section 3.1) and how it can formulate an ensemble of RSs (Section 3.2). In Section 4, four recommendation datasets are described and the experimental setup in designing and testing the proposed model is described. Next, the obtained results of comparing the

proposed ensemble model against other methods on these four datasets are presented and discussed in Section 5. Finally, we draw conclusions and outline some directions for future research in Section 6.

## 2 Related work

Hypergraph learning has been applied to generate recommendation lists in several applications. For instance in the music domain, Bu et al. [3] used hypergraph learning to recommend music tracks where the relations between users, tracks, albums and artists were modeled using a unified hypergraph. Hypergraph ranking has been also used in news recommendation tasks [14,12]. News usually contains very rich features such as text, tags and named entities. Therefore, hypergraph learning can effectively model the relations between these entities. Moreover, Pliakos et al. [19] used hypergraph ranking for a tag recommendation task. They built a hypergraph ranking model to capture the complex relations between different entities in the system, such as users, images, tags, and geo-tags. Hypergraph-based RSs have been also used in e-commerce applications [16,22]. For instance in [16], a multipartite hypergraph is used to model the relations between users, restaurants and attributes in a multi-objective setting. In such applications, item attributes and sequences of user-item interactions are effectively modeled in hypergraphs.

Hypergraph learning has been employed to address various issues in RSs. A hypergraph can model the relations between different types of stakeholders and objects and therefore, it can be intrinsically used as a multi-stakeholder RS [11]. Additionally, it can be used to burst the filter bubble around the user by querying a more diverse recommendation list based on the user history [14,12]. Moreover, hypergraph learning has been used to address fairness [12], the cold-start problem [24] as well as context-awareness [23] in recommendation tasks.

An ensemble RS is a type of hybrid RSs that integrates the recommendations of multiple individual RSs. Aggarwal [2] categorized hybrid RSs to monolithic, ensembles, and mixed RSs. Burke et al. [4] provided another categorization where hybrid models are categorized into weighted, switching, cascade, feature augmentation, feature combination, meta-level and mixed RSs. A weighted hybrid RS uses the weighted average of the scores from individual RSs to generate the recommendation list. For instance, Do et al. [6] applied a weighted hybrid RS based on collaborative and content-based filtering approaches on *Movielens* dataset and showed that it is more effective compared to the individual collaborative and content-based RSs. Here, we employ a unified hypergraph as an ensemble RS. Although hypergraph learning is very promising and effective in addressing many problems in RSs, to the best of our knowledge, it has never been studied as an ensemble RS.

### 3 Methodology

#### 3.1 Hypergraphs as recommender systems

Hereafter, uppercase bold letters are used for matrices, lowercase bold letters represent vectors, uppercase non-bold letters are used for sets and lowercase non-bold letters represent constants. The element in  $i^{th}$  row and  $j^{th}$  column of matrix  $\mathbf{X}$  is denoted as  $\mathbf{X}(i,j)$ .

A hypergraph consists of a set of nodes (vertices)  $N : \{n_1, n_2 \dots, n_{|N|}\}$  and a set of hyperedges  $E : \{e_1, e_2 \dots, e_{|E|}\}$  that connect the nodes. Each hyperedge can connect multiple nodes in the hypergraph. Based on the application, different types of hyperedges can be defined that capture different forms/sources of information. We define these hyperedge types in Section 3.2. In a typical collaborative filtering setting there are two types of entities in a hypergraph: users  $U : \{u_1, u_2 \dots, u_{|U|}\}$  and items  $I : \{i_1, i_2 \dots, i_{|I|}\}$ . Therefore, the set of nodes  $N$  in a hypergraph is formed based on users and items ( $N : \{U \cup I\}$ ).

Let  $\mathbf{H}$  of size  $|N| \times |E|$  be the incidence matrix of the hypergraph, where  $H(n, e) = 1$ , if node  $n$  is in hyperedge  $e$  and *zero* otherwise. Based on  $\mathbf{H}$ , the symmetric matrix  $\mathbf{A}$  can be formed using Eq.1:

$$\mathbf{A} = \mathbf{D}_n^{-1/2} \mathbf{H} \mathbf{W} \mathbf{D}_e^{-1} \mathbf{H}^T \mathbf{D}_n^{-1/2} \quad (1)$$

where  $\mathbf{D}_n$  and  $\mathbf{D}_e$  are the diagonal matrices that contain the node and hyperedge degrees and  $\mathbf{W}$  is the diagonal hyperedge weight matrix (here  $\mathbf{W} = \mathbf{I}$ ). Each element  $\mathbf{A}(i, j)$  reflects the relatedness between nodes  $i$  and  $j$ . Higher values indicate stronger relations between the corresponding nodes. Then, the recommendation problem is formulated as finding a ranking (score) vector  $\mathbf{f} \in \mathbb{R}^{|N|}$  that minimizes the following loss function [3]:

$$Q(\mathbf{f}) = \frac{1}{2} \mathbf{f}^T \mathbf{L} \mathbf{f} + \vartheta \|\mathbf{f} - \mathbf{y}\|_2^2 \quad (2)$$

where  $\mathbf{L}$  is the hypergraph Laplacian matrix (i.e.  $\mathbf{L} = \mathbf{I} - \mathbf{A}$ ),  $\vartheta$  is a regularizing parameter and  $\mathbf{y} \in \mathbb{R}^{|N|}$  is the query vector. Every item of the ranking vector  $\mathbf{f}$  or query vector  $\mathbf{y}$  corresponds to a node. Typically, to generate the recommendation list for user  $u$  in a regular recommendation task, one can query the hypergraph for user  $u$  by setting the corresponding value in the query vector to *one* ( $\mathbf{y}(u) = 1$ ) and all the other values that correspond to other nodes to *zero*. By solving the optimization problem in Eq.2, the optimal score (ranking) vector can be calculated using Eq.3:

$$\mathbf{f}^* = \frac{\vartheta}{1 + \vartheta} \left( \mathbf{I} - \frac{1}{1 + \vartheta} \mathbf{A} \right)^{-1} \mathbf{y}. \quad (3)$$

Finally, the top  $k$  items that have the highest scores in  $\mathbf{f}^*$  are recommended to the user  $u$ .

### 3.2 An ensemble hypergraph-based recommender system

An ensemble RS<sup>4</sup> utilizes the decisions of multiple individual RSs to decide which items should be ranked higher in the final recommendation lists. Let  $M : \{m_1, m_2 \dots, m_{|M|}\}$  be the set of individual methods that we want to incorporate in our ensemble RS. Each of these individual methods  $m_i$  can generate its own top  $k$  rankings  $\mathbf{R}_i \in \mathbb{R}^{|U| \times k}$  where each row in  $\mathbf{R}_i$  is the top  $k$  ranked items for the corresponding user. Then, based on the recommendation lists of each RS, hyperedges are formed to connect users to their top  $k$  recommendations.

As is mentioned previously, the hypergraph consists of multiple types of hyperedges. We consider three types of hyperedges, which are defined in Table 1. The  $E_{UI}$  hyperedges connect the users with the items that they have interacted with. To make the relations between users with similar tastes more explicit, the  $E_{UU}$  hyperedges connect users to their  $k$  nearest neighbors. To find these neighbors we use the user-item interaction matrix  $\mathbf{Z}$ , where  $\mathbf{Z}(i, j) \in \{0, 1\}$ . The  $k$  nearest neighbors of user  $u$  are users that have the highest cosine similarity with  $u^{th}$  row of matrix  $\mathbf{Z}$ . The  $E_M$  hyperedges are considered to integrate the recommendations of multiple RSs in the hypergraph. These RSs can be from different families such as collaborative filtering or content-based approaches. The fact that recommendations from any type of RS can be directly modeled as hyperedges in our system is a vital advantage of the proposed method.

We constructed the  $E_M$  hyperedge set using two well-established and powerful matrix completion-based recommendation methods, namely Bayesian Personalized Ranking (BPR) [20] and Weighted Regularized Matrix Factorization (WRMF) [13,18]. *BPR* is a learning-to-rank matrix completion approach which uses user-specific relative preferences between observed and unobserved items to learn items' and users' low rank matrices. *WRMF* is a matrix factorization approach for implicit feedback datasets that uses the alternating-least-squares optimization process to learn items and users' parameters.

**Table 1.** Hyperedge definitions

hyperedge	definition	# of hyperedges
$E_{UI}$	Each user is connected to the items that the user has interacted with	$ U $
$E_{UU}$	Each user is connected to the $k$ most similar users	$ U $
$E_M$	Each user is connected to top $k$ recommended items by a RS	$ M  \times  U $

The hypergraph and its incidence matrix  $\mathbf{H}$  are constructed using the hyperedge sets of Table 1. Following that, the affinity matrix  $\mathbf{A}$  is computed and the

<sup>4</sup> The source code is available at [https://github.com/alirezagharahi/ensemble\\_hypergraph](https://github.com/alirezagharahi/ensemble_hypergraph).

recommendation task is addressed as was described in Section 3.1. For the sake of simplicity we consider equal weights for recommendations of different models and also similar weights for items with different rankings in top  $k$  recommendation lists and leave the weight optimization as future work.

## 4 Experimental Setup

To evaluate the performance of the proposed approach we use four datasets from news, music and movie application domains. These datasets are described in Table 2. AOTM is a publicly available dataset collected from the Art-of-the-Mix platform that is based on user playlists [17]. Movielens<sup>5</sup> is a publicly available movie rating dataset [5]. As we only encode interactions in the hypergraph for this dataset we transform ratings to binary feedback. Globo<sup>6</sup> and Roularta<sup>7</sup> are news datasets that contain readers’ interactions with news articles.

**Table 2.** Datasets descriptions

	AOTM	Movielens	Globo	Roularta
<b>item type</b>	music track	movie	news article	news article
<b># users</b>	1,605	1,573	3,903	5,082
<b># items</b>	2,199	2,053	1,246	2,739
<b>sparsity</b>	3.8%	19.9%	5.7%	8.5%

In our experiments we consider the following five approaches<sup>8</sup>:

- **BPR**: Bayesian Personalized Ranking (BPR) [20] is a learning-to-rank matrix completion approach as presented in the previous section.
- **WRMF**: Weighted Regularized Matrix Factorization (WRMF) [13,18] is a MF approach using the alternating-least-squares optimization process to learn items and users’ parameters as presented in the previous section.
- **Hybrid**: A weighted hybrid model that uses scores of *BPR* and *WRMF* and then considers the weighted average of these scores to generate the final ranking lists.
- **H**: A hypergraph-based RS explained in Section 3.1 that only contains the hyperedge types of  $E_{UI}$  and  $E_{UU}$  from Table 1.
- **H<sub>Ens</sub>**: The proposed hypergraph-based ensemble RS explained in Section 3.2.

To validate the performance of the proposed method against the compared methods we randomly hide *ten* interactions of each user from training and then

<sup>5</sup> <http://www.grouplens.org>

<sup>6</sup> <http://www.globo.com>

<sup>7</sup> <http://www.roularta.be>

<sup>8</sup> For BPR and WRMF we used implicit library.

measure the ability of the methods in predicting these hidden interactions<sup>9</sup>. We use *precision@10* to measure the accuracy of predictions. *Precision* is a standard information retrieval accuracy measure that reflects the proportion of relevant items in the recommendation list. As the number of relevant items and length of the recommendation lists are the same (10 items), *precision*, *recall* and *F1-score* are all the same. Therefore, we only report *precision* in this paper. The compared methods have some hyperparameters to be tuned. *BPR* and *WRMF* have number of latent features, number of iterations, regularizing parameter and learning rate, *Hybrid* model has a hybridization weight and *H* as well as *H<sub>Ens</sub>* have a regularizer as a hyperparameter. To tune these hyperparameters we form a validation set for each dataset by randomly drawing *five* interactions of each user from the training set as the validation set. The final tuned hyperparameter values are based on *precision@10* and are reported in Table 3.

**Table 3.** Hyperparameters

		range	AOTM	MovieLens	Globo	Roularta
BPR	# iterations	[1000,2000]	1645	1984	1598	1984
	# latent features	[100,250]	129	500	168	129
	regularizing parameter	[0.01,0.05]	0.0194	0.0412	0.0374	0.0412
	learning rate	[0.001,0.07]	0.0284	0.0092	0.0174	0.0092
WRMF	# iterations	[1000,2000]	1276	1393	1129	1288
	# latent features	[100,250]	201	107	152	109
	regularizing parameter	[0.01,0.05]	0.0374	0.0225	0.0432	0.0315
Hybrid	hybridization weight	[0.01,0.99]	0.6664	0.6664	0.3986	0.2701
H	regularizing parameter	[0.01,0.99]	0.2414	0.2414	0.0656	0.0616
<i>H<sub>Ens</sub></i>	regularizing parameter	[0.01,0.99]	0.4554	0.4554	0.8301	0.6325

## 5 Results and Discussion

The results of the proposed hypergraph-based ensemble RS and the selected approaches on the four datasets are reported in Table 4. The reported values are in terms of average *precision@10* of the recommendation lists generated by the compared approaches. As is shown in Table 4, the proposed hypergraph-based ensemble RS (*H<sub>Ens</sub>*) has superior predictive performance compared to all the competitor approaches including the hybrid model in all datasets. The competitor methods have different performance rankings in the four datasets. Each of these methods processes the information based on different assumptions and learning approaches. The effectiveness of these assumptions and learning approaches differs across different applications. For instance, the pair-wise

<sup>9</sup> Users with few interactions are omitted from experiments.

learning-to-rank approach in *BPR* is more effective in *Globo* dataset compared to the point-wise error minimization approach in *WRMF*, while this does not hold in the other datasets. An ensemble RS exploits the combined predictive power of the individual methods. It considers all assumptions and decisions of various independent RSs and achieves overall superior performance regardless of the application domain of the recommendation task.

**Table 4.** Results (*precision@10*)

	<b>AOTM</b>	<b>MovieLens</b>	<b>Globo</b>	<b>Roularta</b>
BPR	0.0373	0.1716	0.0937	0.0704
WRMF	0.0402	0.1718	0.0921	0.0764
H	0.0338	0.1503	0.1125	0.0657
Hybrid	0.0388	0.1828	0.0979	0.0769
$H_{Ens}$	<b>0.0412</b>	<b>0.1860</b>	<b>0.1140</b>	<b>0.0773</b>

In this study we keep the experiments simple by only using the collaborative information, i.e. user-item interactions, to make them applicable on available datasets and various application fields (i.e. movies, music, news). Nevertheless, in cases where side information is available for users or items, content-based approaches can be included in the ensemble RS. Hypergraph learning has the natural capability of modeling the complex relations between different types of entities in a unified hypergraph and therefore is a deliberate choice to construct an ensemble of RSs with different types of information.

## 6 Conclusion

We proposed a new ensemble hypergraph learning-based RS. A unified hypergraph can integrate multiple connections between entities (here users and items) and therefore can combine the predictive power of various individual RSs boosting the precision of final recommendation lists. We empirically tested this method on four datasets from different application domains, such as news, music, and movies. The obtained results showed that the hypergraph-based ensemble RS achieves superior performance compared to all the individual models, as well as compared to a hybrid approach that averages individual scores to produce final rankings, in all datasets.

For future work we outline the following directions:

- **Weight optimization:** For the sake of simplicity we considered equal weights for individual RSs in the hypergraph-based ensemble RS and also similar weights for items in different rankings. These weights can be optimized to achieve even better performance.
- **Beyond accuracy evaluation:** In this paper we only used user-item interactions. Future approaches could include additional information and relevant stakeholders so that fairness [12] and diversity [9] are also taken into account.

- **Consumption level:** We only captured the binary feedback between users and items. In real applications usually the user feedback is graded [7] which shows to what extent the user is interested in the item. This graded feedback could be reflected in the hypergraph to model user preferences more precisely.
- **Long-term vs short-term preferences:** In some applications such as news [8] and music [10] recommendation tasks, users’ short-term preferences play important roles. Session-based RSs have been used to model such user short-term preferences. An ensemble RS could include models for both long-term and short-term preferences.

**Acknowledgments** This work was executed within the imec.icon project News-Butler, a research project bringing together academic researchers (KU Leuven, VUB) and industry partners (Roularta Media Group, Bothrs, ML6). The News-Butler project is co-financed by imec and receives project support from Flanders Innovation & Entrepreneurship (project nr. HBC.2017.0628). The authors also acknowledge support from the Flemish Government (AI Research Program).

## References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering* **17**(6), 734–749 (2005)
2. Aggarwal, C.C.: *Ensemble-Based and Hybrid Recommender Systems*, pp. 199–224. Springer International Publishing, Cham (2016). [https://doi.org/10.1007/978-3-319-29659-3\\_6](https://doi.org/10.1007/978-3-319-29659-3_6)
3. Bu, J., Tan, S., Chen, C., Wang, C., Wu, H., Zhang, L., He, X.: Music recommendation by unified hypergraph: combining social media information and music content. In: *Proceedings of the 18th ACM international conference on Multimedia*. pp. 391–400 (2010)
4. Burke, R.: Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction* **12**(4), 331–370 (2002)
5. Cantador, I., Brusilovsky, P., Kuflik, T.: 2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011). In: *Proceedings of the 5th ACM conference on Recommender systems*. RecSys 2011, ACM, New York, NY, USA (2011)
6. Do, H., Le, T., Yoon, B.: Dynamic weighted hybrid recommender systems. In: *2020 22nd International Conference on Advanced Communication Technology (ICACT)*. pp. 644–650 (2020). <https://doi.org/10.23919/ICACT48636.2020.9061465>
7. Gharahighehi, A., Vens, C.: Extended bayesian personalized ranking based on consumption behavior. In: *Artificial Intelligence and Machine Learning*, pp. 152–164. Springer (2020)
8. Gharahighehi, A., Vens, C.: Making session-based news recommenders diversity-aware. In: *Proceedings of the Workshop on Online Misinformation- and Harm-Aware Recommender Systems* (2020)
9. Gharahighehi, A., Vens, C.: Diversification in session-based news recommender systems. *Personal and Ubiquitous Computing* (2021). <https://doi.org/10.1007/s00779-021-01606-4>

10. Gharahighehi, A., Vens, C.: Personalizing diversity versus accuracy in session-based recommender systems. *SN Computer Science* **2** (2021). <https://doi.org/10.1007/s42979-020-00399-2>
11. Gharahighehi, A., Vens, C., Pliakos, K.: Multi-stakeholder news recommendation using hypergraph learning. In: *ECML PKDD 2020 Workshops*. pp. 531–535. Springer International Publishing, Cham (2020)
12. Gharahighehi, A., Vens, C., Pliakos, K.: Fair multi-stakeholder news recommender system with hypergraph ranking. *Information Processing Management* **58**(5), 102663 (2021). <https://doi.org/10.1016/j.ipm.2021.102663>, <https://www.sciencedirect.com/science/article/pii/S0306457321001515>
13. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: *2008 Eighth IEEE International Conference on Data Mining*. pp. 263–272. Ieee (2008)
14. Li, L., Li, T.: News recommendation via hypergraph learning: encapsulation of user behavior and news content. In: *Proceedings of the sixth ACM international conference on Web search and data mining*. pp. 305–314 (2013)
15. Lops, P., de Gemmis, M., Semeraro, G.: *Content-based Recommender Systems: State of the Art and Trends*, pp. 73–105. Springer US, Boston, MA (2011). [https://doi.org/10.1007/978-0-387-85820-3\\_3](https://doi.org/10.1007/978-0-387-85820-3_3)
16. Mao, M., Lu, J., Han, J., Zhang, G.: Multiobjective e-commerce recommendations based on hypergraph ranking. *Information Sciences* **471**, 269 – 287 (2019). <https://doi.org/10.1016/j.ins.2018.07.029>
17. McFee, B., Lanckriet, G.R.: Hypergraph models of playlist dialects. In: *ISMIR*. vol. 12, pp. 343–348. Citeseer (2012)
18. Pan, R., Zhou, Y., Cao, B., Liu, N.N., Lukose, R., Scholz, M., Yang, Q.: One-class collaborative filtering. In: *2008 Eighth IEEE International Conference on Data Mining*. pp. 502–511. IEEE (2008)
19. Pliakos, K., Kotropoulos, C.: Simultaneous image tagging and geo-location prediction within hypergraph ranking framework. In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. pp. 6894–6898. IEEE (2014)
20. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012)
21. Vinayak, R.K., Hassibi, B.: Crowdsourced clustering: Querying edges vs triangles. In: *Advances in Neural Information Processing Systems*. pp. 1316–1324 (2016)
22. Wang, J., Ding, K., Hong, L., Liu, H., Caverlee, J.: Next-item recommendation with sequential hypergraphs. In: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. p. 1101–1110. SIGIR '20, Association for Computing Machinery, New York, NY, USA (2020). <https://doi.org/10.1145/3397271.3401133>
23. Yu, C.A., Tai, C.L., Chan, T.S., Yang, Y.H.: Modeling multi-way relations with hypergraph embedding. In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. pp. 1707–1710 (2018)
24. Zheng, X., Luo, Y., Sun, L., Ding, X., Zhang, J.: A novel social network hybrid recommender system based on hypergraph topologic structure. *World Wide Web* **21**(4), 985–1013 (2018)