A Modular Diversity Based Reviewer Recommendation System

Marcin Maleszka[1(✉)], Bernadetta Maleszka[1], Dariusz Król[1], Marcin Hernes[2], Denis Mayr Lima Martins[3], Leschek Homann[3], and Gottfried Vossen[3]

[1] Wroclaw University of Science and Technology, 50-370 Wroclaw, Poland
{marcin.maleszka,bernadetta.maleszka,dariusz.krol}@pwr.edu.pl
[2] Wroclaw University of Economics, Wroclaw, Poland
marcin.hernes@ue.wroc.pl
[3] WWU Munster, Munster, Germany
{denis.martins,leschek.homann,vossen}@wi.uni-muenster.de

Abstract. A new approach for solving the problem of reviewer recommendation for conference or journal submissions is proposed. Instead of assigning one best reviewer and then looking for a second-best match, we want to start from a single reviewer and look for a diverse group of other possible candidates, that would complement the first one in order to cover multiple areas of the review. We present the idea of an overall modular system for determining a *grouping* of reviewers, as well as three modules for such a system: a keyword-based module, a social graph module, and a linguistic module. The added value of modular diversity is seen primarily for larger groups of reviewers. The paper also contains a proof of concept of the method.

Keywords: Recommender system · reviewer recommendation · Social recommendation · linguistic recommendation · Paper-reviewer assignment problem

1 Introduction

The assignment of scientific papers to the most suitable reviewers is an essential and crucial aspect of accepting papers to a scientific journal or conference. Each paper should be reviewed by a few experts in the scientific fields of the paper. In case of a very specialized or narrow range of topics of the conference or journal, the best experts are authors of the other submitted papers (peer review). Based on the reviewers' opinions, it is decided whether a paper is accepted or rejected. Only the high quality of the review process allows keeping the highest scientific level of the conference or journal. The (ideally automatic) recommendation of appropriate reviewers is the subject of this paper.

In the literature, one can find a few approaches to the review process organization. The most popular is a manual process where each reviewer has access

© Springer Nature Singapore Pte Ltd. 2020
P. Sitek et al. (Eds.): ACIIDS 2020, CCIS 1178, pp. 550–561, 2020.
https://doi.org/10.1007/978-981-15-3380-8_48

to the list of all papers, and he or she has to select a subset of papers to review. It is also possible to mark papers with a conflict of interest [1].

More advanced systems gather information about reviewers' interests and compare them with keywords of papers, using statistical approaches, e.g., based on similarity, to select the most suitable reviewer [2] or more sophisticated methods from artificial intelligence [3], [7], [18], and [21].

Many systems try to balance the number of papers to review, and the topic of the paper is frequently overlooked. In this paper, we focus mostly on the paper point of view — the aim is to select the most suitable group of reviewers in terms of topic compatibility and heterogeneity of reviewer approaches. The Recommender System we propose in this paper is based on using multiple diversity measures that are then averaged into a single one, in order to create a list of recommended reviewers that will cover different aspects of the reviewing process. This allows us to determine a group of reviewers for a paper in a step-by-step process. We consider interest diversity (based on reviewer's keywords), social diversity (based on co-authorship graph) and style diversity (based on differences in text style of reviews).

We want to verify the proposed method in an experiment using real conference data with anonymized papers. We will use review texts from a 2018 conference to train the style module and a co-authorship graph from 2018 for the social module. We will then test the recommended reviewers against data from a 2019 conference from the same series. Due to privacy reasons, this evaluation will be done as part of our future work. In this paper we instead present a short description of the process on simulated data.

The remainder of this paper is organized as follows: in Sect. 2 we discuss related research in the area of reviewer recommendation; in Sect. 3 we provide details on the system as a whole and the inner workings of each module; in Sect. 4 we present details on the planned experimental evaluation of the system, as well as include a proof of concept of the method and we conclude the paper in Sect. 5 with some final remarks on future works and applicability.

2 Related Work

There are different approaches to reviewer recommendation. The work [5] presents the Word Mover's Distance – Constructive Covering Algorithm. The authors transform the reviewer recommendation problem into a classification problem by integration the research field information of a submission. Based on information about reviewer candidates and a newly submitted paper, the research field labels for reviewers are predicted and assigned one to the submission characterized by the same label issue.

[6] proposes the recommendation of reviewers based on a *conflict of interest* between authors of submitted papers and reviewers. On the basis of institution-to-institution relationships (extracted from academic activity records), the distance between authors and reviewers is measured. The measure of distance is based on the similarity between the content of submitted papers and publications

of reviewers. The assignment of a paper to a reviewer is based on maximizing the topic relevance and minimizing the *conflict of interests*.

Paper [7] presents a two-layers method for reviewer recommendation. In the first layer, a cluster-based model for representing the research interests of experts is developed. In the second layer, a similarity computation method between the paper and the expert's research interests is applied. Based on the similarity, reviewers get recommended.

There are many papers related to reviewers recommendation of software code recommendation. A reviewer recommendation approach based on Latent Dirichlet Allocation (LDA) is proposed in [8]. First, the review expertise for topics of the source document from the review history is extracted. Then review scores are calculated on the basis of the topic of the source document and the review expertise.

[9] presents a recommendation model to recommend reviewers for GitHub projects. The developers to review the target projects based on a hybrid recommendation method are proposed in the first layer of the model. The second layer aims to specify whether the target developer will participate in the reviewing process.

Paper [10] proposes a code reviewer recommendation technique. It takes into consideration the relevant cross-project work history and, in addition, the experience of a developer in certain specialized technologies.

A two-layer reviewer recommendation model to recommend reviewers for Pull-Requests in GitHub projects is developed in [9]. The hybrid recommendation method for developers to review the project is developed in the first layer. On the basis of the recommendation results from the first layer, the specification of whether the target developer will technically or managerially participate in the reviewing process is performed in the second layer.

Taking graph-based recommendation into consideration, the work [11] presents a generic graph-based embedding model. Embedding learning techniques are used to this end. The model captures the sequential effect, geographical influence, temporal cyclic effect, and semantic effect in a unified way by embedding the four corresponding relational graphs into a shared low dimensional space. The recommendation is based on the time-decay method for performing dynamical computation the user's latest preferences.

Paper [12] presents the meta-graph to HIN-based recommendation. A matrix factorization and factorization machine are used in this purpose. A general graph-based model to recommendation problems related to event-based social networks is presented in [13]. The recommendation problem is transformed into a query-dependent node proximity problem. A learning scheme to set the influence weights between different types of entities has been developed.

Paper [14] presents a collaborative filtering method for the recommendation problem resolution. A novel graph clustering algorithm has been developed to obtain the appropriate clusters of users on top of which the recommendation process is leveraged.

Additionally, the problem of social recommendation is considered in the literature. For example, paper [15] analyzes a CROKODIL's folksonomy, mainly its hierarchical activity structure. Scoring approaches are proposed for recommending learning resources. Additional semantic information gained from activity structures is used.

In [16], tags and friendship links are analyzed to determine the accuracy of a graph-based recommender. The impact the features extracted from this data on the recommendations is measured.

The authors of [17] develop three taxonomies that partition recommender systems according to the properties of social media data. The following categorization criteria are taken into consideration: the objective of the recommendation (locations, users, activities, or social media), the methodologies (content-based, link analysis-based, and collaborative filtering-based methodologies), the data sources used (user profiles, user online histories, and user location histories).

Paper [18] presents several deep learning models for a recommendation based on users' needs. Large scale graph partitioning is used for improving the accuracy of models. An approach for automatically recommending a sub-type of evolution, evolutionary growth is presented in [19]. The groups are extracted from the social graph.

To sum up, there are many solutions related to the recommendation of software code reviewers. However, they cannot be indirectly used for recommending reviewers of scientific papers, because, in the case of scientific articles, the thematic scope is much broader than in the case of source code. There is also no system based on a reviewer's keywords, co-authorship graph, and differences in text style of reviews at the same time.

3 Recommender System Overview

The aim of our system is to recommend a list of the most suitable reviewers for a given paper. The reviewers should be experts in the field of the paper, but simultaneously they should have a broader area of interests and take different aspects of paper content into account to obtain less overlap in their reviews of the same papers.

The general idea of the proposed recommendation system may be described as follows: We start with a single reviewer determined by some other method (in this paper we choose one reviewer randomly, based on fitting keywords). Each module of the system is provided with this reviewer and the details of the paper as input. It will produce an output consisting of a list of potential other reviewers, ordered based on some diversity measure from best to worst fitting. Currently, we consider three such modules: interest (reviewer keywords, besides those fitting the paper, are most different), social (the reviewers are most distant in a co-authorship graph) and style (the reviewers focus on different elements in their reviews). The system averages those ordered lists into a single one, which presents potential reviewers ordered from most different to most similar from the current one. The most diverse reviewer is then selected and added to the pool of accepted reviewers. If we need more than those two reviewers, we repeat the procedure, providing the modules not with a single reviewer as input, but with the list of already accepted ones. We repeat this until the number of accepted reviewers is as desired.

One may note that this approach works better when selecting larger numbers of reviewers (scientific papers usually need only two to four), thus it may also be used to determine reviewers for a special session during a conference (e.g., out of 200 total reviewers for the conference, select 10 for the special session).

In the following sections, we present the details of all three modules: The Interest, Social, and Style Diversity Modules.

3.1 The Interest Diversity Module

The recommendation provided by the Interest Diversity Module is based on the assumption that reviewers with different research interests will focus on different aspects during the review process. In this sense, this module searches for reviewers whose research interests have a large intersection with the seed paper (i.e., paper to be reviewed), while maintaining the diversity of such interests.

To achieve this, we create a database containing reviewer candidates and their research interests in the form of a vector of keywords. Each reviewer is described by a tuple $< s, k_1, k_2, \ldots, k_{n_s} >$, where s is the name of the reviewer and $k_1, k_2, \ldots, k_{n_s}$ are the keywords representing his or her interests, and n_s denotes the number of keywords for each reviewer candidate s. Additionally, we precompute the Jaccard index between all vectors. In this approach, even if two reviewers have identical interests, the Jaccard index between them will not be 1, due to the assumption that some interests are not stated in the form of keywords.

Upon receiving the input in the form of one or more reviewer names and the keyword(s) for the reviewed paper, a temporary list of reviewers is created by selecting from the database those reviewer candidates for which at least one keyword intersects with the paper. Then the list is ordered by increasing the distance with the input reviewer(s). In case of multiple input reviewers, the distances are added before sorting the list. Thus the output of the module is a list of pairs $< s, d >$ where s is the reviewer name and d is the distance.

3.2 The Social Diversity Module

The Social Diversity module aims to find a list of reviewers that are weakly connected in a co-authorship graph. The assumption behind this module is that reviewer candidates that do not collaborate with each other in the past are likely to focus on diverse aspects of the paper review process, and, therefore, may contribute to producing a richer review.

In this context, recommending reviewers to a seed paper involves the following task: Given a co-authorship graph G and a reviewer r (e.g., one discovered by one of the methods mentioned previously in this work) as input, find a list of recommended reviewers R such that the distance between r and each reviewer candidate $r' \in R$ w.r.t. G is maximized. Hence, the module outputs a list of reviewers ordered by the most distant to the nearest ones in the co-authorship graph G.

In order to create a co-authorship graph G the Social Diversity module uses two data sources. The primary data source, as described in Sect. 4, is the database of the Digital Bibliography and Library Project (DBLP), which contains information about scientific articles and their authors, and is updated regularly. As a second data source, a list L of potential reviewers for the considered conference is used. Such a list is frequently available since the conference organizers are likely to define a potential review board composed by scholars that are familiar with the conference topic.

With these data sources, the co-authorship graph is constructed by first filtering the DBLP dataset[1] to consider publications where at least one reviewer in the reviewer list L participated in. Then, for every selected publication P, a set of nodes $a_1, a_2, \ldots, a_n$ is added in the graph G, where each a_i corresponds to one of the authors of P. Likewise, an edge (a_i, a_j) is created between authors of the same publication, that is, an edge (a_i, a_j) in G represents a co-authorship relation between authors a_i and a_j. As a result, a co-authorship graph has been created, which can be used to determine the diversity between two authors by determining their distance in the graph.

To illustrate the construction of the graph, consider the following example: Let a_1, a_2 and a_3 be authors of the paper p_1, but only a_1 and a_2 are the potential reviewers, that is, both a_1 and a_2 are in the provided list L. Furthermore, let a_3 and a_4 be authors of the paper p_2 with only a_4 being a potential reviewer. By adding a node for a_3, although not part of the potential reviewer, a connection between a_1 and a_4, respectively, and another one between a_2 and a_4 over a_3 is visible. In this case, the distance between a_1 and a_4 is one. The resulting graph of the example is depicted in Fig. 1.

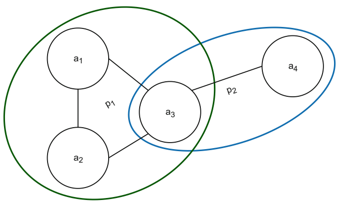

Fig. 1. The authors a_1, a_2 and a_4 are potential reviewers. a_1, a_2 and a_3 are authors of the paper p_1 indicated by the green ellipse and the authors a_3 and a_4 of the paper p_2 as indicated by the blue ellipse. The author a_3 is not part of the potential reviewer but creates a connection among the other authors.

[1] https://dblp.uni-trier.de/xml.

3.3 The Style Diversity Module

The Style Diversity Module is based on distinguishing reviewers by the aspects of the review they focus on. Some reviewers look more into language problems than others; some focus on the experimental part of papers, and some on literature review. Additionally, the language used may differ, for some reviewers positive terminology in the text may be common while the numerical grading of the review is low, while other reviewers do not write any praise but grade papers highly.

For considering the reviewer's style in the recommendation process, this module employs a deep learning method based on the research conducted by [20]. We call this approach *deep recommendation*.

The method is based on processing text reviews and ratings given to papers from previous conference editions, which may later be used to distinguish reviewers by the so-called user-word composition vector model (UWCVM). Each word w_i from a review is represented by a continuous vector $e_i \in R^d$ and each user (i.e., a reviewer) u_k by a matrix $U_k \in R^{d \times d}$, where d is the dimension of continuous vector. Due to large size of such matrix, it is reduced by approximation $U_k = U_{k1} \times U_{k2} + diag(u')$ where $U_{k1} \in R^{d \times r}$, $U_{k2} \in R^{r \times d}$ and $u' \in R^d$ is the shared background (e.g., keyword, reviewed paper, etc.). Each word is then transformed to a modified vector $p_i = tanh(e_{ik}) = tank(U_k \times e_i) = tanh((U_{k1} \times U_{k2} + diag(u')) \times e_i)$. Thus, the same words have different values for different reviewers, which appropriately captures the style of a reviewer.

In next step, a predictor is trained for ratings given, as $softmax_i = \frac{exp(z_i)}{\sum_{i'} exp(z_{i'})}$, where $z \in R^C$ is a linear vector based on the review representation $vec(doc)$: $z = W \times vec(doc) + b$, $W \in R^{C \times d}$ and $b \in R^C$ are parameters and C is the number of possible ratings. Using $f(r, l)$ as the probability of predicting review r with rating l we can train the predictor with $L(r) = \sum_{l \in C} f^g(r, l) \cdot log(f(r, l)) + \lambda_\theta \cdot |\theta|_F^2$ where f^g is a normal distribution of ratings (in original word, gold distribution; but for papers we assume that the normal distribution is proper), $|\theta|_F^2$ is a Frobenius norm regularization term and θ represents the parameters.

Finally, the distance between reviewers may be measured as a sum or average of distances between their reviews, which is calculated by using cosine similarity between vectors. This allows not only distinguishing the reviewers by their style but maintains the *normal* distribution of the reviews (with only a small number of reviewers assigned to a single paper this distribution will not be visible, but maybe crucial in case of selecting a pool of reviewers for a special session, out of a larger number of reviewers). As with previous modules, we use this distance to create a list of most diverse reviewers.

3.4 Result Integration

The output of all the diversity modules in the system is a list containing reviewer candidates and their distance from previously selected reviewers. More modules may be added as necessary, or the system may be reduced to a single diversity module, depending on the application requirements and system evaluation. The case of multiple modules requires some method of integrating multiple results into a single list of recommended reviewers. Different use cases may also require different integration methods; thus we present several simple methods that we intend to evaluate further:

- Average ranking
- Weighted and non-weighted average distance
- Weighted and non-weighted median distance

4 Evaluation of the Recommendation System

The presented system is currently being finalized, with different datasets being prepared for testing. Due to privacy reasons, much of this process is internal and cannot be presented in this paper. Due to this in this section we discuss only the three datasets used and provide a proof of concept for the proposed method with generalized data.

4.1 Dataset Description

We consider three primary datasets for the purposes of experimental evaluation of the proposed system, two for learning in different modules, and one for testing.

The DBLP [4] database provides information about scientific publications and is publicly available in the form of a downloadable XML file. According to the published statistics on the official web site [4], the database contains about 4.7 million publications, 2.3 million authors, 5,700 conferences, and 1,600 journals. The structure of the XML file is described by a Document Type Definition (DTD)[2], with basic elements being articles, proceedings, books, collections, master theses, Ph.D. theses, Web sites, and persons. In particular, the proposed system focuses on exploiting article elements, which contain information like the title, the participating authors, the journal, the publisher, or the publication date.

For the Style Diversity Module and Interest Diversity Module, we obtained data on reviews from a conference taking place in 2018. The keywords for Interest Diversity are from both input by the reviewer and extraction of keywords from the paper they wanted to review (but were not necessarily assigned to). There are a total of 216 reviewers with keyword data and 330 text reviews in this dataset.

For testing of the proposed system, we intend to use data on reviews from a conference taking place in 2019 (the next edition of the conference considered previously). There are a total of 337 reviewers in this dataset, but only 172 are present in both 2018 and 2019 data. It was possible to extract 69 groups of reviewers (2 to 4 reviewers assigned to the same paper) out of those, that are common to both sets. This will be the basis for testing the proposed approach.

[2] https://dblp.uni-trier.de/xml/dblp.dtd.

The evaluation of experimental results will require input from conference chairs responsible for assigning the reviewers. While the 2019 conference dataset contains groupings of reviewers (2 to 4 reviewers assigned to the same paper), these were not necessarily done with diversity in mind. For this reason, the simple comparison of existing groupings with system output will probably be insufficient, and conference chairs' comments will be required to evaluate the results correctly. The chairs will be presented with a table with several groupings in a row: Dataset, only Social module, only Style module, only Keyword module, and one or more Integrated groupings. They will be then asked to point the best and worst grouping in their opinion, as well as present general comments on the results.

4.2 Proof of Concept

As proof of concept we have performed evaluation based on simulated data of ten potential reviewers ($A0, \ldots, A9$) that previously wrote reviews using 10 distinct words ($w_0, \ldots, w_9$) and have selected three keywords each out of the set of 10 distinct ones ($k_0, \ldots, k_9$). Instead of full DBLP database, we use a social graph presented in Fig. 2. The keyword and review words combinations are shown in Table 1. Note that all reviewers have one keyword in common (k_0), as it represents the keyword of the reviewed paper.

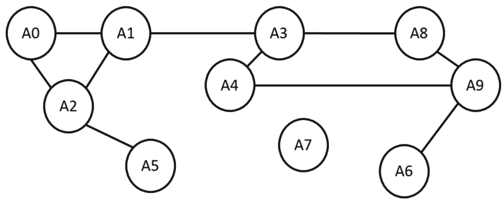

Fig. 2. The simplified social graph used for the proof of concept.

Out of the group of ten potential reviewers, the aim is to select a group of 3 most diverse ones (covering multiple aspects of the review). The proposed method requires the first reviewer to be selected by some other algorithm. Here, we use reviewer A0 for simplicity. The proposed method will need to be used twice, first to determine the second reviewer – most different from the first; then to determine the third reviewer – most different from both previously selected. Such group should be most diverse. For result integration we used average approach, as most suited to this type of data.

The approach requires first for each module to calculate the distance from each already added reviewer, using the specific method for each module. These distances are then used to rank the reviewers from most distant (most different) to closest (most similar). For the considered case, these ranks are presented in Table 2 in columns "S1: Style", "S1: Key" and "S1: Social" representing the

Table 1. Review data used for proof of concept. Three keywords and a combination of words used in reviews are given for all ten reviewers.

Reviewer	Words in reviews	Keywords
A0	w0, w1, w2	k0, k1, k2
A1	w4, w5	k0, k2, k9
A2	w4, w6, w7, w9	k0, k3, k4
A3	w2, w3, w8, w9	k0, k5, k6
A4	w0, w1, w4, w8	k0, k1, k4
A5	w1, w5, w6, w7	k0, k4, k8
A6	w0, w1, w8	k0, k5, k7
A7	w4, w5, w8	k0, k6, k7
A8	w4, w7, w9	k0, k2, k8
A9	w0, w2, w3, w4, w5, w9	k0, k3, k5

results of the style diversity module, keyword diversity module and social diversity module, respectively. Next, we calculate the average of those ranks, which allows us to determine that the most different reviewer is $A7$. Thus, after the first step the new set of reviewers is $\{A0, A7\}$.

In the next step we repeat the procedure, this time calculating the combined distance from both already selected reviewers. The results for specific modules are presented in Table 2 in columns "S1: Style", "S1: Key" and "S1: Social". With these results we calculate the average of those ranks, this time determining the most different reviewer to be $A9$. Thus the final set of reviewers selected is $\{A0, A7, A9\}$.

Table 2. Double application of the method. Determining the second reviewer based on difference from the first. Determining the first reviewer based on difference from previous two. Only ranks of distance shown, from farthest (rank 1) to closest.

Reviewer	S1: Style	S1: Key	S1: Social	S1: Total	S2: Style	S2: Key	S2: Social	S2: Total
A1	1	2	6	3	4	2	5	3,6
A2	1	1	6	2,6	1	1	5	2,3
A3	3	1	5	3	2	2	4	2,6
A4	4	2	4	3,3	4	2	3	3
A5	3	1	5	3	2	1	4	2,3
A6	5	1	2	2,6	4	2	1	2,3
A7	1	1	1	**1**	–	–	–	–
A8	1	2	4	2,3	2	2	3	2,3
A9	2	1	3	2	3	1	2	**2**

5 Conclusions

In this paper, we propose a modular recommender system for reviewer recommendation. Unlike other previous approaches, our system focus on determining a diverse group of reviewers instead of independently searching for the best fitting reviewers. In fact, other reviewer recommendation methods may be used to find the first reviewer, who will then be used as a *seed* for our proposed approach.

This paper also describes a structured plan for evaluating the proposed system, which includes a description of the real conference data sets to be employed as well as a user study to be conducted in order to obtain feedback from conference chairs. Due to privacy reasons and ongoing conference-related tasks, this data could not be used in this paper. It will also not be possible to put this data to any open repository, but we will also use it in some follow-up publications.

As noted previously, the proposed approach works better for larger groups of reviewers. Thus it may be applied even better to determining reviewers for a special session, out of a larger pool of reviewers for a conference.

Acknowledgments. This research within Polish-German cooperation program is financially supported by the German Academic Exchange Service (DAAD) under grant PPP 57391625 and by the Wrocław University of Science and Technology under project 0401/0221/18. This research was also financially supported by the Brazilian National Council for Scientific and Technological Development (CNPq) – Science without Borders Program.

References

1. Charlin, L., Zemel, R.S., Boutilier, C.: A framework for optimizing paper matching (2012). CoRR. http://arxiv.org/abs/1202.3706. Accessed 25 Sept 2019
2. Mimno, D., McCallum, A.: Expertise modeling for matching papers with reviewers. Computer Science Department Faculty Publication. Series 73 (2007)
3. Li, X., Watanabe, T.: Automatic paper-to-reviewer assignment, based on the matching degree of the reviewers. Procedia Comput. Sci. **22**, 633–642 (2013)
4. DBLP Computer Science Bibliography. https://dblp.uni-trier.de/. Accessed 25 Sept 2019
5. Zhao, S., Zhang, D., Duan, Z., Chen, J., Zhang, Y.P., Tang, J.: A novel classification method for paper-reviewer recommendation. Scientometrics **115**(3), 1293–1313 (2018)
6. Yan, S., Jin, J., Geng, Q., Zhao, Y., Huang, X.: Utilizing academic-network-based conflict of interests for paper reviewer assignment. Int. J. Knowl. Eng. **3**, 65–73 (2017)
7. Rahman, M.M., Roy, C.K., Collins, J.A.: Correct: code reviewer recommendation in GitHub based on cross-project and technology experience. In: 2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C), pp. 222–231. IEEE, May 2016
8. Kim, J., Lee, E.: Understanding review expertise of developers: a reviewer recommendation approach based on latent dirichlet allocation. Symmetry **10**(4), 114 (2018)

9. Yang, C., et al.: RevRec: a two-layer reviewer recommendation algorithm in pull-based development model. J. Cent. South Univ. **25**(5), 1129–1143 (2018). https://doi.org/10.1007/s11771-018-3812-x
10. Zhao, H., Tao, W., Zou, R., Xu, C.: Construction and application of diversified knowledge model for paper reviewers recommendation. In: Zhou, Q., Miao, Q., Wang, H., Xie, W., Wang, Y., Lu, Z. (eds.) ICPCSEE 2018. CCIS, vol. 902, pp. 120–127. Springer, Singapore (2018). https://doi.org/10.1007/978-981-13-2206-8_11
11. Xie, M., Yin, H., Wang, H., Xu, F., Chen, W., Wang, S.: Learning graph-based POI embedding for location-based recommendation. In: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, pp. 15–24. ACM, October 2016
12. Zhao, H., Yao, Q., Li, J., Song, Y., Lee, D.L.: Meta-graph based recommendation fusion over heterogeneous information networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 635–644. ACM, August 2017
13. Pham, T.A.N., Li, X., Cong, G., Zhang, Z.: A general graph-based model for recommendation in event-based social networks. In: 2015 IEEE 31st International Conference on Data Engineering, pp. 567–578. IEEE, April 2015
14. Moradi, P., Ahmadian, S., Akhlaghian, F.: An effective trust-based recommendation method using a novel graph clustering algorithm. Physica A **436**, 462–481 (2015)
15. Anjorin, M., Rodenhausen, T., Domínguez García, R., Rensing, C.: Exploiting semantic information for graph-based recommendations of learning resources. In: Ravenscroft, A., Lindstaedt, S., Kloos, C.D., Hernández-Leo, D. (eds.) EC-TEL 2012. LNCS, vol. 7563, pp. 9–22. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33263-0_2
16. Tiroshi, A., Berkovsky, S., Kaafar, M.A., Vallet, D., Kuflik, T.: Graph-based recommendations: make the most out of social data. In: Dimitrova, V., Kuflik, T., Chin, D., Ricci, F., Dolog, P., Houben, G.-J. (eds.) UMAP 2014. LNCS, vol. 8538, pp. 447–458. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08786-3_40
17. Bao, J., Zheng, Y., Wilkie, D., Mokbel, M.: Recommendations in location-based social networks: a survey. GeoInformatica **19**(3), 525–565 (2015)
18. Bathla, G., Aggarwal, H., Rani, R.: Improving Recommendation Techniques by Deep Learning and Large Scale Graph Partitioning. Int. J. Adv. Comput. Sci. Appl. **9**(10), 403–409 (2018)
19. Bartel, J.W., Dewan, P.: Towards evolutionary named group recommendations. Comput. Support. Coop. Work, 27, 3-6 (December 2018), 983-1018 2018. https://doi.org/10.1007/s10606-018-9321-5
20. Tang, D., Qin, B., Liu, T., Yang, Y.: User modeling with neural network for review rating prediction. In: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015) (2015)
21. Kolasa, T., Król, D.: A survey of algorithms for paper-reviewer assignment problem. IETE Tech. Rev. **28**(2), 123–134 (2011). https://doi.org/10.4103/0256-4602.78092