Subgraph Representation Learning for Team Mining

Radin Hamidi Rad
radin@ryerson.ca
Ryerson University
Toronto, ON, Canada

Ebrahim Bagheri
bagheri@ryerson.ca
Ryerson University
Toronto, ON, Canada

Mehdi Kargar
kargar@ryerson.ca
Ryerson University
Toronto, ON, Canada

Divesh Srivastava
divesh@research.att.com
AT&T Chief Data Office
Bedminster, NJ, USA

Jaroslaw Szlichta
jarek@ontariotechu.ca
Ontario Tech University
Oshawa, ON, Canada

ABSTRACT

Team mining is concerned with the identification of a group of experts that are able to collaborate with each other in order to collectively cover a set of required skills. This problem has mainly been addressed either through graph search, which looks for subgraphs that satisfy the skill requirements or through neural architectures that learn a mapping from the skill space to the expert space. An exact graph-based solution to this problem is intractable and its heuristic variants are only able to identify sub-optimal solutions. On the other hand, neural architecture-based solutions are prone to overfitting and simplistically reduce the problem of team formation to one of expert ranking. Our work in this paper proposes an unsupervised heterogeneous skip-gram-based subgraph mining approach that can learn representations for subgraphs in a collaboration network. Unlike previous work, the subgraph representations allow our method to mine teams that have past collaborative history and collectively cover the requested desirable skills. Through our experiments, we demonstrate that our proposed approach is able to outperform a host of state-of-the-art team mining techniques from both quantitative and qualitative perspectives.

CCS CONCEPTS

• **Information systems** → *Retrieval models and ranking*; **Expert search**; • **Computing methodologies** → *Search methodologies.*

KEYWORDS

Team Formation, Expert Networks, Task Assignment, Graph Representation Learning

ACM Reference Format:
Radin Hamidi Rad, Ebrahim Bagheri, Mehdi Kargar, Divesh Srivastava, and Jaroslaw Szlichta. 2022. Subgraph Representation Learning for Team Mining. In *14th ACM Web Science Conference 2022 (WebSci '22), June 26–29, 2022, Barcelona, Spain.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3501247.3531578

WebSci '22, June 26–29, 2022, Barcelona, Spain

ACM ISBN 978-1-4503-9191-7/22/06…$15.00
https://doi.org/10.1145/3501247.3531578

1 INTRODUCTION

In many domains, challenging problems can only be addressed through effective collaboration between different experts with complementary skill sets who have the willingness to work with each other towards a common goal. Hence, the team mining task is focused on facilitating the process of identifying efficient teams of experts that can work towards an objective that requires experts with a variety of skill sets. More formally stated, the *team mining* task is concerned with identifying a group of experts who can maximally cover a set of desirable skills and have shown to have effective past collaboration experience with each other. A widely adopted approach to addressing this problem has been to view the past collaboration history of experts in the form of a graph structure that would include information about experts, their skills and past collaborative efforts. Then, the problem of team mining is defined as one of mining subgraphs from this graph structure in such a way that the nodes of the identified subgraph collectively respect properties, such as being closely connected and covering the required skillsets. While a theoretically eloquent approach, such problems have shown to be computationally intractable because subgraph optimization techniques are NP-hard by nature [10]. For this reason, authors such as Kargar et al [8], Bryson et al. [2], Sun et al. [19] and Chen et al. [4], to name a few, have proposed heuristic techniques to reduce the graph search space in order to be able to practically address the problem. Hence, The reduction in the large search space leads to the mining of sub-optimal teams.

To address the challenges faced by the graph-based techniques, more recent approaches have focused on using neural architectures to learn mappings between the skill and expert spaces. The idea behind these approaches is that the team mining task can be seen as one of mapping a subset of the skills into a subset of the expert spaces. To this end, Sapienza et al. [15] utilize an autoencoder architecture to learn team representations, so that mapping can be done between skills and experts. Similarly, Rad et al. [14] propose to adopt a variational Bayesian neural network to learn an explicit mapping between the skill and expert spaces. In these approaches, the trained mapping function allows for the selection of the corresponding experts that might be relevant for the input skills. While such neural based methods have shown to be significantly more effective than graph-based techniques, they face two major challenges: (C1) given the past collaboration history of experts and the association of the experts with skills are very sparse, i.e., each expert often only collaborates with a restricted number of collaborators and has a limited number of skills, such neural methods

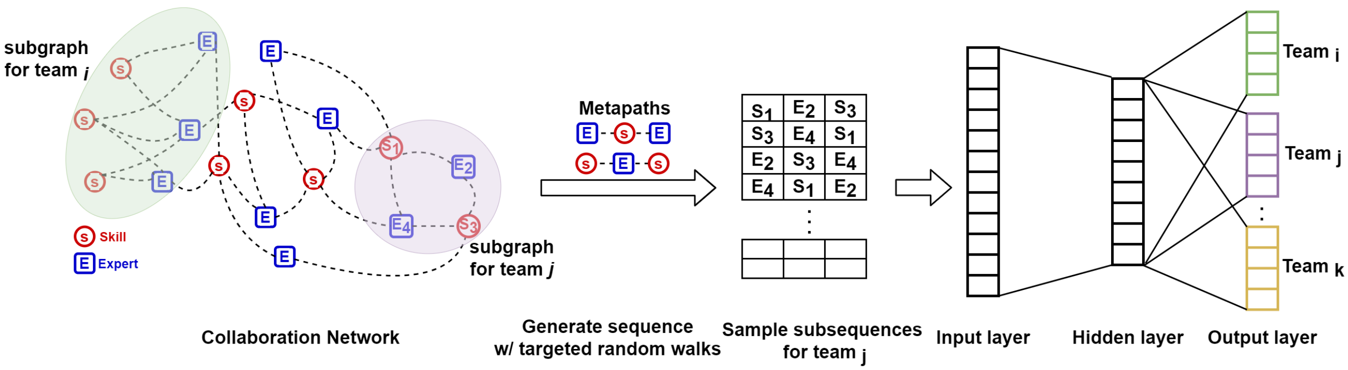

Figure 1: The overview of our proposed approach.

are prone to overfitting; and (C2) while the neural mapping methods learn associations between individual skills and experts, they do not explicitly maintain the structural team relationships that were observed during training. As such, when proposing teams, the neural mapping methods rank individual experts based on their association with the input skills. This could lead to teams that may not always have synergistic skill sets or be collaborative.

The objective of this work is to propose a lightweight neural architecture that addresses the main two challenges of existing neural methods. To this end, we propose an unsupervised approach that learns *subgraph representations* based on the past collaboration history of the experts and the association between experts and skills. The *major distinguishing* aspects of our work is that: (1) we do not learn individual representations for skills and experts, however, rather learn subgraph representations for experts who have collaborated in the past and skills that were observed in tandem in past teams. For this reason, our approach addresses challenge C2 of neural mapping approach, as the learnt subgraph representations preserve past collaboration structure and hence, the recommended teams are cognizant of effective past collaboration, as well as the relevance of the experts to the required skills; and (2) our approach does not directly learn a mapping from the skill space to the expert space in a supervised way, but rather learns subgraph representations based on a heterogeneous skipgram model. For this reason, our approach is able to overcome the collaboration network sparsity problem and avoid overfitting; hence, addressing Challenge C1 faced by neural mapping techniques. We experimentally show that our unsupervised approach for subgraph representation learning is able to show superior quantitative and qualitative performances over both state-of-the-art graph and neural mapping based approaches.

2 METHODOLOGY

2.1 Problem Definition

The objective of the *team mining* problem is to form a group of experts that collectively maximally covers a set of desirable input skills. We let $\mathcal{S}$ represent all of the available skills, and $\mathcal{E}$ denote the complete set of experts. We formally define each team as a set of experts ε which cover a set of skills s. Thus, each team can be represented as (s, ε), where $\varepsilon \subseteq \mathcal{E}; \varepsilon \neq \emptyset$ and $s \subseteq \mathcal{S}; s \neq \emptyset$. We let $\mathcal{T} = \{(s_k, \varepsilon_k)\}_{k=1}^{|\mathcal{T}|}$ be a dataset consisting of $|\mathcal{T}|$ teams. On this basis, the team mining task can be defined as:

$$TM(s_i) \longrightarrow (\varepsilon_j) \tag{1}$$

where $s_i \subseteq \mathcal{S}$, $\varepsilon_j \subseteq \mathcal{E}$ and $TM(.)$ is a function that maps across the skill space to the expert space.

2.2 Approach Overview

In order to be able to learn the function that connects the skill and expert spaces, we first model the relationship between skills and experts in the form of a heterogeneous graph, which we refer to as a *collaboration network*. We let *G(V, E, T)* denote the collaborations network, where V is the set of nodes, each of which is associated with a type defined by the mapping function $\phi(v) : V \longrightarrow T_V$ and E is a set of edges connecting the heterogeneous nodes. In this collaboration network, T_V consists of two types of nodes, namely expert type and skill type. Based on the collaboration network, each team (s_k, ε_k) represents two subgraphs of the collaboration network, where s_k is a subgraph of G ($s_k \subseteq G$), such that all nodes in s_k are of type skill, and ε_k is also a subset of G ($\varepsilon_k \subseteq G$), where all nodes in ε_k are of type expert. On this basis, we define the problem of *team mining* as one of *subgraph representation learning* where representations of subgraphs for experts on the same team are placed geometrically speaking close to the representations of the subgraphs representing the skills of that team in embedding space. Once learnt, the embedding representation of the collaboration network subgraphs can be used to determine subgraph similarities. As such, given an input set of skills, which can be represented as a subgraph consisting of skills, we identify the closest subgraph of experts to the skills subgraph in embedding space. The identified expert subgraph would be the mined team for the input skills.

2.3 Model Architecture

Our proposed approach for team mining, as depicted in Figure 1, adopts an unsupervised architecture for subgraph representation learning. In order to learn the embedding representations for a particular subgraph, we employ a heterogeneous skip-gram model to capture the heterogeneous neighborhood of each node in the subgraph. A heterogeneous skip-gram model requires a sequence of observed heterogeneous nodes to be able to learn representations. Several existing techniques apply random walk over the network to build node sequences that can be used to learn representations [3, 16]. However, Sun et al. [18] showed that using a random walk technique can cause bias towards node types that are more dominant in the graph. Therefore, we perform targeted random walk by using a meta-path scheme as suggested by Fard et al. [7] and Dong et al. [5] to capture the semantic and structural characteristics of a targeted set of node neighbors. For each subgraph, meta-path-based random walks need to be performed for each of the nodes in the subgraph. In order to perform a meta-path-based random walk, the probability of choosing a certain node for the next hop of the random walk can be formulated as follows:

$$p\left(v^{i+1} \mid v_t^i, \mathcal{P}\right) = \begin{cases} \frac{1}{\left|N_{t+1}\left(v_t^i\right)\right|} & \left(v^{i+1}, v_t^i\right) \in E, \phi\left(v^{i+1}\right) = t+1 \\ 0 & \left(v^{i+1}, v_t^i\right) \in E, \phi\left(v^{i+1}\right) \neq t+1 \\ 0 & \left(v^{i+1}, v_t^i\right) \notin E \end{cases} \tag{2}$$

where $v_t^i \in V_t$ and $N_{t+1}\left(v_t^i\right)$ specifies the type of the next node (V_{t+1}). Node types are determined given a meta-path scheme $\mathcal{P}$. Meta-path schemes are generally designed in a symmetric way, which means that for the scheme $\mathcal{P}$ with length l, the first node type V_1 and last node type V_l are the same [17, 18]. Using a symmetrical scheme enables the random walker to take recursive walks as follows:

$$p\left(v^{i+1} \mid v_t^i\right) = p\left(v^{i+1} \mid v_1^i\right) \text{ when } t = l \tag{3}$$

The targeted meta-path-based random walk process generates a set of node sequences for any given start node. Let us assume a subgraph representing team (s_k, ε_k), which consists of a set of nodes in $s_k \cup \varepsilon_k$. We additionally produce an expert subgraph and a skills subgraph from each team. Therefore, three subgraphs from each team consisting of $\{s_k, \varepsilon_k, s_k \cup \varepsilon_k\}$ are generated.

Now, given the set of all team subgraphs $\mathcal{T} = \{(s_k, \varepsilon_k)\}_{k=1}^{|\mathcal{T}|}$, and the three subgraphs derived from each team, i.e., $\{s_k, \varepsilon_k, s_k \cup \varepsilon_k\}$, we denote the collection of such three subgraphs for all teams as $\mathcal{T}'$. Furthermore, the set of all node sequences observed when traversing the heterogeneous collaboration network starting from any node in any of the three subgraphs $(\{s_k, \varepsilon_k, s_k \cup \varepsilon_k\})$ are the set of node sequences representing that subgraph. We enable the heterogeneous skipgram to learn effective subgraph representations by maximizing the heterogeneous context for nodes in each subgraph in the following form:

$$\arg\max_{\theta} \sum_{sg \in \mathcal{T}'} \sum_{v \in sg} \sum_{t \in T_V} \sum_{c_t \in N_t(v)} \log p\left(c_t \mid v; \theta\right) \tag{4}$$

where $N_t(v)$ is the meta-path-based node sequences harvested for node v with node type t. The probability function, $p\left(c_t \mid v; \theta\right)$ in Equation 4 is a softmax function [1]. In order to develop an efficient implementation, we adapt the negative sampling technique proposed by [13] through which a small set of subgraph instances are sampled for the computation of the softmax function. Assuming a set of M negative samples $U = \{u^i\}_{i=1}^{M}$, Equation 4 is modified by replacing $p\left(c_t \mid v; \theta\right)$ with $\log \sigma\left(X_{c_t} \cdot X_v\right) + \sum_{m=1}^{M} \mathbb{E}_{u^m \sim P(u)}\left[\log \sigma\left(-X_{u^m} \cdot X_v\right)\right]$, where $\sigma(x) = \frac{1}{1+e^{-x}}$ and $P(u)$ is the pre-defined distribution from which a negative sample u^m is drawn for M times and X is the set of embedding representations for subgraphs in $\mathcal{T}'$, formally defined as $X \in \mathbb{R}^{|\mathcal{T}'| \times d}, d \ll |\mathcal{T}'|$.

On this basis, we learn subgraph representations for expert, skill and team subgraphs separately, which can be effectively used for the purpose of team mining. Concretely, given a skill subgraph representing the desired set of skills in the input, we identify the most similar expert subgraph whose embedding representation has the highest similarity to that of the input skill subgraph to serve as the mined team that is the best match for the required skills.

Table 1: DBLP dataset Attributes

Attribute	Value
#Papers	33,002
#Authors	2,470
#Venues	21
#Skills	2,000
#Edge	301,369
#Nodes	37,493
#Skill/Paper	6.5
#Paper/Venue	1,571
#Avg. Node Degree In Graph	16.076

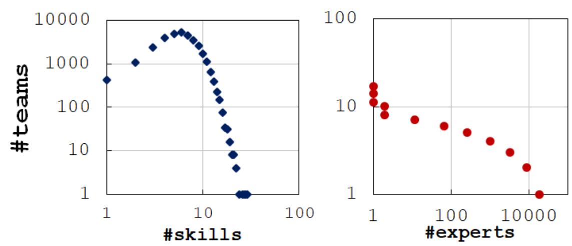

Figure 2: Distribution of articles over the skills and experts.

3 EVALUATION

3.1 Experiment Setup

Dataset. In order to benchmark our proposed approach against state-of-the-art methods, as suggested by [9, 12], we adopt the widely used DBLP dataset for our task. The DBLP dataset consists of open bibliographic information of Computer Science publications. The statistical properties of the dataset used in our experiments are shown in Table 1. In our work, we consider paper authors to be experts, the set of authors on each paper to form a team, and the top-2000 unigrams, bigrams and trigrams with the highest tf-idf values to constitute the set of skills. As shown in the table, our dataset consists of 33,002 papers (teams), 2,470 authors (experts) and 2,000 skills. The distribution of skills over the team sizes is shown in Figure 2. In order to avoid leakage in our experiments, we split the papers in the dataset based on a ten-fold cross-validation strategy. The implementation of our proposed method along with

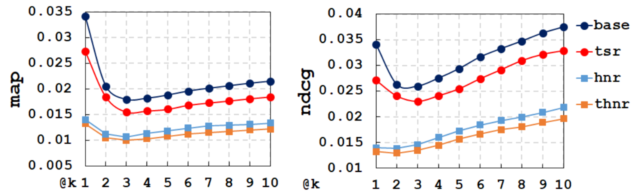

Figure 3: Results of ablation study for NDCG & MAP metrics.

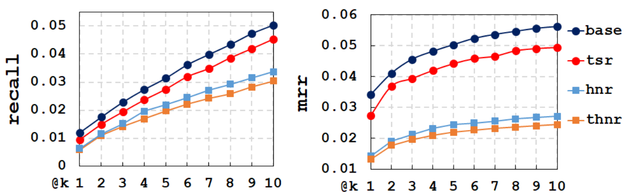

Figure 4: Results of ablation study for Recall and MRR metrics.

the dataset and the results obtained from the evaluations are publicly available[1].

Metrics. We adopt two complementary perspectives to evaluate the effectiveness of our approach, namely *ranking-based* and *quality-based* perspectives. In the ranking-based approach, we use standard ranking metrics, including mean average precision (MAP), mean reciprocal rank (MRR), normalized discounted cumulative gain (NDCG) and recall. These metrics show how many of the recommended ranked experts are a part of the gold standard team. In addition, we evaluate the quality of the recommended teams based on two metrics: skill coverage (sk) and team comparability (tc). The skill coverage metrics measures to what extent the recommended team covers the set of skills that were required for the developed team. The purpose of this metric is to reward those teams that while do not have the expected team members but still consist of experts that have the relevant expertise. The team comparability metric, on the other hand, measures the compatibility of the proposed team with that of the actual expected team. To this end, tc measures the difference between the average h-index of the proposed team and the expected team. Higher sk and a lower tc values are desirable.

Baselines. We adopt the state-of-the-work for comparing our proposed approach. From the state-of-the-art, `Kargar et al.` [8] view the team formation problem as one of identifying subgraphs within a heterogeneous collaboration network. They specifically define the problem through an optimization solution that searches for keywords (skills) over the graph. `Du et al.` [6] propose that the team (group in their context) formation process can be viewed as one of learning-to-rank experts. They specifically propose a Bayesian group ranking method for this task. `Sapienza et al.` [15] focus on the problem of team formation for the sake of developing teams that can lead to enhanced skills development by the team members. They adopt an autoencoder architecture to learn team

[1] *https://github.com/radinhamidi/Subgraph-Representation-Learning-for-Team-Mining*

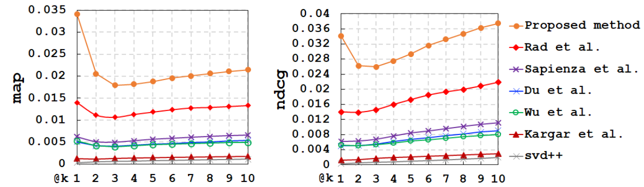

Figure 5: Comparing with baselines using NDCG and MAP metrics.

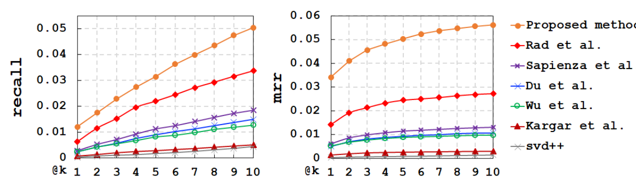

Figure 6: Comparing with baselines using Recall and MRR metrics.

representations. `Rad et al.` [14] build on the work by `Sapienza et al.` and employ a variational Bayesian neural network to learn mappings between the skill and expert spaces, which is then used to derive teams for a set of input skills. Finally, we note that the team formation task can also be viewed as one of recommending a set of experts given some skills. As such, we adopt the work by `Wu et al.` [20] that adopts a recurrent neural recommender network to learn the association between users and items. We also include the widely adopted matrix factorization approach by Koren [11]. We note that all baseline hyper-parameters were either tuned or set as defined in the corresponding papers.

3.2 Results

Ablation Study. We first perform an ablation study to investigate the impact of the variations that can be built based on our approach. There are two areas where our proposed architecture can be modified to develop variations: (1) we can substitute the subgraph representation learning component of our architecture with an alternative heterogeneous graph representation learning technique [7]. (2) we can implement a task-specific variation of our method where the representation of the subgraphs or nodes are used to learn a mapping between the skill and expert spaces based on [14]. This creates four variations of our work that we study: (1) proposed work in this paper, (base), (2) base with heterogeneous node representations (hnr), (3) task-specific mapping with subgraph representations (tsr) and (4) task-specific mapping with heterogeneous node representations (thnr).

The results of the ablation study, in Figures 3 and 4, show that our proposed approach, i.e., base has the best performance on all four ranking metrics. We further observe that the next best variation is tsr, which adopts the proposed subgraph representations but in a task-specific architecture. This shows that the adoption of subgraph representations over node-based heterogeneous graph representations is a more effective strategy. This is especially evident when comparing the thnr and tsr variations, which only differ

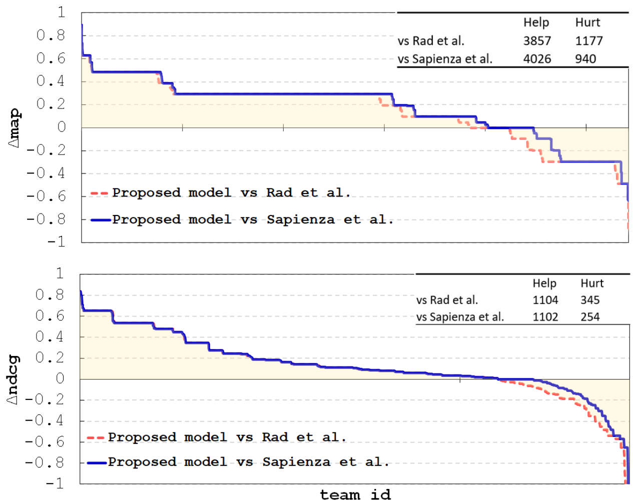

Figure 7: Performance comparison on a per-team basis.

in their adopted representations. Here, we observe that the tsr variation significantly outperforms thnr and shows that our proposed subgraph representations are more effective for team retrieval compared to heterogeneous node representations. We also find that our approach is in general more effective than the task-specific counterpart that attempts to learn explicit mappings between the skill and expert sub-spaces. This can be explained by the fact that the collaboration network is extremely sparse and hence learning a mapping between the two spaces is not very effective. In the rest of the paper when mentioning our proposed approach, we are referring to the *base* model reported in the ablation study.

Ranking Metrics. We compare our proposed approach, i.e., the *base* model, to the baselines based on ranking metrics in Figures 5 and 6. We make the following important observations: (1) our proposed approach has significantly outperformed all of the state-of-the-art baselines on all four ranking metrics. When considering team sizes of 10, the improvements shown by our method compared to the best baseline is at least 85%. (2) The next two baselines, i.e., `Rad et al.` [14] and `Sapienza et al.` [15] are based on autoencoder and variational Bayesian neural architectures. Both of these works adopt a supervised architecture and a neural mapping approach. (3) We additionally observe that the state-of-the-art graph heuristics-based team formation method, i.e., `Kargar et al.` [8], does not show competitive performance with any of the other baselines, except the simple matrix factorization model [11]. This can be due to the fact that finding optimal subgraphs associated with a keyword query (set of skills) is essentially an NP-hard problem and hence heuristic approaches such as [8] fall short of finding an optimal solution.

We also compare the performance of our approach with the best two baselines, i.e., `Rad et al.` and `Sapienza et al.` on an individual team basis. To do so, we focus on the MAP and NDCG metrics and report the difference between the average metrics value (MAP, NDCG) of each team reported by our approach compared to the baseline. This results have been reported in Figure 7. As seen in the figure, the number of teams that have been helped (improved) by our proposed approach has been substantially greater than the

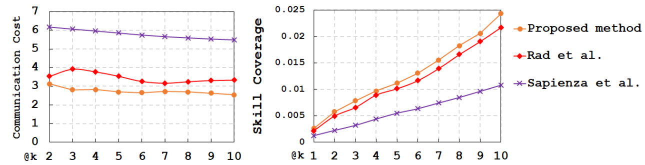

Figure 8: Quality-based comparison with best baselines.

number of teams that have been negatively impacted, i.e., 3.2 times and 4.2 times, respectively when compared to `Rad et al.` and `Sapienza et al.`

Quality Metrics. Based on the performance of the baselines on the ranking metrics, we further compare the performance of our proposed approach on quality metrics with the strongest baselines. The quality metrics include (1) skill coverage (sk) that considers the extent to which the proposed team is able to cover the requested skills and (2) team comparability (tc), which measures how similar the selected team members are compared to the members of the expected team. A high quality team would be one that has a high sk measure (sk=1 is when all skills are covered) and a low team comparability value (tc=0 shows there are no differences between the average h-index of the proposed and expected teams). We report the quality metrics in Figure 8. We analyze our findings from the perspective of the two quality metrics: (1) from the point of view of skill coverage, we find that our approach has been able to increase the sk metric by at least 12% over the best baseline (`Rad et al.` when the team size is ten, which means that the teams developed by our approach are able to satisfy the input skill requests to a greater extent. (2) from the team comparability perspective, our approach has reduced the difference between the average h-index of the proposed team and the expected team by 23.5% over the best baseline at team size ten. This means that even if the proposed team members by our approach are not the actual expected team members, they show closer resemblance to the expected team than those proposed by the two strongest baselines. The results of the ranking-based and quality-based measures show that (a) our proposed approach is able to retrieve a higher number of expected team members compared to all baselines, and (b) even when such members are not returned, the retrieved team members have higher coverage of the requested skills and a closer likeness to the expected experts.

4 CONCLUDING REMARKS

In this paper, we have modeled the problem of team mining through a subgraph representation learning process, which adopts a heterogeneous skipgram architecture. The advantage of our proposed approach is that it is quite light-weight, preserves the semantics of team structure, past collaboration experience between the experts and the relation between experts and skills. To the best of our knowledge, our work is among the first to address the team mining problem through learning heterogeneous subgraph representations from within the collaboration network. Based on our experiments on the DBLP dataset, we have shown that the proposed approach is able to show improved performance over the state of the art graph-based and neural architecture-based methods in terms of both ranking and quality metrics.

REFERENCES

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35, 8 (2013), 1798–1828.

[2] Spencer Bryson, Heidar Davoudi, Lukasz Golab, Mehdi Kargar, Yuliya Lytvyn, Piotr Mierzejewski, Jaroslaw Szlichta, and Morteza Zihayat. 2020. Robust keyword search in large attributed graphs. *Inf. Retr. J.* 23, 5 (2020), 502–524. https://doi.org/10.1007/s10791-020-09379-9

[3] Lu Chen, Chengfei Liu, Xiaochun Yang, Bin Wang, Jianxin Li, and Rui Zhou. 2016. Efficient Batch Processing for Multiple Keyword Queries on Graph Data. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*, Snehasis Mukhopadhyay, ChengXiang Zhai, Elisa Bertino, Fabio Crestani, Javed Mostafa, Jie Tang, Luo Si, Xiaofang Zhou, Yi Chang, Yunyao Li, and Parikshit Sondhi (Eds.). ACM, 1261–1270. https://doi.org/10.1145/2983323.2983806

[4] Lu Chen, Chengfei Liu, Rui Zhou, Jiajie Xu, Jeffrey Xu Yu, and Jianxin Li. 2020. Finding Effective Geo-social Group for Impromptu Activities with Diverse Demands. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash (Eds.). ACM, 698–708. https://doi.org/10.1145/3394486.3403114

[5] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 135–144.

[6] Yulu Du, Xiangwu Meng, Yujie Zhang, and Pengtao Lv. 2020. GERF: A Group Event Recommendation Framework Based on Learning-to-Rank. *IEEE Trans. Knowl. Data Eng.* 32, 4 (2020), 674–687. https://doi.org/10.1109/TKDE.2019.2893361

[7] Amin Milani Fard, Ebrahim Bagheri, and Ke Wang. 2019. Relationship Prediction in Dynamic Heterogeneous Information Networks. In *Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 11437)*, Leif Azzopardi, Benno Stein, Norbert Fuhr, Philipp Mayr, Claudia Hauff, and Djoerd Hiemstra (Eds.). Springer, 19–34. https://doi.org/10.1007/978-3-030-15712-8_2

[8] Mehdi Kargar, Lukasz Golab, Divesh Srivastava, Jaroslaw Szlichta, and Morteza Zihayat. 2020. Effective Keyword Search over Weighted Graphs. *IEEE Transactions on Knowledge and Data Engineering* early access (2020). https://doi.org/10.1109/TKDE.2020.2985376

[9] M. Kargar, M. Zihayat, and A. An. 2013. Finding Affordable and Collaborative Teams from a Network of Experts. In *SDM*. 587–595.

[10] Richard M. Karp. 1972. Reducibility Among Combinatorial Problems. In *Proceedings of a symposium on the Complexity of Computer Computations*. Plenum Press, New York, 85–103.

[11] Yehuda Koren. 2009. Collaborative filtering with temporal dynamics. In *KDD*.

[12] T. Lappas, L. Liu, and E. Terzi. 2009. Finding a Team of Experts in Social Networks. In *KDD*. 467–476.

[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *NIPS*. 3111–3119.

[14] Radin Hamidi Rad, Ebrahim Bagheri, Mehdi Kargar, Divesh Srivastava, and Jaroslaw Szlichta. 2021. Retrieving Skill-Based Teams from Collaboration Networks. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tetsuya Sakai (Eds.). ACM, 2015–2019. https://doi.org/10.1145/3404835.3463105

[15] A. Sapienza, P. Goyal, and E. Ferrara. 2019. Deep Neural Networks for Optimal Team Composition. *Frontiers in Big Data* 2, 1 (2019), 1–13.

[16] Fabian Schliski, Jörg Schlötterer, and Michael Granitzer. 2020. Influence of Random Walk Parametrization on Graph Embeddings. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12036)*, Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins (Eds.). Springer, 58–65. https://doi.org/10.1007/978-3-030-45442-5_8

[17] Yizhou Sun and Jiawei Han. 2012. *Mining Heterogeneous Information Networks: Principles and Methodologies*. Morgan & Claypool Publishers. https://doi.org/10.2200/S00433ED1V01Y201207DMK005

[18] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S. Yu, and Tianyi Wu. 2011. PathSim: Meta Path-Based Top-K Similarity Search in Heterogeneous Information Networks. *Proc. VLDB Endow.* 4, 11 (2011), 992–1003. http://www.vldb.org/pvldb/vol4/p992-sun.pdf

[19] Yahui Sun, Xiaokui Xiao, Bin Cui, Saman K. Halgamuge, Theodoros Lappas, and Jun Luo. 2021. Finding Group Steiner Trees in Graphs with both Vertex and Edge Weights. *Proc. VLDB Endow.* 14, 7 (2021), 1137–1149. http://www.vldb.org/pvldb/vol14/p1137-sun.pdf

[20] Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J Smola, and How Jing. 2017. Recurrent recommender networks. In *WSDM*. 495–503.