Knowledge Graph-based Embedding for Connecting Scholars in Academic Social Networks

Xiyao Cheng[1], Yuanxun Zhang[2], Harsh Joshi[1], Mayank Kejriwal[3], Prasad Calyam[1]
[1]Department of EECS, University of Missouri-Columbia, USA; [2]Department of Data, Weee!, USA;
[3]University of Southern California, USA
Email: *xcheng@mail.missouri.edu, bigfishinriver@gmail.com, hjqmv@mail.missouri.edu, kejriwal@isi.edu, calyamp@missouri.edu*

***Abstract*—In recent years, research tasks have increasingly involved using multi-disciplinary knowledge through collaborations of scholars from multiple fields. However, identifying a team of suitable collaborators from diverse fields for a given research task is a challenging and time-consuming process. In this paper, we propose a novel "ScholarTeamFinder" model that uses knowledge graph based link prediction to identify collaborators within an academic social network (ASN) to form a research team to address a multi-disciplinary research problem. Our approach involves building a heterogeneous knowledge graph within an ASN using entities such as scholars, publications, research grants, and the relationship among these entities. Following this, we use graph-based deep learning to learn the node embedding from the knowledge graph that can be used for scholar team recommendation. More specifically, we used the classical *metapath2vec* as our base graph learning algorithm and improved its performance by considering semantic meaning of entities and encoding edge embeddings in the graph. Finally, we propose a beam-search algorithm for scholar team prediction based on our model embeddings. Our evaluation of ScholarTeamFinder is performed using large ASN datasets including a unique dataset (i.e., NSF award dataset) of federal grant awards collected over the last ten years and the scholars' publication data, as well as three other widely used datasets (i.e., APS, SCHOLAT and Gowalla). Experiment results show that our model outperforms the state-of-the-art models across the different datasets.**

***Index Terms*—heterogeneous knowledge graph, scholar team recommender, deep learning, node embedding, link prediction**

I. Introduction

We are witnessing an information explosion, marked by a rapid increase in published content and easily accessible data archives. Consequently, advanced search engines (e.g., Google, Bing) and recommendation systems (e.g., Amazon, Netflix) have become crucial for efficiently filtering pertinent information. In academia, research tasks demand not only the discovery of relevant multidisciplinary data, but also the identification of suitable experts across various fields [1]. Considering the dynamic evolution of scholars' profiles over time, influenced by factors such as publications and research grants, manually identifying collaborators for multidisciplinary research problems poses a significant challenge.

Recent work in [2] employs a knowledge graph based model to recommend individual scholars. They propose a heterogeneous network-based approach to recommend scholar-friends by leveraging the entity and relationship data in online academic communities, or Academic Social Networks (ASNs). Alternately, authors in [3] propose a personalized scholar recommendation model using multi-dimensional features. In addition, in [4], a graph embedding-based scholar recommendation system is proposed by leveraging auxiliary academic information. Authors in [5] apply the Node2vec and Multi-node2vec algorithms to predict collaborations between authors in the gene editing field. However, in these prior works, recommendations of scholar friends or similar scholars through building of an ASN which does not include link labels as needed in a knowledge graph. Moreover, these models are based on the data from online academic community platforms, which might have inconsistent information. Furthermore, most prior models focus on recommending single scholar in a particular field rather than identifying *scholar teams* with multi-disciplinary expertise.

As shown in Fig. 1, we define a scholar team as a research group of two or more scholars who come from the same or different institutions, and focus on synergistic research areas that collectively are essential to solve a multi-disciplinary research problem. Identifying suitable collaborators amongst numerous scholars is a challenging problem. In the example shown, scholars from each discipline (i.e., computer science, health care, communication) bring distinct knowledge about solving a research problem involving: (a) a science gateway tool that is used to collect data related to a drug or gene knowledge discovery related to a medical diagnosis, and (b) an analysis that is performed for finding a cure. Once a cure is found, principles of communication are crucial to measure the benefits of the solution and determine how to diffuse its adoption across stakeholders in a health care community.

To find the suitable team of scholars, current practice where a scholar finds another scholar to collaborate involves a manual and trial-and-error process. This process is obviously limits the ability to efficiently and accurately identify the relevant potential collaborators and there is a need for a data-driven process. Academic Social Networks (ASNs) provide access to information about hundreds of thousands of scholars and many links between scholars e.g., some might have already worked closely with each other, and others might be within a common organizations (e.g., university or university system). However, there are cases where scholars do not inherently have commonality, and there are innumerable scholars who experience far too many changes in their academic positions/interests. In such cases, it is highly challenging to track scholars and characterize their latest research interests.

This material is based upon work supported by the National Science Foundation under Award Number OAC-2006816. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

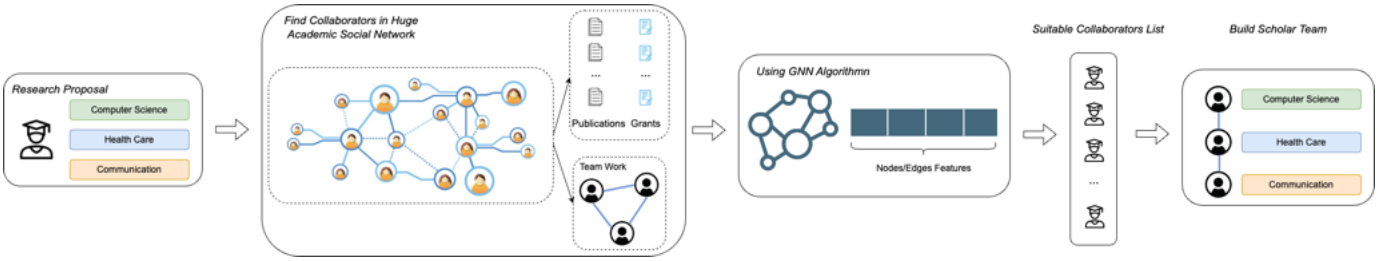


Fig. 1: Example to show how a potential scholar team is identified to address a multi-disciplinary research problem.

In this paper, we address the above problem by proposing a "ScholarTeamFinder" model to recommend potential scholars that can collaborate as a scholar team to solve a multi-disciplinary research problem. As a solution approach, we convert the scholar recommendation to a scholars link prediction problem. At the first step, we create a knowledge graph with scholars related information in an ASN. Then, we embed the knowledge graph with our modified *metapath2vec* model that considers semantic features of the nodes and encodes metapath edges types in the learning process. Such an approach can help the models to use more features of nodes and edges to learn better representations of the entities.

To develop our ScholarTeamFinder model, we use a novel "follow-the-money" strategy, that involves a large collection of the NSF (National Science Foundation) grant awards dataset [6]. This unique dataset has been collected over the last ten years and contains more than 120,000 award records (with project names and abstracts), and 60,714 scholars who received grants. In addition, we collect these scholars' publications and related information from Google Scholar, then add them into our graph for providing more research information for the model. Empirically, we apply this dataset to our ScholarTeamFinder model to evaluate it. To show broad applicability of our ScholarTeamFinder, we also use other large datasets that are widely used (i.e., APS [7], SCHOLAT [8] and Gowalla [9]). Experiment results are performed to show that our ScholarFinder model outperforms state-of-the-art baseline models (e.g., *deepwalk*, *node2vec*, *graph convolutional network* and *metapath2vec*) across the different datasets.

The remainder of this paper is organized as follows: Section II discusses the related works. Section III introduces our proposed ScholarTeamFinder model. In Section IV, we detail the methodology used to develop our ScholarTeamFinder model. In section V, we discuss the datasets used in evaluation experiments and present the model performance results. Section VI concludes the paper.

II. Related Work

A. Academic Social Networks

Academic Social Networks (ASNs) are similar to social networking sites, but are designed for the academic community. Nowadays, ASNs have gotten widespread attention in academia and industry due to the overwhelming production of the scholarly data [10]. Further, various academic data can be easily obtained online from multiple sources using web services, which makes it easier for building ASNs. Such an increased accessibility of open datasets via ASNs has sparked many new research problems in this area involving e.g., knowledge extraction [11] [12], and recommendation systems [13] [14] [15].

In terms of recommender applications on ASNs, there are specific recommendation methods proposed in prior works such as e.g., HeteroRWR [16] proposed an approach to recommend top k co-authors for a target scholar by implementation of multiple random walks in a heterogeneous network, which integrates a citation network and a co-authorship network. A scientific article recommendation approach was presented in [17] based on the discovery of thematic community structures. Therein, the authors build a topic model based on a community detection method to recommend scholars and academic articles. Authors in [18] analyze the 'global cooperation and correlation' for each pair of authors in an ASN, and grade these relationship to recommend scholars. Similarly, authors in [19] propose a community-based scholar recommendation model in ASNs by constructing research fields-based graphs, and then make scholar recommendations by calculating friendship scores. Nevertheless, most prior works focus on only one aspect of ASNs which is to identify a suitable scholar for a given research topic, or the authors build the ASNs only with one aspect of information, like scholar's papers and related information. In this work, we leverage scholar funding records and build an ASN with a unique NSF award dataset to develop scholar team recommendations using deep learning strategies.

B. Representation Learning

Machine learning on graphs is an important and ubiquitous task with applications ranging from drug design to friend recommendations within social networks [20]. During the model training process, finding a way to represent is a very important but also a difficult problem. A good representation method can affect the performance of the final model directly. Moreover, a good representation learning approach can extract useful information from speech, text, and figures, and can help researchers to save time in the process of handcrafting features.

Recently, there have been many research works on representation learning in a knowledge graph context. Many researchers focus on homogeneous knowledge graphs as in e.g., DeepWalk [21], Line [22], node2vec [23]. These models process particular kinds of nodes in a knowledge graph. However, with the information explosion and diversity, a homogeneous knowledge graph cannot satisfy the actual network's requirement. Consequently, more recent works focus on heterogeneous knowledge graph methods.

For instance, authors in [24] design a deep embedding algorithm for networked data which can capture the complex interactions between the heterogeneous data in a network. Similarly, authors in [25] propose a unified framework to solve the problem of embedding learning for an Attributed Multiplex Heterogeneous Network. The authors of [26] propose the multi-perspective social recommendation (MPSR), to construct hierarchical user preferences and assign friends' influences with different levels of trust from varying perspectives. In addition, the work in [27] proposed two scalable representation learning models, *metapath2vec* and *metapath2vec++* that can embed the graph features with a high performance in many data mining tasks. These existing models can extract the features from a knowledge graph efficiently. However, they do not consider the nodes' text features.

Based on the above point, we expand the *metapath2vec* model by adding semantic features and edge types to embed our knowledge graph to perform a scholar team recommendation. In our ScholarTeamFinder model, we also go beyond prior state-of-the-art models by combining knowledge graph embedding and link prediction, and propose a novel model that can recommend scholars to help build a high-quality research team, which is crucial to solve multi-disciplinary research problems requiring multi-disciplinary experts.

III. Heterogeneous Graph Learning

A. metapath2vec Model

Our proposed ScholarTeamFinder model leverages the *metapath2vec* [27] model to learn node embeddings from the knowledge graph. The *metapath2vec* model is proposed to learn nodes embedding for heterogeneous graphs, which use random walks based on predefined metapaths to generate the heterogeneous neighborhood of each vertex, and then uses the Skip-Gram model [23] to learn node embeddings.

In the *metapath2vec* model, for a heterogeneous graph $G = (V, E, T)$, where V represents the nodes in the G, E represents the edges, T represents the types of graph nodes and their relationships, and when $|T_E| + |T_V| > 2$, it represents the heterogeneous graph. In the training stage, it maximizes the probability of having the heterogeneous context $N_t(v), t \in T_V$ given a node v:

$$\arg\max_{\theta} \sum_{v \in V} \sum_{t \in T_v} \sum_{c_t \in N_t(v)} log p(c_t|v;\theta) \tag{1}$$

where $N_t(v)$ denotes node v's neighborhood with the t^{th} type of nodes and $p(c_t|v;\theta)$ is commonly defined as a softmax function as

$$p(c_t|v;\theta) = \frac{e^{X_{c_t} X_v}}{\sum_{u \in V} e^{X_u X_v}} \tag{2}$$

where X_v is the v^{th} row of X, representing the embedding vector for node v.

metapath2vec applies negative sampling [28] to efficiently optimize the Equation (1). Given a negative sample size M, the Equation (1) can be transformed as followed: $log\sigma(X_{c_t} \cdot X_v) + \sum_{m=1}^{M} \mathbb{E}_{u^m \sim P(u)}[log\sigma(-X_{u^m} \cdot X_v)]$, where $\sigma(x) = \frac{1}{1+e^{-x}}$ and $P(u)$ is the pre-defined distribution from which a negative node u^m is drew from for M times. *metapath2vec* builds the node frequency distribution by viewing different types of nodes homogeneously and draws negative nodes regardless of their types.

In addition, *metapath2vec* uses random walk to capture both semantic and structural relationships among different types of nodes, which transforms the heterogeneous graph into multiple meta-paths. One meta-path scheme $\mathbb{P}$ is defined as: $V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots V_t \xrightarrow{R_t} V_{t+1} \cdots \xrightarrow{R_{l-1}} V_l$, where R_i represents the relationships between node types of V_i and V_{i+1}. Note that the transition probability at step i is defined as follows:

$$p(v^{i+1}|v_t^i, \mathbb{P}) = \begin{cases} \frac{1}{|N_{t+1}(v_t^i)|} & (v^{i+1}, v_t^i) \in E, \phi(v^i + 1 = t + 1) \\ 0 & (v^{i+1}, v_t^i) \in E, \phi(v^i + 1 \neq t + 1) \\ 0 & (v^{i+1}, v_t^i) \notin E \end{cases} \tag{3}$$

where $v_t^i \in V_t$, $N_{t+1}(v_t^i)$ denote the V_{t+1} type of neighborhood of node v_t^i. Thus these three situations mean: (1) there is an edge between two nodes, and the next node type belongs to the defined metapath's next nodes' type, (2) there is an edge between two nodes, and the next node type does not belong to the defined metapath's next nodes' type, and (3) there is no edge between the two nodes. With this strategy, the model ensures that the semantic relationships between different types of nodes can be properly incorporated into a skip-gram.

B. metapath2vec Improvements

More importantly, we have improved the *metapath2vec* model from two aspects for achieving better representation learning performance: (1) we initialize the node embedding weights with the pre-trained BERT embeddings; (2) we also encode edge embedding in the *metapath2vec* model and jointly learn with node embedding to achieve better performance.

1) Initializing node embedding weights using the pre-trained BERT embeddings: The original *methpath2vec* model uses only node id feature in the model that initialize s nodes embeddings randomly, and then learn their node embeddings using the Skip-gram based method, which is not efficient to learn from a large graph.

In order to learn the node embeddings more efficiently, we initialize the node embeddings using pre-trained Sentence BERT embeddings [29]. In our work, each node (such as "proposal", "scholar" and "publication") has a semantic meaning. For example, we can directly get scholar nodes' semantic meanings from their short bio or research interests; publications can be gotten from their abstracts. Hence, we first collect these text information for each node, and then we use Sentence BERT to encode the text to generate the text embedding. Finally, we use these text embeddings to initialize node embedding weights for each node. Hence, similar proposals or scholars tend to be close initially, and this in turn helps to improve the model performance.

2) Learning edge embedding: The original *methpath2vec* ignores edge information in the graph, which is not reasonable in a heterogeneous graph setting with different types of edges. For example, a scholar can have "work at" edge at some institutions, and have "writes" edge on some papers. Hence, *metapath2vec* ignores the edge type and learn scholars

embeddings' from neighboring nodes equally. In other words, it treats institutions nodes and papers nodes equally, which may result in some difficulties to differentiate institutions' nodes and paper nodes.

In order to overcome these limitations, we consider edge embedding for different types t of edges Y_t to learn node embedding and edge embedding simultaneously, and the edge embedding has same dimension with node embedding.

We encode the edge type t embedding Y_t using a Hadamard product (or element-wise product) with node embedding X_{c_t} as $X_{c_t} \odot Y_t$. Then, we can modify Equation 2

$$p(c_t|v;\theta) = \frac{e^{(X_{c_t} \odot Y_t)(X_v \odot Y_t)}}{\sum_{u \in V} \sum_{t \in T} e^{(X_u \odot Y_t)(X_v \odot Y_t)}} \tag{4}$$

where X_v is the v^{th} row of X, representing the embedding vector for node v. We can also use negative sampling to optimize the Equation 4.

C. Link Prediction Model

As shown in Fig. 2, after learning the embedding from the knowledge graph, based on this, then we predict the links between scholars. In our model, we extract the collaborate relationship between scholars. For example, one scholar s_1 is the PI of a proposal, and another scholar s_2 is the co-PI of the same proposal or they published one publication. In this case, we conclude that they have collaborative relationship with each other. According to this, we get the positive samples. For the negative samples, we randomly sample from the negative candidates pools. For a particular scholar, we define his/her negative candidate pools if the scholars do not work on the same proposal or the same publication. For every sample, we predict if there is a possible connection between scholar s_1 and scholar s_2 by computing the cosine similarity score with Equation 5, where the x_1, x_2 denote the scholar s_1, s_2's embeddings and y_t denotes the edge embedding between scholar s_1 and scholar s_2, which can be learnt from our revised *methpath2vec* model. This score can help us to find the similarity score between two scholars, and in turn we can find whether there is a possible link between them.

$$score(s_1, s_2) = \frac{(x_1 \odot y_t) \cdot (x_2 \odot y_t)}{(||x_1 \odot y_t||||x_2 \odot y_t||)} \tag{5}$$

IV. ScholarTeamFinder Methodology

As shown in Fig. 3, there are four major components in our ScholarTeamFinder: (i) Web Crawler; (ii) Data Pre-processing; (iii) Deep Learning model based on knowledge graph; and (iv) Visualization. In the following, we detail each of the above components.

A. ScholarTeamFinder Components

1) Web Crawler: We use a Web Crawler to download the NSF award dataset from the official website and extract scholar information from the organized awards data. Next, we gather scholars' publications and related details such as title, publication year, and journal name using a custom Python script. The raw data from Google Scholar is stored in *MongoDB*, and another Python script extracts relevant data for further pre-processing.

2) Data Pre-processing: This component is essential for processing the dataset to create a knowledge graph. As our data comes from various sources, we first structure the data. Data pre-processing sub-steps include: lowercasing text; removing frequent, non-meaningful stop words (e.g., "the", "a"); performing lemmatization to group inflected word forms; and applying tokenization.

3) Deep Learning Model based on Knowledge Graph: Our system uses the ScholarTeamFinder model to generate node embedding using the knowledge graph we created. With the node vectors, we perform link prediction through computing similarity. After finishing the training procedure, the model can be used for future link prediction which identify potential collaborators for scholars, and visualization. Based on these results, we propose a method for constructing high-quality research teams.

4) Model Visualization: We use visualization to monitor the model training status e.g., plotting the loss curve. In this way, we can observe the process of training models so that we can update model's configurations easily. Also, we use this component to evaluate the model performance.

B. Knowledge Graph Building

To construct our knowledge graph, we collect ten years of NSF award data, comprising 121,102 proposals and associated information, such as title, abstract, NSF organization, PI, and Co-PI details. From this, we obtain data for 60,714 scholars and use a web crawler to gather their publications and research interests, enabling the creation of our heterogeneous knowledge graph.

Knowledge graphs allow multiple entity types and relationships to co-exist within the same schema [30]. Our knowledge graph includes entity nodes such as *"Scholar"*, *"Proposal"*, *"Division, Org"*, *"Institution"*, *"Place"*, *"Publication"*, *"Journal"* and *"Publish_Year"*, as well as types of relationship edges such as *"writes"*, *"is_written"*, *"is_supported_by"*, *"includes"*, *"belong_to"*, *"collaborate_with"*, *"is_published_by"*, and *"is_published_at"*, shown in Fig. 4.

For entity nodes type, the nodes labeled *"Scholar"* represent the scholar who has been granted the NSF award, the *"Proposal"* represents the award, *"Division, Org"* show which NSF organization supports the award, *"Institution"* means the scholar's affiliated institution, and *"Place"* indicates the scholar location. *"Publication"*, *"Journal"* and *"Publish_Year"* respectively show the publications scholars write, the publisher, and the publication year.

Based on the entity nodes, we set related graph edges to represent the relationship between them. *"writes"* and *"is_written"* express the relationship between scholars and proposals. According to the NSF organization to which the proposal belongs, we build the *"is_supported_by"* edge to describe it. Similarly, for scholars' publications, they also have the relationship of *"writes"* and *"is_written"*. For *"Publication"*, *"Journal"* and *"Publish_Year"* nodes, their relationship are *"is_published_by"* and *"is_published_at"*. The last part relationships are for scholars, the first one is *"Scholar"* belong to *"Institution"*, it means scholar is affiliated with this

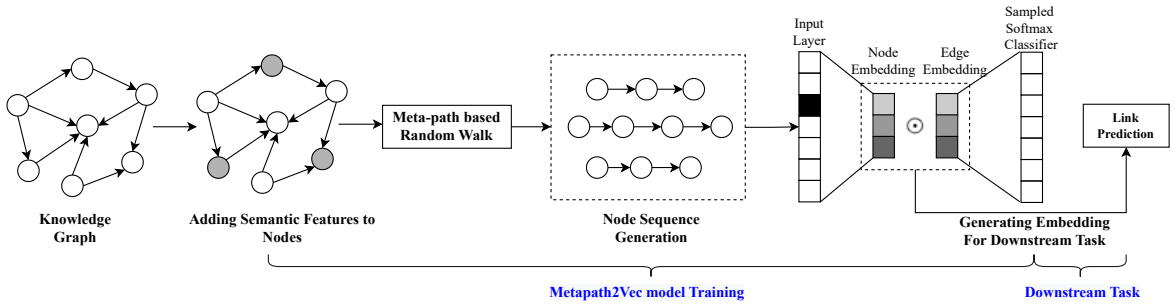


Fig. 2: ScholarTeamFinder model architecture based on knowledge graph.

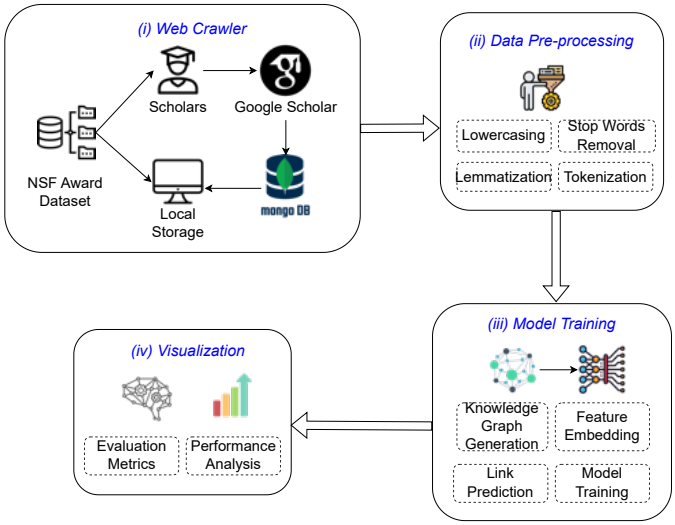


Fig. 3: Major components of the ScholarTeamFinder.

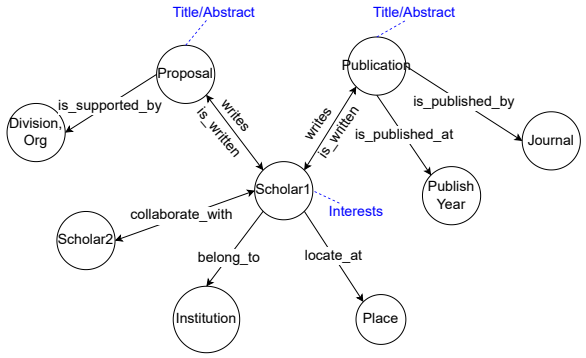


Fig. 4: The data schema of the proposed academic knowledge graph with connections of properties associated with the entities (Title/Abstract, Interests) of the knowledge graph.

institution. The second one shows the location of the scholar. The last one, *"collaborate_with"*, represents the collaboration relationship between scholars. In addition to identifying all nodes and edges, we also add some features for specific nodes. For example, for *"Proposal"* or *"Publication"* nodes, we add its text of title and abstract as a nodes' feature, and for *"Scholar"* nodes, we add scholar's research interests as scholars' features.

C. Scholar Team Types in ASNs

Herein, we introduce the definition of a scholar team, and how ScholarTeamFinder can help scholars in building their own scholar team. In this work, we use link prediction to predict the probability of connection between two scholars or if they have collaborated together, also extend link prediction for scholar team prediction that can help scholars find a high quality scholar team to do some multi-disciplinary research projects. We consider two types of scholar teams that can be defined as follows:

- *Finding an existing scholar team*: Currently, there are increasing number of research problems that need knowledge from two or more domains to determine a solution. Consequently, scholars are needing to find cross-field collaborators to build their research teams. In our model, if we can find a scholar that is most related to a target scholar, according to this scholar, we can find his/her collaborators through following the graph's *"collaborate_with"* edge. In this way, we can use our knowledge graph and deep learning model to provide the recommendation about the existed scholar team.
- *Generating a new scholar team through identification of related scholars*: Our model also can help scholars find several related scholars to build their own team. Our model can predict the links between scholars. Then scholars can use the predicted result to find specific field experts according to scholars research interests; with the related scholars' information, they can build their own scholar team for their research project.

D. Scholar Team Prediction

As noted earlier, link prediction is used to predict the probability of whether two scholars could be connected or have collaborated together. In this paper, we also extend our link prediction to scholar team prediction to help scholars find a high quality scholar team to work on multi-disciplinary research problems. Hence, we proposed a beam-search based algorithm to achieve this. A beam-search algorithm [31] is a greedy algorithm that is widely used in the area of natural language processing to find a sub-optimal solution of an output sequence.

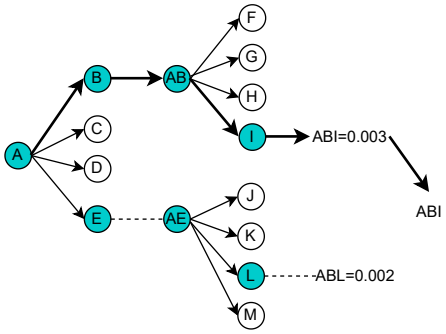


Fig. 5: Beam search algorithm for identifying a scholar team with $K = 2$.

In our scholar team finder problem, we usually have a target scholar s and a research problem t initially. Our goal is identify K ideal candidates to work on the research problem, which can be formulated mathematically,

$$\arg\max_{c} \sum_{i=1}^{K} logp(c_i|s,t) \tag{6}$$

We expand the Equation(6) using a chain-rule, where we only consider the first degree of connection for the purpose of performance as considered in the example of when k = 3,

$$\arg\max_{c} \ (logp(c_1|s,t) + logp(c_2|c_1,t) + logp(c_3|c_2,t)) \tag{7}$$

We improve the *methpath2vec* model with sentence Bert, and thus their embedding contains research topics. Following this, the probability of $logp(c_1|s,t)$ can be easily calculated by Equation(5). Subsequently, we can preform a beam-search algorithm shown in Fig. 5 to find a scholar team based on a research problem topic. In the example of Fig. 5, the Scholar A, scholar B, and scholar I are selected to form a scholar team.

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of our model with state-of-the-art *deepwalk* [21], *node2vec* [23], *GCN* and *metapath2vec* [27] models. Following this, we demonstrate how ScholarTeamFinder model has better performance, and use a actual case to show our model's downstream steps and validity.

A. Datasets

For our evaluation experiments, in addition to our NSF award dataset, we also choose two public datasets i.e., APS [7] and SCHOLAT [8] and Gowalla [9] as described below. And we organized the statistics information of the datasets in table I.

- *NSF award dataset*: We collected an unique open and large scholar related dataset based on NSF Awards [6], which span multiple research fields. This dataset summarizes grant awards information and which scholars received any of the awards over the last ten years. The main information in this dataset is awards name, description, PI's name, email, and PI's institute name, location, etc. We extract PI information to get their publications from Google Scholar. We collected 121,102 awards from the NSF award dataset, which contains information that is relevant to 60,714 scholars who successfully obtained competitive funding from NSF grants over the past ten years. And for every proposal's description, we collect the title and abstract, and then embed them to a proposal node's feature. We also collect the scholars' publications information, which includes title, abstract, journal, publication year. We put the publications into our knowledge graph and embed title or abstract as publication nodes features.
- *APS* [7]: APS (American Physical Society) collects publications information from 18 core physics journals, and store these information as JSON format. We select PRA (Physical Review A: Atomic, Molecular, and Optical Physics) journals to construct the APS dataset. We extract the publications information which include publication's title, authors, and use co-author relationship to be the collaboration relationship between the scholars.
- *SCHOLAT* [8]: It is an emerging vertical ASN system designed and built specifically for researchers in China. The main goal of SCHOLAT is to enhance collaboration and social interactions focused on scholarly and learning discourses among the community of scholars. In addition to social networking capabilities, SCHOLAT also incorporates various modules to encourage collaborative and interactive discussions, for example, chat, email, events, and news posts. In our experiment, we collect 10,607 scholars and 168,540 collaboration relations between them.
- *Gowalla [9]*: It is a location-based social networking website where users share their locations by checking-in. The friendship network is undirected and was collected using their public API, and consists of 196,591 nodes and 950,327 edges. In this paper, we extract 4,141 nodes and 10,324 relationships for deriving our dataset.

TABLE I: Statistics of the datasets used in our study.

Datasets	#Nodes	#Relationships	#Text Features
APS	10,651	24,360	10,508
Scholat	10,607	168,540	10,607
Gowalla	4,141	10,324	-
NSF	60,714	19,336	121,102

B. Experimental Setup

1) Metrics: We use NDCG@K, Precision@K, and Recall@K as our evaluation metrics, and we choose K = {5, 10, 20, 30, 40, 50, 100}. Our model generates a set of scholars as potential collaborators for a given research problem input.

- *NDCG@K* ((Normalized Discounted Cumulative Gain)) is a measure that evaluates the quality of the ranking by considering both the relevance and the order of the retrieved items.

$$NDCG@K = \frac{DCG@K}{IDCG@K} \tag{8}$$

$$DCG@K = \sum_{i=1}^{k} \frac{2^{r_i - 1}}{log_2(i+1)} \tag{9}$$

where K is the length of recommendation list, i is the ith recommend scholar. r_i is whether the ith recommend scholar is the actual link with the target scholar. And *IDCG@K* and *DCG@K* have the same formula, the difference is $r_i = 1$ is put at the beginning of recommendation list, and $r_i = 0$ at the last of recommendation list.

- *Precision@K* is the proportion of relevant items among the top-k retrieved items. It measures how accurate the system is at finding relevant items within a fixed number of results. The higher precision@K means the more of the top-k items are relevant.

$$Precision@K = \frac{1}{k} \sum_{i=1}^{k} \frac{R_a \bigcap T_a}{R_a} \tag{10}$$

where k, R_a, and T_a represent the number of scholars, the predicted list of scholars, and the ground-truth scholars associated with scholar i, respectively.

- *Recall@K* is the proportion of relevant items found in the top-k retrieved items out of all possible relevant items. It measures the system's ability to retrieve all relevant items within a fixed number of results. The higher recall@K means the model can find relevant items within the specified range.

$$Recall@K = \frac{1}{k} \sum_{i=1}^{k} \frac{R_a \bigcap T_a}{T_a} \tag{11}$$

where k, R_a, and T_a represent the number of scholars, the predicted list of scholars, and the ground-truth scholars associated with scholar i, respectively.

2) Baseline Methods: To evaluate our proposed ScholarTeamFinder model, we consider two state-of-the-art methods that include deep learning and traditional machine learning models. These models are trained using our knowledge graph.

- *deepwalk* [21]: It is a type of graph neural network. It directly works on the graph structure. It uses a randomized path traversing technique to learn the inner structure of the graph network. It has been applied to many areas with satisfactory performance.
- *node2vec* [23]: It is a widely-used algorithm for learning continuous node embeddings in network graphs. Extending the skip-gram model, it preserves the network's structural properties through random walks. The algorithm generates multiple random walks starting from each node and treats these random walks as sequences. Node2vec optimizes node representation based on context, capturing both local and global graph properties.
- *Graph Convolutional Networks (GCN)*: It is a class of deep learning models tailored for graph-structured data. By applying convolution-like operations on graph nodes, they extend traditional CNNs and learn meaningful node representations while capturing the graph's local structure and features. GCNs are versatile for various applications across different domains.
- *metapath2vec* [27]: It is a network embedding method that is suitable for a heterogeneous network. It relies on random walk which is based on meta path and uses the Skip-Gram model for embedding network nodes.

3) ScholarTeamFinder: As discussed in Section III, we propose a novel model that can predict the link between scholars with knowledge graph embedding. In the evaluation experiments, we compared our ScholarTeamFinder model's performance with other state-of-the-art models.

C. Evaluation Results

For evaluation of ScholarTeamFinder model's performance, we compare with other baseline models on the different datasets, and adopt NDCG@K, precision@K and Recall@K as the evaluation metrics for the results.

1) Model performance for different datasets and metrics: As shown in Table II, we compared ScholarTeamFinder performance with *deepwalk*, *node2vec*, *GCN* and *metapath2vec*. From the results, we can observe that our ScholarTeamFinder shows better performance in comparison to the baseline models in terms of the NDCG@K score. More specifically, especially compared to *deepwalk* and *node2vec*, the performance is improved by at least 0.3 on the APS dataset. In addition, our model clearly outperforms *GCN* and *metapath2vec*. This in turn, means that our embedding method is more effective to perform link prediction between scholars. The reason is that we add node features (sentence Bert embedding) and edge embedding into the model that can capture more information between and within nodes. However, the other models only consider id information to capture hidden relationships between node entities.

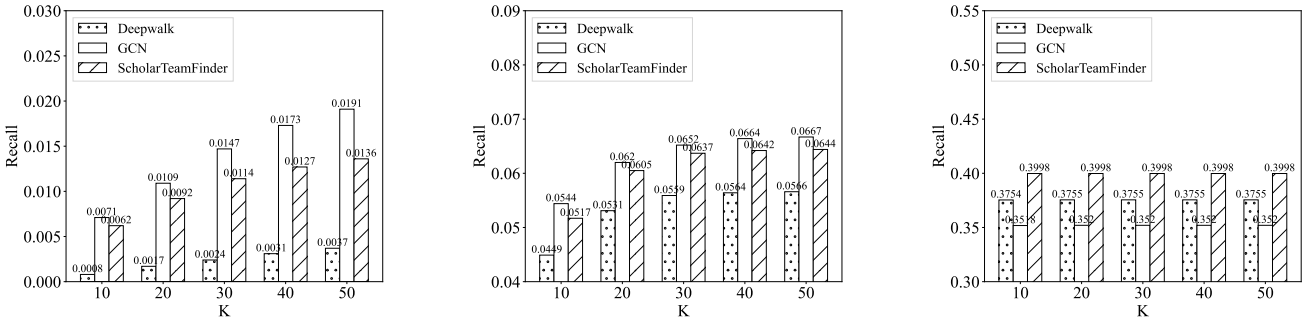
In terms of the two datasets, the APS dataset has less nodes but has more relationships between the nodes, and the NSF dataset has more nodes and less relationships. However, the NSF dataset can provide more text features and edge features for the models, and thus enables the models to have better performance results. Also, since we collect NSF award datasets and Google Scholar datasets separately and then combine them into one dataset, there is more noise in it than the APS dataset. Hence, ScholarTeamFinder has the better performance on the APS dataset, although the advantage is slightly significant.

Further, we also compare the models' performance on other datasets in terms of Precision@K and Recall@K as shown in Fig. 6 and Fig. 7. Here, we compare the results on Scholat, Gowalla and NSF datasets. We can see that *GCN* has better performance on Scholat and Gowalla datasets. Also, for the NSF dataset, ScholarTeamFinder shows better performance. Scholat and Gowalla datasets do not include many text features and two more link types between the nodes - consequently, the ScholarTeamFinder model cannot show its advantage on them. On the NSF dataset, ScholarTeamFinder has better performance with Recall@K score, and *GCN* has better result with Precision@K. This is because - the NSF dataset can provide more information of the graph nodes and more link types, and thus allow our model to more effectively extract features from them.

2) ScholarTeamFinder Model with and without edge embedding performance comparison: Compared to *metap-*

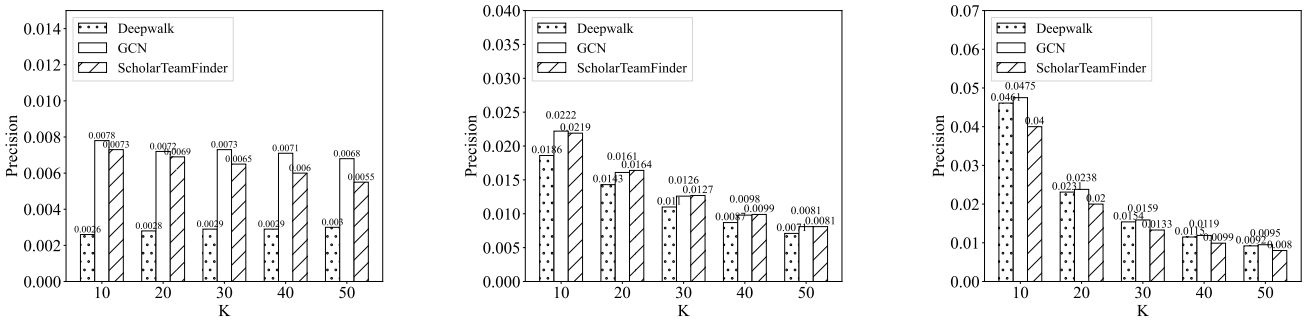
TABLE II: Results of our proposed ScholarTeamFinder model comparison with state-of-the-art models in terms of NDCG@K.

NDCG@K	Datasets	*Deepwalk*	*Node2vec*	*GCN*	*Metapath2vec*	*ScholarTeamFinder*
5	*APS*	0.0581	0.0904	0.0734	0.3104	**0.4045**
	NSF	0.2634	0.3262	0.2956	0.3314	**0.3510**
10	*APS*	0.0605	0.0964	0.0797	0.3185	**0.4045**
	NSF	0.2636	0.3267	0.2981	0.3316	**0.3510**
20	*APS*	0.0622	0.0990	0.0823	0.3202	**0.4045**
	NSF	0.2636	0.3267	0.2982	0.3316	**0.3510**
30	*APS*	0.0628	0.0996	0.0829	0.3204	**0.4045**
	NSF	0.2636	0.3267	0.2982	0.3316	**0.3510**
40	*APS*	0.0629	0.0998	0.0830	0.3205	**0.4045**
	NSF	0.2636	0.3267	0.2982	0.3316	**0.3510**
50	*APS*	0.0630	0.0998	0.0830	0.3205	**0.4045**
	NSF	0.2636	0.3267	0.2982	0.3316	**0.3510**
100	*APS*	0.0631	0.0999	0.0830	0.3206	**0.4045**
	NSF	0.2636	0.3267	0.2982	0.3316	**0.3510**



(a) Scholat dataset. (b) Gowalla dataset. (c) NSF dataset.

Fig. 6: The comparison results of Deepwalk, GCN and ScholarTeamFinder model on different datasets in terms of Recall@K.



(a) Scholat dataset. (b) Gowalla dataset. (c) NSF dataset.

Fig. 7: The comparison results of Deepwalk, GCN and ScholarTeamFinder model on different datasets in terms of Precision@K.

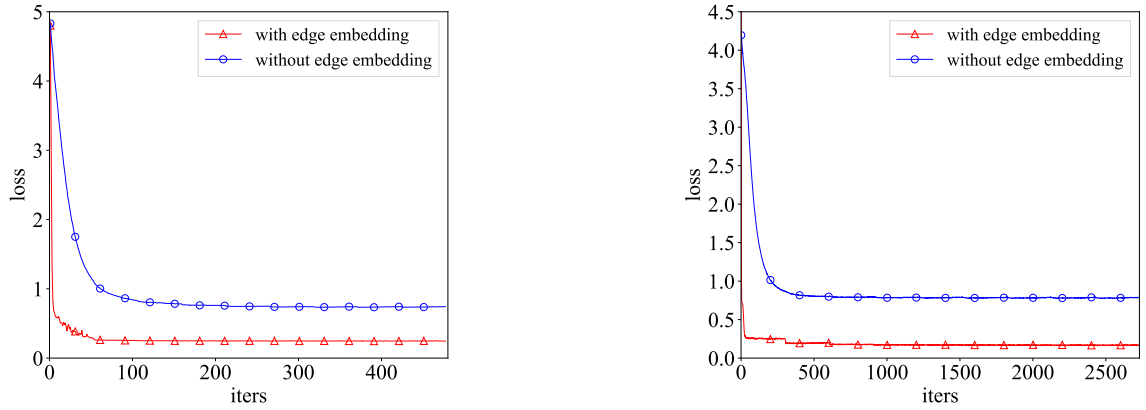
ath2vec, we add edge embedding into our ScholarTeamFinder model, and compare the loss values in the two cases of: with edge embedding and without edge embedding. Fig. 8 shows the experiment results using tests on the two datasets, which clearly prove that edge embedding can further improve our model's performance.

D. Parameter Sensitivity Analysis

We also investigate the influence of embedding dimensions in our ScholarTeamFinder. For this, we change the dimension of embedding to compare the performance. The results we obtained in this regard are shown in Table III. We observe that, with the increase of the dimension, the NDCG@K scores are boosted. Two of datasets exhibit the same trend, however when the dimension of embedding is increased to more than 64, the NDGC@K scores start to be stable. Further, they decrease with the dimension for increasing values.

E. Application Case Study

To intuitively demonstrate the efficiency of our ScholarTeamFinder model, we pick up one scholar as a case from the actual world to evaluate it. In our dataset [32], we assign one unique scholar id for every scholar. Hence, in this case study, we use the scholar id to represent the scholars to protect the fairness and privacy. This example shows the actual links from the NSF datasets, and also shows the prediction results



(a) APS dataset.

(b) NSF dataset.

Fig. 8: ScholarTeamFinder models with and without edge embedding comparison using loss values on different datasets.

TABLE III: Results of our proposed ScholarTeamFinder model for different dimensions of embedding for different datasets and in terms of NDCG@K.

NDCG	Datasets	*16*	*32*	*64*	*128*	*256*
10	*APS*	0.4025	0.4031	**0.4045**	0.4044	0.4040
	NSF	0.3427	0.3460	**0.3510**	0.3440	0.3453
20	*APS*	0.4026	0.4031	**0.4045**	0.4044	0.4040
	NSF	0.3428	0.3461	**0.3510**	0.3440	0.3453
30	*APS*	0.4027	0.4031	**0.4045**	0.4045	0.4040
	NSF	0.3428	0.3461	**0.3510**	0.3440	0.3453
40	*APS*	0.4027	0.4031	**0.4045**	0.4045	0.4040
	NSF	0.3428	0.3461	**0.3510**	0.3440	0.3453
50	*APS*	0.4027	0.4031	**0.4045**	0.4045	0.4040
	NSF	0.3428	0.3461	**0.3510**	0.3440	0.3453

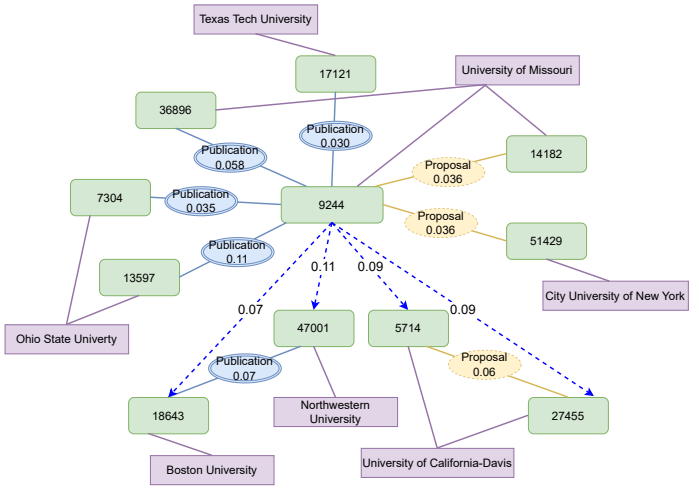


Fig. 9: Exemplar visualization of ScholarTeamFinder model output showing potential collaborators for a given scholar.

from the model, which is shown in Fig. 9, in which the green rectangles represent scholars, the purple rectangles indicate institutions, and the blue and yellow edges between the scholars have attributes and the probability. The edges' attributes include two kinds, *proposals* and *publications*. This implies that two scholars work on one proposal or one publication, so that they have links between them. The probability of the edges reveal the cosine similarity score for two or more scholars.

From Fig. 9, we observe that the center scholar (we set this scholar as the first scholar, the content in this figure all come from this scholar's links) is associated with other scholars with different probabilities in terms of working on the same publications or proposals. These already existing links in the knowledge graph are used as positive example to train the model. Then the model can predict whether there is a possibility of the collaboration between two scholars, which are shown in Fig. 9 with the blue doted line paths. Through manually checking the research interests, their researches can do some collaborations from different perspectives. Thus, this example shows how our ScholarTeamFinder model works as discussed in Section IV, and proves its efficient. Following the first hop prediction results, we can find the scholars (the second hop prediction results) who have collaboration with the first scholar predicted e.g., scholar *18643* and scholar *27455* in Fig. 9. Through computing the probabilities of the center scholar with the the second hop prediction results, they do have possible links. According to this, finding the existed research team with ScholarTeamFinder can be proved.

Additionally, it is noteworthy that scholars from the same institution may have increased potential for collaboration, as evidenced by scholars *36896* and *9244*. Institution information is therefore an important feature for future recommendations. Additionally, co-author links yield higher similarity scores than co-PI links, potentially due to the specificity of publication data compared to grant proposal scopes. Our model assists scholars in expanding their academic connections to discover potential collaborators from other institutions. For example, in Fig. 9, scholar *9244* can connect with scholars *47001* and *18643* from Northwestern University and Boston University, respectively. This case verifies that our model can help scholars to find potential collaborators from same/other institutions and help them get touch with new areas of experts within a scholar team.

VI. Conclusion

In this paper, we proposed a novel ScholarTeamFinder model to recommend multiple-disciplinary scholars to help them build scholar teams. This model mainly includes two parts, knowledge graph representation learning and the link prediction. For knowledge graph representation learning, we enhance the *metapath2vec* model through addition of semantic features to embed our knowledge graph's nodes and jointly train with edge embeddings, thereby extracting additional features from the knowledge graph. Subsequently, we compute node similarity for link prediction using the dataset combined with NSF grant awards data and Google Scholar publication data. From the performance comparison of the ScholarTeamFinder model with state-of-the-art models (*deepwalk*, *node2vec*, *GCN* and *metapath2vec*), the experiment results show that our model has better performance. Further, the results also prove that our method is efficient in recommending scholars. At the same time, the results also show that the embedding method can extract the nodes and links information efficiently, and it can improve the model performance. We also proposed a beam search based algorithm for finding a scholar team based on our model and research interests of identified scholars. In addition, we demonstrated using a case study to show how the ScholarTeamFinder model can work effectively and efficiently in other real-world scenarios as well.

Future work could expand the knowledge graph with additional scholar features and links, and also integrate user query processing to obtain targeted recommendations with high usability.

References

[1] F. Xia, W. Wang, T. M. Bekele, and H. Liu, "Big scholarly data: A survey," *IEEE Transactions on Big Data*, vol. 3, no. 1, pp. 18–35, 2017.

[2] Y. Xu, D. Zhou, and J. Ma, "Scholar-friend recommendation in online academic communities: an approach based on heterogeneous network," *Decision Support Systems*, vol. 119, pp. 1–13, 2019.

[3] H. Jin, P. Zhang, H. Dong, M. Shao, and Y. Zhu, "Personalized scholar recommendation based on multi-dimensional features," *Applied Sciences*, vol. 11, no. 18, p. 8664, 2021.

[4] C. Yuan, Y. He, R. Lin, and Y. Tang, "Graph embedding for scholar recommendation in academic social networks," *Frontiers in Physics*, p. 659, 2021.

[5] F. Wang, J. Dong, W. Lu, and S. Xu, "Collaboration prediction based on multilayer all-author tripartite citation networks: A case study of gene editing," *Journal of Informetrics*, vol. 17, no. 1, p. 101374, 2023.

[6] "Nsf awards info," https://www.nsf.gov/awardsearch, accessed: 2019-10-15.

[7] "Aps dataset," https://www.aps.org/index.cfm, accessed: 2022-09-26.

[8] "Scholat dataset," https://www.scholat.com/research/opendata/, accessed: 2022-09-26.

[9] "Gowalla dataset," https://snap.stanford.edu/data/loc-gowalla.html, accessed: 2023-05-07.

[10] X. Kong, Y. Shi, S. Yu, J. Liu, and F. Xia, "Academic social networks: Modeling, analysis, mining and applications," *Journal of Network and Computer Applications*, vol. 132, pp. 86–103, 2019.

[11] H. Wan, Y. Zhang, J. Zhang, and J. Tang, "Aminer: Search and mining of academic social networks," *Data Intelligence*, vol. 1, no. 1, pp. 58–76, 2019.

[12] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: extraction and mining of academic social networks," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008, pp. 990–998.

[13] C. Porcel, A. Ching-López, G. Lefranc, V. Loia, and E. Herrera-Viedma, "Sharing notes: An academic social network based on a personalized fuzzy linguistic recommender system," *Engineering Applications of Artificial Intelligence*, vol. 75, pp. 1–10, 2018.

[14] V. A. Rohani, Z. M. Kasirun, and K. Ratnavelu, "An enhanced content-based recommender system for academic social networks," in *2014 IEEE Fourth International Conference on Big Data and Cloud Computing*. IEEE, 2014, pp. 424–431.

[15] R. Zheng, L. Qu, B. Cui, Y. Shi, and H. Yin, "Automl for deep recommender systems: A survey," *ACM Transactions on Information Systems*, 2023.

[16] S. Zhao, R. Peng, M. Zhang, and L. Tan, "Heterorwr: a novel algorithm for top-k co-author recommendation with fusion of citation networks," *IEICE Transactions on Information and Systems*, vol. 103, no. 1, pp. 71–84, 2020.

[17] S. Boussaadi, H. Aliane, O. Abdeldjalil, D. Houari, and M. Djoumagh, "Recommender systems based on detection community in academic social network," in *2020 International Multi-Conference on:"Organization of Knowledge and Advanced Technologies"(OCTA)*. IEEE, 2020, pp. 1–7.

[18] G. R. Lopes, M. M. Moro, L. K. Wives, and J. P. M. De Oliveira, "Collaboration recommendation on academic social networks," in *Advances in Conceptual Modeling–Applications and Challenges: ER 2010 Workshops ACM-L, CMLSA, CMS, DE@ ER, FP-UML, SeCoGIS, WISM, Vancouver, BC, Canada, November 1-4, 2010. Proceedings 29*. Springer, 2010, pp. 190–199.

[19] J. Chen, Y. Tang, J. Li, C. Mao, and J. Xiao, "Community-based scholar recommendation modeling in academic social network sites," in *Web Information Systems Engineering–WISE 2013 Workshops: WISE 2013 International Workshops BigWebData, MBC, PCS, STeH, QUAT, SCEH, and STSC 2013, Nanjing, China, October 13-15, 2013, Revised Selected Papers 14*. Springer, 2014, pp. 325–334.

[20] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," *arXiv preprint arXiv:1709.05584*, 2017.

[21] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 701–710.

[22] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 1067–1077.

[23] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 855–864.

[24] S. Chang, W. Han, J. Tang, G.-J. Qi, C. C. Aggarwal, and T. S. Huang, "Heterogeneous network embedding via deep architectures," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 119–128.

[25] Y. Cen, X. Zou, J. Zhang, H. Yang, J. Zhou, and J. Tang, "Representation learning for attributed multiplex heterogeneous network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1358–1368.

[26] H. Liu, C. Zheng, D. Li, Z. Zhang, K. Lin, X. Shen, N. N. Xiong, and J. Wang, "Multi-perspective social recommendation method with graph representation learning," *Neurocomputing*, vol. 468, pp. 469–481, 2022.

[27] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 135–144.

[28] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[29] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[30] N. Kanakaris, N. Giarelis, I. Siachos, and N. Karacapilidis, "Shall i work with them? a knowledge graph-based approach for predicting future research collaborations," *Entropy*, vol. 23, no. 6, p. 664, 2021.

[31] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[32] X. Cheng, Y. Zhang, and H. Joshi, "Scholarteamfinder - nsf dataset," 2023. [Online]. Available: https://www.kaggle.com/dsv/6272506