11-8-2023

Geo-location informed Team Formation using GNN

Karan Saxena
University of Windsor

Geo-location informed Team Formation using GNN

By

Karan Saxena

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2023

Geo-location informed Team Formation using GNN

by

Karan Saxena

APPROVED BY:

M. Hassanzadeh
Department of Electrical and Computer Engineering

J. Lu
School of Computer Science

H. Fani, Advisor
School of Computer Science

October 11, 2023

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Establishing a competent team is crucial to the success of a project and is influenced by skill distribution and geographic proximity. A team not only benefits from the shared knowledge amongst the team members derived from geographic closeness but also affects the outcome of the project the team is assigned to perform. A team benefits by sharing resources among each member, collaborating efficiently on a given task, brainstorming on an idea more effectively and saving time and money for both the team members and the organization. This thesis uses a neural-based multi-label classifier after a spatial team formation that uses graph neural networks to transfer information from a heterogeneous collaboration network among experts. Our approach to maximizing the effectiveness of team composition considers the dynamic relationship between members' shared skill sets and geographic proximity to one another. Specifically, we build a heterogeneous network with the nodes being experts, skills, and places to represent the intricate connections between the specialized knowledge of experts and the regions in which they are present. We use graph neural networks to learn vector representations of skill profiles and geographic proximities using meta paths. Then, we follow that up with a feedforward neural model to recommend a ranked list of experts as a team. Following this pipeline allows us to maximize skill coverage while minimizing geographic dispersion, balancing effective collaboration and efficient communication among team members. We evaluate the accuracy of the recommended teams of experts concerning the requisite abilities and geographical distribution by utilizing classification and information retrieval measures. Our methodology was influential in building skilled and geographically coherent teams, as evidenced by experimental assessments of our suggested method on a real-world dataset of patents and computer science articles compared to baseline methods. We experiment our methodology on uspt and dblp with range of graph and neural architectures across different hyperparameters. The outcomes of this study contribute to the process of team creation by drawing attention to the advantages of using graph neural networks that consider both a person's skills and their location.

DEDICATION

I would like to dedicate this thesis to my mom and dad for their incredible love and support. Their utmost love and support throughout my career and their selfless commitment to giving me the future I always wanted have provided me with the necessary boost I needed in my life to achieve success.

Furthermore, I dedicate it to my brother Abhishek Saxena and sister-in-law Kirti Abhishek Saxena for being ever so lovable, kind, supportive, and the backbone of what I am today.

ACKNOWLEDGEMENTS

I would like to sincerely express my most profound gratitude towards my supervisor Dr.Hossein Fani, whose input helped me immensely. With his input, I was able to look at my research with a different perspective and a more critical eye.

Secondly, I would like to express my gratitude to my thesis committee members for their beneficial advice and suggestions for my thesis.

I humbly extend my thanks to the School of Computer Science and all concerned people who helped me in this regard.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

fnn	feed-forward neural network
bnn	bayesian neural network
vbnn	variational bayesian neural network
MST	Minimum Spanning Tree
emb	embedding
rnn	recurrent neural network
uspt	US patents
gith	GitHub
aucroc	area under curve of receiver characteristic operator
gnn	graph neural network
fnn	feed forward neural network
pr	precision
rec	recall
ndcg	normalized discounted cumulative gain
map	mean average prevision
#bs	number of bayesian samples
#nns	number of negative samples
OR	Operation Research

CHAPTER 1

Introduction

1.1 What is Team Formation and Why is it important?

Variety of factors comes into play when one seeks people suitable for a specific task. Individuals who usually react to such requests are candidates from all different fields. Sometimes, employers organize a team that has experts who are skilled in all tasks. However, it is not always possible. The need of an expert in one field is on the rise. A wide variety of employers are spending more on a diverse collection of candidates. Keeping task a priority, the longing for a close knit group of professional is higher. Having a group of professionals where everyone is close to each other gives benefits. A team formed with this thought in mind is successful and proves resourceful.

The art and science of team formation emerge as crucial threads in the complicated fabric of organizational dynamics and project management. At its root, team formation is defined as finding the right group of people that fulfil certain requirements of a task set by the organization and are fit in achieving success in the given task. But it is more than just putting together a collection of people; it is a strategic endeavour. It entails the laborious process of bringing together people from various backgrounds with talents, experiences, and viewpoints to work cooperatively toward a common goal. This approach goes beyond the simple act of grouping people based on availability or rank; it is about curating a team where members complement each other, ensuring that the team's collective capabilities not only meet but exceed the sum of individual contributions. In many ways, effective team formation is critical. To begin

with, it ensures skill complementarity. Any complex project requires diverse skills, ranging from analytical and technical to creative and interpersonal. A well-organized team ensures that all these skills are available and can be employed effectively, ensuring task efficiency and precision. Furthermore, diverse teams built through purposeful formation processes function as innovation incubators. When people with various experiences and expertise get together, they bring various perspectives. When properly directed, variety may lead to out-of-the-box thinking, fostering innovation, and yielding new ideas that a more homogeneous company might overlook. Productivity rises dramatically in well-formed teams. With clear roles, complementing strengths, and mutual respect, team members can operate smoothly, reducing redundancies and achieving goals in a simplified manner.Furthermore, the sense of belonging and purpose from being part of a cohesive team boosts morale and job satisfaction, increasing productivity and minimizing turnover. In today's rapidly changing global landscape, organizations face increasingly challenging difficulties. These various issues demand multidisciplinary and holistic solutions. Building strong, cohesive, and capable teams becomes crucial in such a situation. It is no longer regarded as a luxury but a necessity for ensuring organizational success in a competitive environment. This current computing era has brought some new problems to team creation. While some firms claim that this has enabled them to empower their staff to work from anywhere and anytime, this has ignored the requirement for group members to work together to achieve a goal. Working with experts in close proximity benefits the firm in terms of saving resources per team. It allows members to share collective knowledge and brainstorm ideas more effectively than working virtually. This may have been shown historically when the ruler used to ask his ministers to operate from their different offices rather than their quarters throughout the reign of kings and kingdoms. The urge to establish this in every walk of life where a team is crucial for a project, has opened up new paths of research in team building, where the need for having a physically cohesive team consisting of experts who not only have the essential skills but have also been proven to generate credible and successful work has been identified.

1.2 How has it been done?

The formation of teams has seen significant change throughout history, with changes brought about by societal norms, technological advances, and organizational structure alterations. The challenge of putting up effective teams has existed since the beginning of time. Throughout history, the first teams comprised members of the same family or tribe. Lineage was used to determine the formation of teams or groups, and age, gender, and experience were considered for assigning tasks and duties. Craft guilds were extremely important to the process of putting together teams during the Middle Ages. Teams of journeymen and apprentices would work under the direction of master artisans leading the teams. The formation was hierarchical, and the jobs were determined by a person's experience level and talent. Apprentices worked under the direction of masters, intending to work their way through the ranks. A paradigm shift occurred during the time of the Industrial Revolution. Around the various pieces of machinery and industrial lines, teams were organized. The activities that needed to be completed by each worker led to the forming of distinct groups of workers. Supervisors or managers were in charge of supervising the teams. With the rise of corporate culture, teams started to be formed around functions or departments, such as sales, marketing, HR, and finance. Hierarchical structures were prevalent, with teams often formed top-down based on the company's needs and strategies. As organizations recognized the need for cross-functional collaboration, project-based teams became more common. Such teams were temporary, formed to achieve specific project goals. Team members were chosen based on the unique skills required for the project, often pulling from different departments. Advances in communication technology enabled team formation across geographies. Global teams and outsourcing became popular, especially in the tech and service sectors. Teams were often formed based on skill sets, time zones, or client needs. Team formation's specifics have varied greatly by region, culture, industry, and individual organizational philosophies. The common thread, however, is the evolving understanding of the importance of team dynamics and the quest for optimal productivity and innovation through effective team formation.

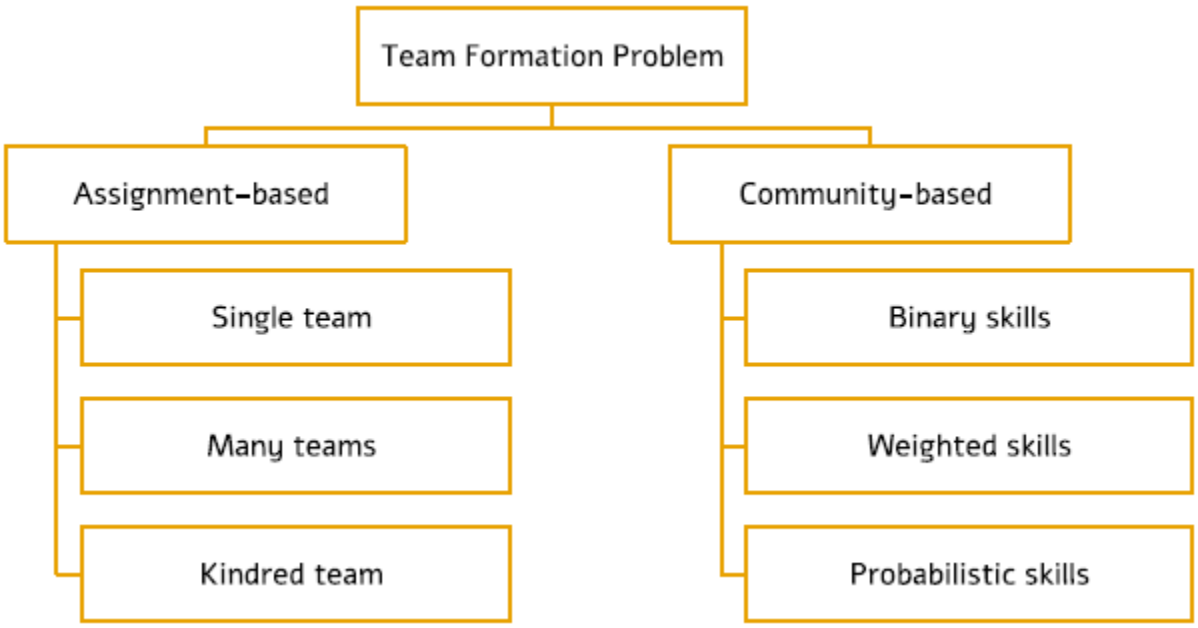

Fig. 1.3.1: Taxonomy of TFP[19]

1.3 How we plan to do it?

The Team Formation Problem or the TFP may be informally defined as finding the team members, out of a group of skillful experts, that would form a team of maximum "effectiveness" with minimum "expenditure" to undertake a specific task. A taxonomy[19] on the tfp breaks down the problem into two broader categories: Assignment based and Community based. The vast nature of the tfp makes it even more difficult to capture all the varying factors into one algorithm. We take in the approach of involving physical location of experts in accordance with experts' skills to recommend the most probable set of experts for a given task. In this thesis, we address the spatial team formation problem; that is, given a set of experts, skills and locations, which includes experts' geographical location in terms of country, province, or city, the goal is to find the optimal team whose success is almost surely guaranteed. Specifically, we aim to figure out whether the combination of skills and locations in team formation has synergistic effects. Majority of existing team formation methods address the problem of team formation by using skills as a primary factor [5], [43], [7] but overlooking geographical location and the corresponding ties it leads to between

experts within a team. Although remote work over online platforms has facilitated today's globalized work environment, geographical proximity remains important for face-to-face interactions, cultural understanding, time zone differences, and access to local resources such as availability of certain region-locked services by companies, availability of cloud servers where the segregation of services is based on physical location of employees [40], which can impact team dynamics, coordination, and effectiveness [48]. Organizations can strive to create skill-driven and geographically cohesive teams by considering both skills and locations in team formation. For example, forming team of experts from different time zones, e.g., from gmt and est time zones, where the business hours/days of one expert are non-working/resting periods for another expert, heavily discounts the efficiency of communication and accrue more costs associated with time, effort, and resources.

Despite its importance, including geographical location as a criterion in conjunction with skill in team formation literature, little work has considered geographical proximity when recommending experts for a team. Selvarajah et al. [40]is one such work that considered the geolocation distance between pairs of experts in a weighted homogeneous graph as communication costs as well as other factors like experts' proficiency level followed by a search for an optimum subgraph using a multi-objective optimization function. Their work, however, falls short for geolocations that are geographically close yet lawfully separated like cities situated at a country's or federal province's border. Their work overlooked a geolocation's direct relation to experts and their complementary skills. Furthemore, subgraph optimization has been proved NP-hard, hence computationally prohibitive for large-scale expert networks and [40] had to use heuristics by cultural algorithms, a class of evolutionary algorithms inspired by social learning in society. Meanwhile, recently a paradigm shift to a machine learning-based approach has been observed due to technological improvements in computing systems and methodological advances in graph neural network (gnn) techniques [46], opening doors to the analysis of massive graph-structured data coming from different fields. Graph neural network has provided an effective yet efficient way to solve the graph analytics problem by converting a graph into a low dimensional space while pre-

serving the graph information and has shown expressive performance for a vast array of AI-hard problems such as natural language processing [44], knowledge graph [47], recommender systems [45], and computer vision, among others. Hence, its application in team formation received attention from a few works, particularly in incorporating geolocation. Among the first, Rad et al. [34] proposed forming skill-based teams of experts using a feedforward neural model to map the vector representation of required skills in the input layer onto a Boolean occurrence vector of experts in the output layer to recommend members of a team. To learn the vector representation of skills, Rad et al. formed a heterogenous graph whose nodes were skills, experts, and geolocations, and applied Dang et al.'s metapath2vec [14] to learn the vector representations of skills in the context of locations of the teams. Improving upon Rad et al.'s work, Sagar et al [23] employed deep graph infomax, a graph convolution network [15] with attention layer as an encoder to generate vector representations of skill. As opposed to random walk in metapath2vec, deep graph infomax uses mutual information that relies on contrastive training from the original graph for positive samples and from the noise-added (corrupted) graph for negative samples. Sagar et al.'s work yielded more effective vector representations in fewer training epochs owing to convolutional architecture and contrastive learning procedure. Such gnn-based works, however, disregard the vector representations for geolocations when training their feedforward neural model to learn recommending optimum teams. We aim to take a step forward and build upon existing gnn-based work by leveraging the vector representations for geolocations directly in the input layer of the feedforward neural model next to the vector representation of skills. As shown in Figure 1, we form a heterogeneous collaborative graph (network) whose nodes are experts, skills and locations connected through edges based on the training instances of teams. We utilize metapath2vec to embed skills and geolocations into dense low-dimensional embeddings using predefined metapaths. A metapath is a random walk taken by a graph neural network method from a source node to a destination node based on predefined eligible sequences of different node types. Finally, we concatenated and fed the learnt skills and locations' embeddings into a Bayesian neural network to predict a

ranked list of experts whose top-k most probable experts form the optimum team. We benchmarked our proposed method against state-of-the-art baselines on two large-scale datasets of US patents (uspt)[usp] and computer science research publications (dblp)[dbl]. Our results show that:

- When the distribution of teams over geolocations is taken into account with skills for predicting experts for a team, the resultant teams end up having more optimal experts than just considering skills alone.
- Random walks with no predefined metapaths on a heterogeneous graph yield better results compared to the metapath walks.
- Considering geolocations does gives an overall improvement in the reoprted metrics in most of the experiments performed.
- Granularity of the location also plays a major role in swaying the results in either direction when the location coverage is decreased to a much more smaller level such as cities.
- Inclusion of teams as graph nodes does not improve the performance of the reported metrics and the graph without the team nodes have better reported results.
- We demonstrate the effect of negative sampling heuristics for neural team formation on a host of information retrieval and classification metrics such as map, ndcg, as well as precision, recall, and rocauc.

To support the reproducibility of work, we publicly release the codebase and running settings at https://github.com/fani-lab/OpeNTF/tree/geo.

CHAPTER 2

Related Works

The dynamics of team formation have been studied in different approaches and could be divided into two broad categories: search-based and learning-based methods. Search-based methods optimize every step via integer programming to find the optimal team given some constraints. These methods take factors such as time constraints, personnel, and communications cost, spatial properties of a candidate, and personal preferences. Although these factors offer a reliable foundation for forming a team from scratch, they need to establish the condition where a task requires a few additional members to complete its assigned duties rather than a whole new team. In recent years, the team formation problem has gained significant attention among researchers in computer science or, more specifically, in information retrieval as researchers and practitioners recognize the crucial role of well-constructed teams in effectively addressing complex information retrieval challenges. A project's success or failure dramatically depends upon the dynamics of a team assigned to perform the tasks; coherence between team members, skills, interpersonal relationships and ability to work in a group primarily affect the outcome. Hence, a significant focus has been given to team formation processes in fields such as academia [42], industry [8] and healthcare sector [10]. A survey by Julio Ju et al. [19], showcases the growing importance of team formation among researchers. The authors show the problem's variability and complex nature and display the problem's NP-hard nature. This means that it is unlikely for any method to produce optimal solutions in a reasonable amount of time. Instead, the approach usually focuses on developing algorithms that deliver approximate solutions.

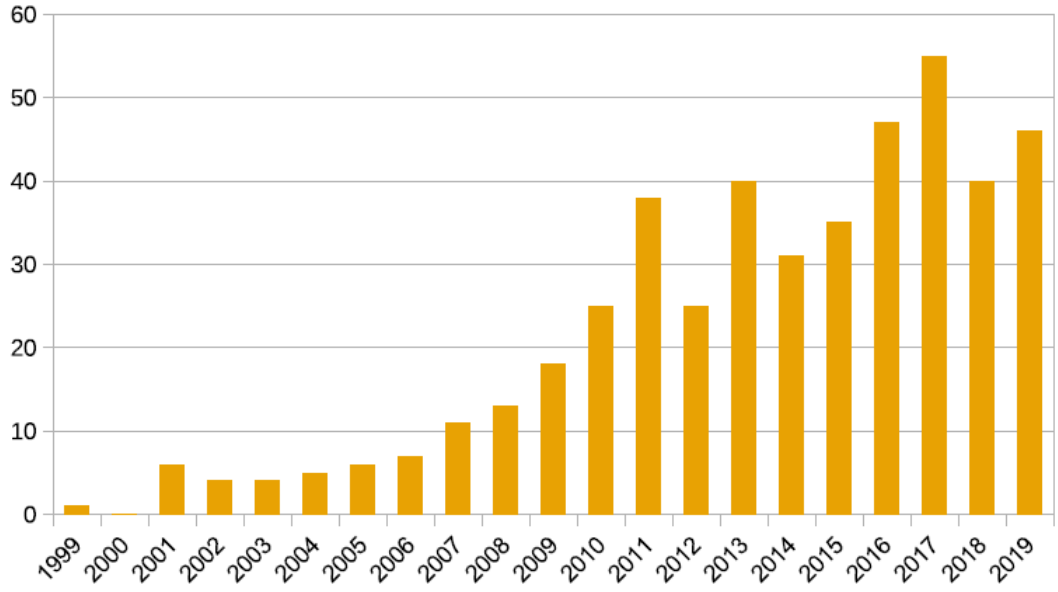

Fig. 2.0.1: Histogram of estimated number of TFP publications per year, for the past 20 years.[19]

The author further proposed a taxonomy structure of the problem that divides the team formation problem(tfp) into two constituents: assignment-based and community-based. The assignment-based team formation is further divided into single teams, many teams, and kindred teams approaches. At the same time, the community-based section is split into binary skills, weighted skills and probabilistic team formation. The research approach for several solutions for the tfp problem falls under one of these subsections. The growing interest in the team formation domain propels the researchers to look for newer approaches, apply varying algorithms and techniques to address the tfp problem and provide a solution that best assesses the problem and produces the outcome as optimal as possible for the given use case. The research on the team formation problem could be dated back to 1998 when Zzkarian et al. [48] first introduced the problem to the masses. Zzkarian et al. [48] defines teams as "a distinguishable set of two or more people who interact, dynamically, independently, and adaptively toward a common and valued goal/objective/mission, who have each been assigned specific roles or functions to perform, and who have limited lifespan of membership". The authors utilize a quality function deployment (qfd) and the analytical hierarchy process (ahp) approach and formulate the problem as an

integer programming model. Qfd is a systematic approach to product development that translates customer requirements into relevant product design characteristics. It identifies the general requirements a new product must satisfy to ensure customer preference. In team formation, the qfd methodology is applied to collect and represent data for the multi-functional team selection model. AHP is a multicriteria decision-making method that uses hierarchical structures to represent decision problems and develop alternative priorities. A significant issue regarding utilizing the ahp approach in the team selection model is the considerable effort needed to complete pairwise comparisons in extensive hierarchies. The eigenvalue approach employed in the AHP requires a substantial number of comparisons, specifically (n(n-1))/2, to determine the priorities of n elements. This can pose computational challenges when dealing with a significant value of n, making the pairwise comparison process computationally demanding. The setup for the approach also gives great importance to the decision-maker making perfectly consistent judgments for the comparison process requires a great deal of subjective knowledge, which in many cases is not achieved, leaving the problem to have an almost mid-level pairwise comparison. Hence, the authors do a great job in introducing the tfp and providing an analytical approach to solving it, leaving the door open for further researchers to address these limitations and offer different solutions to the problem. [32] states that virtual teams face challenges related to geographical distance, temporal distance, perceived distance, the configuration of dispersed teams, and the diversity of workers. Physical factors such as geographic, temporal, and perceived distance impact virtual teams. These factors are tightly coupled with social and emotional factors, including trust, motivation, and conflicts. [11] studies the collaboration rate of experts in the research domain. Some of the problems of having distributed research projects, which the authors outline, are that these research projects often have poor outcomes and coordination mechanisms fail to address these problems. Coordination costs and effort required to sustain strong working relationships are higher in distributed projects. The author points out that collaborative tie strength is influenced by distance, interdisciplinarity, and prior experience. Distance and interdisciplinarity reduce tie strength, while prior experi-

ence increases it. In research projects, distributed project members are likely to use different tools and technologies with different members of their research project rather than a common suite of software for the entire project, hence increasing the project's overall cost. Virtual organizations involving large-scale cooperative work across different institutions can effectively bring together diverse sources of expertise. Still, the distribution of knowledge and technology use can present barriers to open exchange and discourse. These reasons compel the need to understand and research on how incorporating physical location in different domains is going to help an organization form an optimal team and increase the success rate of a project's outcome. The rest of this literature review breaks down the available research in tfp into two subsections: search-based and learning-based approaches. This literature review discusses each of them and the work available for them in detail.

2.0.1 Search-based Methods

Search-based methods form a prominent approach in team formation, leveraging optimization techniques to address the challenges of creating effective teams. These methods involve systematically exploring the space of possible team compositions to find optimal or near-optimal solutions based on predefined criteria or objectives. Employing optimization algorithms, search-based methods aim to identify team compositions that maximize team performance, enhance diversity, ensure expertise coverage, and minimize conflicts or skill gaps. Search-based methods offer a systematic and algorithmic approach to team formation, enabling researchers and practitioners to navigate the vast solution space and identify promising team compositions. These methods provide a means to tackle the combinatorial nature of team formation problems, where the number of possible team configurations grows exponentially with the size of the candidate pool. By applying optimization algorithms, search-based methods can efficiently explore the solution space, evaluating different combinations of team members based on various criteria and objectives

The choice of optimization algorithms plays a crucial role in the effectiveness of search-based methods for team formation. Genetic algorithms, ant colony opti-

mization, particle swarm optimization, and other metaheuristic techniques have been widely employed in this context. These algorithms use iterative search processes that mimic natural evolution, collective behaviour, or different intelligent strategies to explore the solution space efficiently and converge toward optimal or near-optimal solutions. Work by [4] proposes a unique way of quantifying collective intelligence(ci). The authors make this work by combining three different factors into ci. These are the experts' knowledge competence, which measures team members' relation with other experts, a time-decayed trust measure, which gives more weightage to the experts who collaborated recently, and a trust propagation technique that deals with the sparsity of the tfp. The collective information of the above three factors is then subjected to a genetic algorithm-based optimization model for team formation to maximize the proposed quantification of collective intelligence. This method, although good in theory, gives substantial importance to more recent collaborations among experts for forming a probable team but assumes a vital piece of information, which is the success or the favourable outcome of the experts from the most recent contribution, is surely guaranteed which might not always be accurate. The authors themselves also define the parameters for their ci function and leave no decision on the subject matter experts that might be using their approach for team formation. In [26], given some constraints, the author starts with a large group of individuals to make a perfect team. The authors state that the problem with these constraints is prevalent in many fields of work where organizations struggle to find the best-suited subset of individuals for a particular task. The authors tackle this problem with the introduction of communication costs among experts. They define the communication cost as an attribute of the edges in a social network or a graph between nodes of experts. Communication cost is the weight between two nodes(experts), which can be defined in multiple ways depending on the domain to which the tfp is applied. For instance, in academia, the communication cost could be defined as the number of collaborations between two authors. A low-weight edge between nodes implies that the nodes can communicate or collaborate more quickly than the nodes with a higher edge weight between them. Defining an appropriate communication cost

function for a social network, the authors aim to find a team whose cumulative cost is lower than other probable experts. The lower the communication cost, the better the quality of the team. The authors propose two algorithms to address this specific version of tfp for diameter-tf and mst-tf problems. Out of the proposed algorithms to address the problems above, the authors propose rarestfirst algorithm for diameter-tf problem and coversteiner and steinertree algorithms for the mst-tf problem. The authors further state that the coversteiner algorithm completely ignores the network's underlying graph structure, leading to ignoring the complexities of the graph. To address this author proposes enhancedsteiner algorithm. The authors test their proposed method on dblp dataset only and lack further strengthening of their work by not testing on other datasets. The authors do not explain the rationale behind selecting the communication cost. However, they do provide two alternative approaches to calculating the cost via the graphs. This does not justify a team's cost or how they calculate the minimum cost required by the section to function optimally. The authors do not provide any substantial quantitative evidence that the teams chosen by their algorithms are the most optimal. The paper [6] solves the team formation problem with a metaheuristic approach simulated annealing. While the authors does take into account a good number of parameters for their optimization function, the simulated annealing approach remains ineffective of incorporating a change in the proposed methodology, leading to a problem of scalability. While these methods could be computationally extensive, this provides another reason why the authors did not consider more constraints during their research. The authors takes numerous assumptions into consideration such as considering interpersonal skills, conflict resolution, effective decision making, etc which are subjective to each project and requires attentive modifcication based on the project's requirement. The authors' proposed methodology also considers formation of new teams for a project and does not address the scenarios where the teams are already present but only require few of the expert/s to complete a specific task. The authors also does not take into account the cohesive nature of location and skills which leaves an important gap in the formulation of a solution for this problem. aims to propose a framework for analyzing and selecting

project managers and team members based on their knowledge and social network. In [3] proposes a general framework and algorithms for tfp, based on two optimization goals: balancing the workload among people and minimizing the coordination cost of each team. The paper also provides theoretical and experimental results that show the effectiveness of the proposed approach. The authors use a bi-criteria optimization technique that bounds the coordination cost for each task and minimizes the maximum load of a person while considering different measures of coordination cost, such as the Steiner tree, the diameter, and the sum of distances, and different models of team connectivity, such as implicitly or explicitly connected teams. A suitable allocation cost function is defined that depends on the individual loads of the people and solves a subproblem of social task assignment for each incoming task. The approach applied here assumes that the skills and compatibility of people are known and fixed, which may not be the case in practice. Over time, experts can adapt and learn new skills, which this method fails to consider. Works of Selvarajah et al [40] addresses the geographical proximity when recommending experts for a team. The authors consider the geolocation of experts as a distance between two experts and interpret it as a parameter for reducing communication costs. We use an example to explain the flaw not addressed in this paper: Consider a city c situated at a border location very close to another country or city. An organization in the city c wanting to find experts, despite having a distance from the cities on the other side of the border, cannot hire those experts due to restrictions in place because of geographical laws. A more defined example of this could be a healthcare insurance provider is bound to hire experts even from far-off places in its own country rather than going for a location that is close to its in terms of distance but is in another country due to healthcare laws not allowing the data to be transferred to another country. The paper also does not address a location's direct relation to experts and their complementary skills. The authors of [22] describe the tfp as finding an affordable and collaborative team in an expert network subject to cost constraints (communication cost and personnel cost). The communication cost between two experts can be defined according to the application's need. The authors state that two types of costs are generally associated

with team communication and personnel costs. Many recent methods overlook this condition and only consider either one of the two costs. However, other costs, such as overhead costs of forming a team that doesn't belong in the same physical location and infrastructure set-up costs, are frequently overlooked, which can be sought after by incorporating location as a constraint. Also, The work only focuses on one aspect of team formation: forming the team from scratch and not diving deep into the other aspects, such as what if the team already exists and you only want to substitute one player. The work by Kargar et al. [21] which targets the problem of team formation by utilizing two costs as constraints, communication cost and personnel cost, which the above work sidelines from its objective function. The authors propose four algorithms that provide different trade-offs between solution quality and computational efficiency. The approximation algorithm provides a performance guarantee with an approximation ratio of 2, meaning that the solution returned by the algorithm is guaranteed to be at most twice as large as the optimal solution. However, it may not always return the best solution in practice. The iterative replace algorithm is a simple and intuitive heuristic that can quickly find a good solution. Still, it may get stuck in local optima and not find the global optimal solution. The minimal cost contribution algorithm and the mcc-rare algorithm are more sophisticated heuristics that consider both personnel and communication costs when selecting experts to add to the team. They often find better solutions than the iterative replacement algorithm, but they may require more computational time. Also, depending solely on the rarest skill could form teams with unbalanced skill distribution. This might result in incomplete or inefficient project execution due to missing complementary skills. Another work by Selvarajh et al.[37] proposes a weighted structural clustering algorithm called wscan to solve the tfp in social networks. It is an enhanced variant of the Structural Clustering Algorithm for Networks (scan) and is used to detect clusters, hubs, and outliers in networks. The algorithm starts by finding a pool of experts with the required skills and then searches for a highly connected (core) expert among all the experts in the network. The cluster is then expanded from the core to neighbourhood nodes within a threshold range of communication cost. The goal is to

identify experts while minimizing communication costs for the project with specific skills. The paper also introduces the concept of collective expertise, a phenomenon of a certain level of expertise occurring among a group of individuals possessing a set of skills necessary to complete a task as a team. The paper acknowledges the limiting performance of the wscan algorithm compared to other genetic algorithms; however, the WSCAN run-time was better than the other algorithms. Also, the limited testing of the proposed algorithm leaves doubt on how the algorithm might perform on vast real-scale data. Selvarajah et al. [37] also works by generalizes the tfp in palliative care and argues that the limited work in tfp for palliative care has motivated the need for research in this field. An agent-based model was proposed to improve the quality of service in palliative care. The model aimed to find a group of suitable care providers to satisfy the requirements of patients, considering contact costs and resource limitations. This model showed a reduction in operational costs and improvements in the quality of service. An agent-based architecture was proposed to facilitate communication and collaboration among patients and care providers in palliative care. Multi-agent systems and Information and Communication Technologies were used to improve clinical data management for palliative care patients. However, the evaluation of the proposed model was done on synthetic networks, and it may not fully capture the complexities and nuances of real-world palliative care systems and other important factors such as patient preferences, cultural considerations, or individual care provider expertise. The cultural algorithm is also susceptible to its sensitivity to parameter settings and the potential for premature convergence, which are not explicitly addressed in the paper. The most recent work by Selvarajh et al. in [38], closely follows the work by Lappas et al. [26]. The paper proposes a knowledge-based evolutionary optimization algorithm to solve the problem of identifying a team of experts in a social network, considering their past collaboration and communication cost functions. The authors highlight the importance of past collaboration among team members, as it expedites project completion by leveraging existing familiarity and rapport. The paper intends to solve the tfp by introducing the concept of communication cost functions and proposing a method to optimize them while en-

suring the coverage of required skills and proposing a knowledge-based evolutionary optimization algorithm. The algorithm utilizes a cultural framework consisting of a population and belief space that co-evolve during optimization. The algorithm starts by producing a predefined number of random teams as the initial population, where each team is represented by an array structure with cells representing required skills filled by experts possessing those skills.

2.0.2 Learning-Based Methods

Recently, team formation has been approached by learning-based methods. For instance, Li et al. [29] developed a recommendation system based on collaborative filtering techniques to match individuals' skills and preferences. Sapienza et al. [36] utilized a deep neural autoencoder for team formation and introduced a computational framework to identify teammates who contribute to the growth of their peers. However, in scenarios where data is limited, such as in team formation, where only a few teams have successfully collaborated for a specific set of skills, autoencoder neural networks are susceptible to overfitting and are inefficient in capturing data uncertainty. Rad et al. [35] proposed a Variational Bayesian neural architecture to address these limitations. By incorporating a Variational Bayesian framework, the model can better handle uncertainty in the data and provide more robust assessments for team formation. However, their model was trained on published scholarly papers in computer science and lacks observing unsuccessful research (rejected papers). Works in graph neural networks, gnns [25] [27] [41] and in natural language processing [31] [33] proves that inclusion of negative sampling does help in improving the overall metrics but till date not much works have incorporated the use of location where both the positive and negative samples are considered.

2.0.3 Location Inclusive Methods

Geographic factors are crucial in team dynamics, particularly in distributed or remote work environments. Morrison-Smith et al. [32] examined the challenges of

dispersed teams and identified communication and coordination as critical factors for successful collaboration. Similarly, Cummings and Kiesler [11] explored the impact of proximity on forming social ties within teams. They discovered that close physical proximity fostered stronger relationships and higher levels of cooperation. These studies emphasize the need for practical tools and strategies to overcome the barriers imposed by geographical distance. However, despite the importance of location in team dynamics, integrating geographic considerations into team formation processes is still sparsely addressed. This research gap highlights the need to explore the interplay between skills and locations to create skill-diverse and geographically cohesive teams. More recent approaches utilize learning-based methods such as neural networks and graph neural network also addresses the team formation problem considering different constraints for a given team. Hamidi Rad, R et al. [35] employ learning-based neural methods to learn relations between experts and skills but do not incorporate the importance of geographical location in their approach. Dashti et al. [12]provide a convenient framework with negative sampling in its architecture. While these methods give substantial results for team formation, they do not consider location a constraint. Hamidi Rad R. et al. [34] takes a different approach from conventional neural networks and use graph neural network with meta-paths. However, given its importance, the cohesiveness of skills with location is yet to be answered and the lack of original work in this domain, it calls for a novel research to be done in tfp inlcusive of location.

CHAPTER 3

Problem Definition

In this section, we formally define the geo-location team formation problem. Given a set of experts with different skills and geographical locations, the problem of spatial team formation aims to optimize the composition of teams by considering the interplay between skill compatibility and geographic proximity. Formally,

Definition 1 (Team) *Let $S = \{i\}$, $L = \{j\}$, and $E = \{k\}$ be the set of skills, geolocations, and experts, respectively. A team of experts $\mathbf{e} \subseteq \mathcal{E}$ that collectively cover the skill set $\mathbf{s} \subseteq \mathcal{S}$ and are geographically located in $\mathbf{l} \subseteq \mathcal{L}$ is a triple represented by $(\mathbf{s}, \mathbf{l}, \mathbf{e})$ along with its success status $y \in \{0, 1\}$. Further,*

$$\mathcal{T} = \{(\mathbf{s}, \mathbf{l}, \mathbf{e})_y : y \in \{0, 1\}, \mathbf{s} \neq \emptyset, \mathbf{l} \neq \emptyset, \mathbf{e} \neq \emptyset\}$$

indexes all training instances of teams, both successful and unsuccessful.

Definition 2 (Team Formation) *Given a subset of skills $\mathbf{s}$, geolocations $\mathbf{l}$, and all teams $\mathcal{T}$, the GeoSpatial Team Formation problem aims at identifying an optimal subset of experts $\mathbf{e}^*$ such that their collaboration in the predicted team $(\mathbf{s}, \mathbf{l}, \mathbf{e}^*)$ is successful, i.e., $(\mathbf{s}, \mathbf{l}, \mathbf{e}^*)_{y=1}$, while avoiding a subset of experts $\mathbf{e}'$ resulting in $(\mathbf{s}, \mathbf{l}, \mathbf{e}')_{y=0}$. More concretely, the Spatial Team Formation problem is to find a mapping function f with parameters $\boldsymbol{\theta}$ from the powerset of skills and geolocations to the powerset of experts such that*

$$f_{\boldsymbol{\theta}} : \mathcal{P}(\mathcal{S}) \times \mathcal{P}(\mathcal{L}) \rightarrow \mathcal{P}(\mathcal{E}), \quad f_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{l}) = \mathbf{e}^*$$

.

CHAPTER 4

Methodology

In this chapter, we will explain the methods we used to tackle the problems introduced in the Problem Definition section. Our main goal is to estimate f for forming a team of experts that maximizes the overall skill coverage and minimizes the geographic dispersion among the subset of experts based on the training instances of successful teams satisfying the same property. We propose to learn f via a Bayesian feedforward neural network that maps dense vector representations of required subsets of skills $\mathbf{s}$ and geolocations $\mathbf{l}$, obtained from a graph neural network, onto a subset of experts $\mathbf{e}$ who can almost surely successfully accomplish the task in hand. Hence, our pipeline consists of three stages, (1) starting from creating a heterogenous graph, (2) followed by training a graph neural network to learn dense vector representations using meta-paths, and (3) transferring the learnt vectors to a feedforward neural network to learn f. We formally define each step in the following.

4.0.1 Team Graph Creation

We construct Team Graph G that captures the relationships between experts' skills and their geographic locations within the context of teams. Each expert, skill and location is represented as a node in the graph. At the same time, the edges capture the connections between a subset of skills s, a subset of experts e and the spatial proximity between locations l within the team (s, l, e); $\mathbf{s} \subseteq \mathcal{S}$, $\mathbf{l} \subseteq \mathcal{L}$, $\mathbf{e} \subseteq \mathcal{E}$. Formally, Definition 3.3.

Definition 3 (Teams Graph) *Teams Graph is a heterogeneous unweighted undirected graph* $G(\mathbf{V}, \mathbf{E})$ *where* $\mathbf{V}$ *is the set of its nodes including nodes of types skills* $\mathcal{S}$*,*

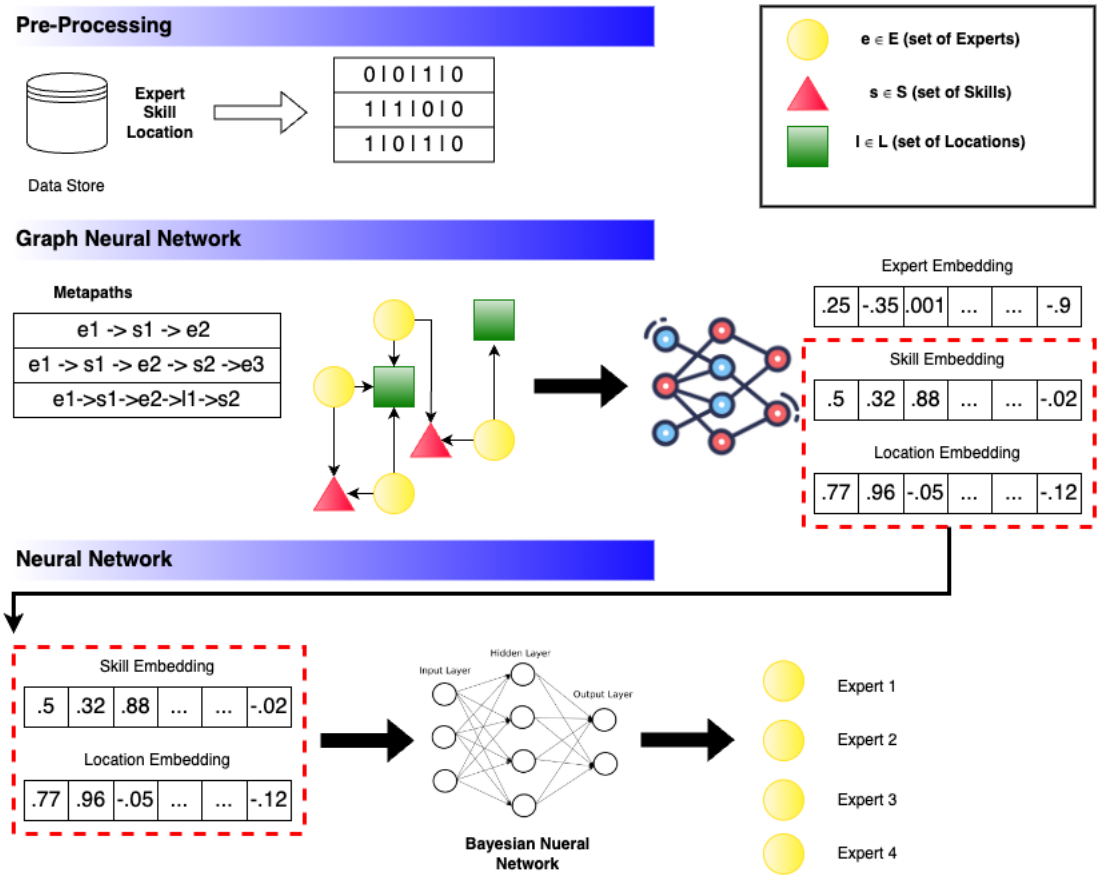

Fig. 4.0.1: Proposed Workflow Architecture

geolocations $\mathcal{L}$*, experts* $\mathcal{E}$*, and teams* $\mathcal{T}$*, i.e.,* $\mathbf{V} = \mathcal{S} \cup \mathcal{L} \cup \mathcal{E} \cup \mathcal{T}$*, and* $\mathbb{E}$ *is the edge set. Given a team* $(\boldsymbol{s}, \boldsymbol{l}, \boldsymbol{e}) \in \mathcal{T}$*, edges exist to connect the team's node to all the skills* $i \in \boldsymbol{s}$*, to all geolocations* $j \in \boldsymbol{l}$*, and to all experts* $k \in \boldsymbol{e}$*.*

To create Team Graph G, we map each team (s, l, e) onto an induced subgraph in G. As shown in Figure *, if an expert e1 has worked on a team t1 with skills s1 and s2 and belongs from a location l1, an edge here would be represented as e1-¿s1, e1-¿s2 and e1-¿l1. This graph models the interdependencies between skills, locations, and experts within the context of teams, forming the basis for recommending new teams. Depending on the underlying benchmark dataset, the skill subset might not be predefined by nature and should be inferred based on what makes intuitive sense. For instance, in dblp [dbl] collection of computer science research papers where each paper is considered as a team whose members e are the paper's authors, the skill

subset s have been inferred by the keywords in the paper's title [26], [28], [30], [20] or by the paper's field of study (fos). Furthermore, the skill subset s of the paper (i.e., all the keywords or fields of study in the paper) would be considered for all the paper's authors even though the authors of the paper may be skillful in a few, not all, of the skills in subset s. Also, the team's location subset l can be either the geographic locations of the authors' affiliations (e.g., universities), or the venue(s) where the paper has been submitted [35]. Another popular dataset is uspt[usp], which includes information about patents issued by the USPTO. To form the Teams Graph G based on the uspt dataset, each patent is considered as a team whose expert members e are the inventors, the patent's classes and subclasses can be the team's required skill subset s, and the geolocations of the inventors' living places can be the team's location subset l.

4.0.2 Vector Representation Learning

Once the Teams Graph G has been created, we utilize a graph neural network (gnn) to encode skills, locations, and experts' relationships and their high-order topological structures in the discrete space of the graph into low-dimensional (dense) vectors in a continuous vector space. A graph neural network can be formally defined at an abstract level as: Definition 3.4.

Definition 4 (Graph Neural Network) *Given a Teams Graph G, a graph neural network method is a mapping function g_φ parameterized by φ that learns the nodes' dense low-rank d-dimensional vector representations $v_u \in \mathcal{R}^d; \forall u \in \mathbb{V}$ with respect to a graph G for Spatial Team Formation task.*

Graph neural network is to address the challenges of traditional graph processing methods, esp., for large-scale graphs, including: (1) most of them are combinatorial computation steps, resulting in high computational complexity. For example, almost all graph-based team formation methods rely on the shortest or average path length between two nodes to represent their distance, which involves enumerating many possible paths between two nodes, (2) they defy parallelizability as the nodes in a

graph are coupled to each other explicitly by edges and distributing nodes in different shards or servers causes demandingly high communication cost among servers, and holds back speed-up ratio, (3) Inapplicability of machine learning methods. Recently, machine learning methods, especially deep learning, are very powerful in many areas. These methods provide standard, general and effective solutions to various problems. For traditionally represented network data, however, most off-the-shelf machine learning methods may not be applicable. Those methods usually assume that independent vectors can represent data samples in a vector space. In contrast, the samples in network data The gnns effectively capture the complex dependencies and interactions among experts' skills and locations. GNN embeddings enable extracting valuable features that capture skill-based profiles and geographic characteristics by propagating information through the graph structure. To construct our graph neural network pipeline, we utilize two graph layers [24], a dropout layer followed by a relu activation function. We concatenate the input nodes into a single node x with edges distinguishing between each node type. These nodes and edge indices are then fed into the graph layers to facilitate the training of node embeddings. The input and output to the graph can be represented in a more concrete mathematical notion as follows: -

$$\begin{aligned} \text{Input:} \quad & g(x = (e, s, l), \text{edges} = (\text{expert} \rightarrow \text{skill}, \text{expert} \rightarrow \text{loc})) \\ \text{Output:} \quad & g'(x' = (e', s', l')) \end{aligned}$$

Where x is a collection of nodes from each node type expert, skill, and location respectively, g' denotes the trained output of embeddings consisting of x'. The collection of x' contains trained embeddings of expert, skill, and location uniquely identified by their respective node ids.

We utilize our gnn module as a reproducible graph network and structure it in a way where an input of a graph with node features and edges between those nodes can be fed to obtain consecutive node embeddings. These embeddings can be utilized in several downstream tasks, such as node classification and link prediction. We extract the trained embeddings based upon their indices and restructure it to represent teams

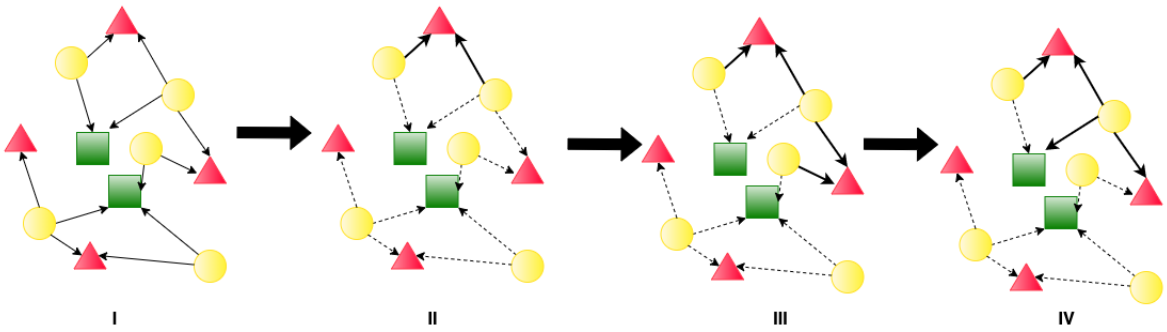

Fig. 4.0.2: Depiction of metapaths. (i) represents the original graph. (ii - iv) represents our choice of three metapaths and how graphs walk from one node to another based on those metapaths.

where each team is a collection of skills vector, expert vector and location vector extracted from the trained embeddings.

4.0.3 Integration of Metapaths

To incorporate a deeper connection of relationships in which a graph takes a predefined path between experts, skills and locations, we explore using metapaths within the graph representation. Metapaths [14] represent domain-specific paths connecting experts based on shared attributes or characteristics. By integrating metapaths, we aim to capture relations among nodes that a random walk-based approach cannot, enabling a more comprehensive understanding of team dynamics and potential synergies. Integrating metapaths can enhance the understanding of expertise interconnections beyond direct skill-based or geographic relationships in team formation using skills and locations. We incorporate three metapaths, such as a = [”experts”, ”skills”, ”experts”], b = [”experts”, ”skills”, ”experts”, ”skills”, ”experts”], c = [”experts”, ”skills”, ”loc”, ”skills”, ”experts”]. The selection of metapaths is made to ensure maximum connectivity. For metapath a, the traversal from experts-¿skills-¿experts ensures that a common skill is shared between two experts and that an expert node can be reached by using another expert's skills. For metapath b, we extend this relationship by adding additional skill and expert. We do this to increase the connectivity by traversing from one expert to another with skill in between. Meta-

path c incorporates location when traversed from expert to skill and then further to another or set of skills of an expert. This enables us to include location in graph traversal. We use an example to illustrate the integration of metapaths: Let's say we have a metapath that represents the relationship "expert e1 from a location l1 has worked with expert e2 from location l2 in team x. The subset of the skills between these two experts is represented as s. The metapath in this scenario would be like $e1- > s- > e2$. This would enable someone outside of team x to connect from expert e1 to e2 by using our defined metapath " This metapath connects two experts based on their shared project experience. By incorporating this metapath into the graph representation, we can capture experts' collective history and teamwork abilities. By including metapaths in the team formation process, we enhance the richness and depth of the analysis, enabling a more holistic evaluation of team compositions and the potential for effective collaboration and knowledge exchange[14]. Using metapaths, we can identify experts who have collaborated on similar projects or possess complementary project-related expertise. This information can be valuable in team formation as it helps identify experts with a proven track record of successful collaboration, improving the likelihood of effective teamwork and project outcomes. Integrating metapaths in the team formation allows us to consider experts' individual skills, geographic locations, collaborative history, and project-specific connections. By incorporating these higher-order relationships, we can better understand the potential interactions and synergies within teams. We strategically employ a random sampling technique to generate walks originating from each source node. This approach not only provides us with diverse pathways that offer a holistic understanding of the network's topology but also ensures that we capture the latent relationships and interactions inherent within the network. By initiating these walks from each source node, we guarantee comprehensive coverage of the entire graph structure, maximizing the likelihood of uncovering crucial patterns and connections. Furthermore, this random sampling method offers an unbiased exploration, mitigating potential risks of overfitting and enabling more generalized representations that can be pivotal for downstream applications.

4.0.4 Spatial Team Formation

Definition 5 (Spatial Team Formation) *Given subsets of skills* $\mathbf{s}$ *and geolocations* $\mathbf{l}$ *and all previous teams* $\mathcal{T}$ *as the training set, Spatial Team Formation estimates* $f_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{l})$ *using a multi-layer neural network that learns, from* $\mathcal{T}$*, to map the dense vector representations of subset of skills* $\mathbf{s}$*,* $v_{\mathbf{s}}$*, and of subset of geolocation* $\mathbf{l}$*,* $v_{\mathbf{l}}$*, obtained from the graph neural network* g_{φ}*, to the occurrence vector representation of subset of experts* $\mathbf{e}^*$*,* $v_{\mathbf{e}^*} \in \mathbb{R}^{|E|}$*, by maximizing the posterior (MAP) probability of* $\boldsymbol{\theta}$ *in* $f_{\boldsymbol{\theta}}$ *over* $\mathcal{T}$*, that is,* $\underset{\boldsymbol{\theta}}{argmax}\ p(\boldsymbol{\theta}|\mathcal{T})$.

We then use the learned vector representations (embeddings) for skills and location and feed this representation into a Bayesian neural network. Instead of learning a single parameter for each edge between nodes, Bayesian neural networks learn a distribution of parameters during the training process. This distribution includes a mean and a standard deviation for each edge. During inference, for each instance in the train/validation/test set, a set of parameters is sampled based on the learned distribution. Bayesian neural networks learn parameter distributions and multiple sample sets of parameters during inference and derive probabilities for each expert based on these samples. We utilize a variational neural network with a single dense variational hidden layer of size d, but it can be extended to multiple hidden layers without losing generality. This neural network employs a mapping function $f(s; \boldsymbol{\theta})$ to predict a team of experts $e \subseteq E$ for a given skill subset $s \subseteq S$ or a subset combining skill $s \subseteq S$ and location $l \subseteq L$. The input layer $v_s(s)$ and the output layer $v_e(e)$ are integrated into the network architecture.

$$
\begin{aligned}
h &= \pi_1(\boldsymbol{\theta}_1 v_s(s) + b_1) \\
v_e(e) &= \pi_2(\boldsymbol{\theta}_2 h + b_2) \\
\boldsymbol{\theta} &= \boldsymbol{\theta}_1 \cup \boldsymbol{\theta}_2 \cup b_1 \cup b_2
\end{aligned}
$$

where, $\pi_{\cdot}$ is a nonlinear activation function, $\boldsymbol{\theta} \sim \mathcal{N}(\mu, \sigma^2)$ whose means and

variances are estimated by minimizing variational free energy, $v_s(s)$ is the vector representation of the input skill subset s, or a combination of vector representation of skill s and vector representation of location l, and $v_e(e)$ is the vector representation of output expert subset e given a team $(s, e) \in \mathcal{T}$.

CHAPTER 5

Experiments and Results

In this chapter, we lay out the details of our experiments and expound on how we examined our proposed methods for the team formation problem. More concretely, we want to address the following research questions:

RQ1: Does considering the geographic location of experts on top of skills help with better team formation?

RQ2: Does heterogeneous graph modeling of teams with meta path-based vector representation for skills and locations improve performance compared to homogeneous graph modeling with random walk-based?

RQ3: Does increasing the location's granularity help increase the model performance?

RQ4: How does negative sampling affect the inclusion of location in the model's performance.

RQ5: Does the inlcusion of teams as nodes help improve the performance of the model or does it not add any value to the knolwedge learned by graphs.

5.0.1 Setup

Dataset We have used the US patents and trademarks dataset (uspt) for experimentation. This dataset contains the granted patents, classifications of patents and inventors associated with each patent, and their living location. We refer a team for each patent, including inventors as expert members, a collection of skills per team spread across each team expert as the required subset of skills, and locations as the experts' living locations. The locations of the inventors are divided into city,

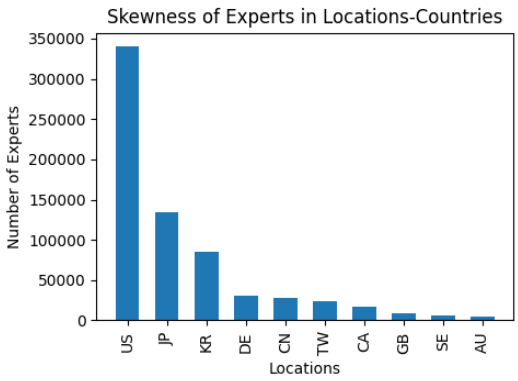

Fig. 5.0.1: Data skewness in location w.r.t # of experts

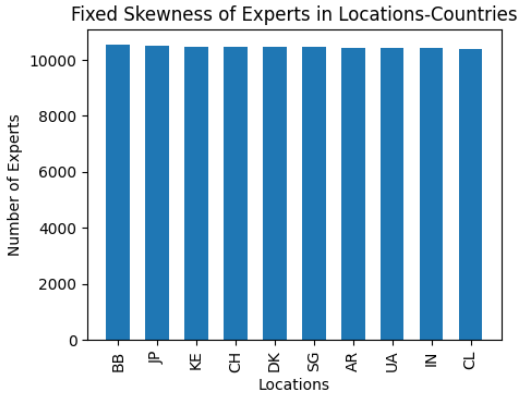

Fig. 5.0.2: Solved data skewness with the addition of synthetic data.

Fig. 5.0.3: Data skewness in a location with increased granularity to City

Fig. 5.0.4: Fixed data skewness with synthetic addition of data.

province, and country. From Figure 3, the distribution of locations within the dataset is skewed, with most inventors from US and Japan. To address this, we randomly sample country locations and city locations and assign it to random individuals to make the locations evenly distributed amongst inventors, as shown in graphs Figure 5.0.1, 5.0.2, 5.0.3, and 5.0.4. The skewness of locations in this dataset stems from the fact that in an ideal scenario, any individual could only belong to one location at a given time but possess multiple skills as part of his/her daily job.

Like Rad et al. [35] and following Dashti et al.[13] we filter out members who participated in less than 75 teams and teams with less than 3 members for dblp,

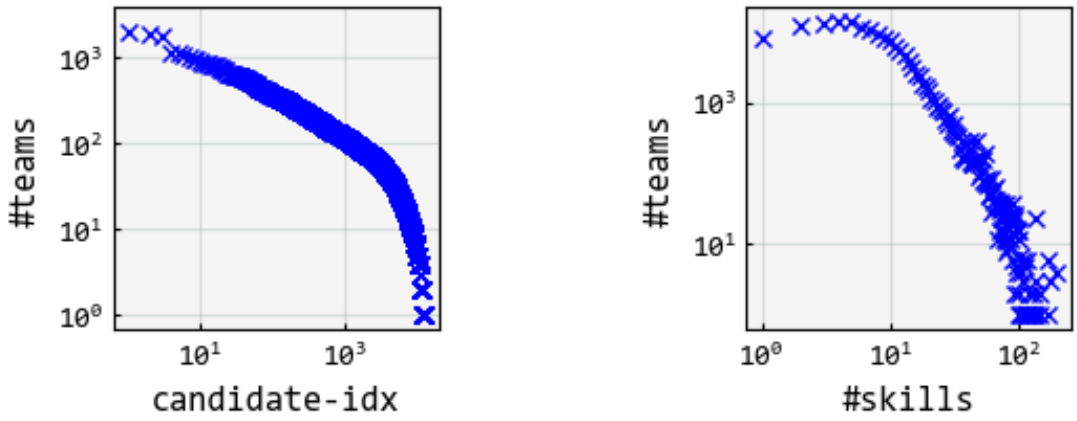

Fig. 5.0.5: Distribution of teams over candidates and skills in U.S. patents (uspt).

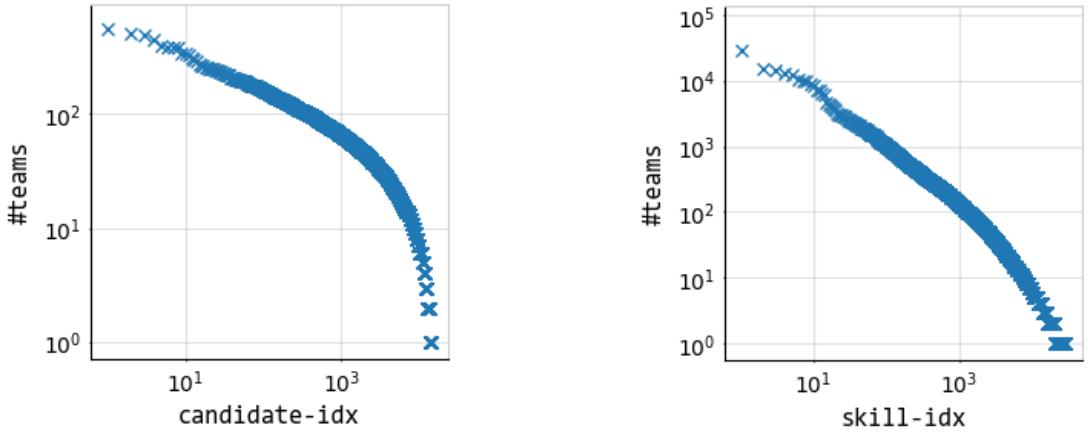

Fig. 5.0.6: Distribution of teams over candidates and skills in computer science publications (dblp).

uspt. In these datasets, we can observe long tails in the distributions of teams over experts. As shown in the left side of Figures 5.0.5 and 5.0.6 after filtering, many experts (researchers in dblp, inventors in uspt) have participated in very few teams (papers in dblp, inventions in uspt).

We utilize a neighbor sampler [17] that enables us for mini-batch training of GNNs on large-scale graphs where full-batch training is not feasible. Given a graph l layer and a specific mini-batch of nodes `node_idx` for which we want to compute embeddings, this module iteratively samples neighbors and constructs bipartite graphs that simulate the actual computation flow of a GNN. With this, we control how many

neighbors we want to sample for each node for each layer. We define our batch sizes as size $(x, 256)$ where x denotes the number of nodes given as an input to a graph layer and the output is of dimensions $(x, 128)$. We experiment with different hyperparameters for generating node embeddings and develop the optimal choice for `learning_rate` as 0.01 and a dropout of 0.4. The trained embeddings are then reconstructed into a previously defined matrix structure of teams X dimensions, where each row represents a team having skills, location, and expert matrices belonging to the same team. We follow the closed-world assumption where no currently known successful team for the required skills is assumed to be unsuccessful. We utilize three negative samplings inspired by the optimization function of Mikolov et al.[31] such as Uniform Negative Sampling, Unigram negative sampling, and smoothed unigram negative sampling. Bayesian neural networks employ a different approach to learning parameters than feedforward neural networks. The average probabilities are then utilized for evaluation purposes. For our neural model, we use a learning rate of 0.1, a batch size of 4096, and train the model for 20 epochs for each of the five folds. We conduct our experimentation in the following combinations: negative sampling: {uniform, unigram_b}, model architecture: {feedforward neural network(fnn), bayesian neural network(bnn)}, graph neural network: {metpaths, random-walk} and display our results from table 5.0.4, 5.0.5, 5.0.6, 5.0.7, 5.0.8, 5.0.9, 5.0.11, 5.0.12, 5.0.13 and 5.0.14.

5.0.2 Evaluation Metrics

To illustrate the effectiveness of graph neural models in predictions, we set aside 15% of teams from our datasets as a test set, conducting 5-fold cross-validation on the remaining teams for both training and validation. This leads to a distinct model for every fold. For any given team (s, e, l) in the test set, we evaluate the ranked list of experts e', as projected by the model from each fold, against the known subset of experts e. We then present the mean performance of models across all folds using metrics like normalized discounted cumulative gain (ndcg), mean average precision (map) at top-{2,5,10}, precision (pr), recall (rec), and area under the receiver op-

Table 5.0.1: Stats of uspt dataset

Utility Patents Stat	Value
#Patents (teams)	152,317
#Unique Inventors (members)	12,914
#Unique Subgroups (skills)	67,315
Avg #Inventors per Patent	3.78
Avg #Subgroup per Patent	9.97
Avg #Patent per Inventor	6.95
Avg #Subgroup per Inventor	102.52
#Patent w/ Single Inventor	0
#Patent w/ Single Subgroup	8110
#Unique Inventor's Locations	261*
Avg Inventors' Locations per Patent	2.50

Table 5.0.2: Stats of dblp dataset

Dblp v12	Value
#Publications (teams)	99,375
#Unique Authors (experts)	14,214
#Unique Field of Study (FOS) (skills)	29,661
Avg #Author per Publication	3.29
Avg #FOS per Publication	9.71
Avg #Publication per Author	23.02
Avg #FOS per Author	96.72
#Publication w/ Single Author	0
#Publication w/ Single FOS	56
#Unique Inventor's Venues	6393

*The Number of Unique Inventor Locations is 261, substantially more significant than the total number of countries worldwide. One Possible explanation of this could be because of the abbreviations used by USPT that might refer to different codes for the same countries or break down countries into different zones.

Table 5.0.3: # Graph Nodes in dataset: uspt and dblp

# of graph nodes	USPT- with country	USPT- with city	DBLP with venues
Experts	13,631	12,914	29,661
Skills	69,679	67,315	14,214
Location	70	9,422	6393

erating characteristic (rocauc) utilizing tools like pytrec eval[1] and scikit-learn[2]. To discern the efficiency benefits of negative sampling during the neural models' training phase, while still retaining inference accuracy, we subjected the baseline models to an escalating epoch count from {1,...,20}, assessing them on the test set after each epoch.

Regarding the influence of the streaming training approach and the integration of temporal data into input embeddings for future team predictions, we reserved the latest year of each dataset for testing. Ensuring the robustness of our method, we applied 5-fold cross-validation annually on teams for training and validation. For a specific team $(s, e, l)_{T+1}$ in the test, the model's expert predictions e' for each fold were juxtaposed with the actual expert subset e, with the cumulative performance of models across folds being evaluated using the previously mentioned metrics.

5.0.3 Results

We demonstrate our results on uspt-country, uspt-country+city and dblp dataset. We evaluate our proposed pipeline with baselines from [35] and [13] with models being used are a combination of bayesian or feedforward neural network with uniform and unigram_b negative sampling. We have used two variants of graph neural models such as metapaths and random walks that includes the use of graph networks.

In response to RQ1, whether considering the geographic location of experts on top of skills helps with better team formation, it is observed that the inclusion of location does leave a positive effect on the metrics but we also observe cases where

[1]https://github.com/cvangysel/pytrec_eval
[2]https://scikit-learn.org

Table 5.0.4: Results of Training of bnn model with metapaths and random walks using gnn on USPT skewed dataset with locations as countries with uniform negative sampling

	P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
bnn_gnn_loc_meta - uni[35]	0.0050	0.0045	0.0038	0.0023	0.0048	0.0083	0.0051	0.0053	0.0067	0.0018	0.0028	0.0035	0.5657
bnn_gnn_meta - uni_b[35]	0.0092	0.0079	0.0065	0.0043	0.0090	0.0145	0.0092	0.0095	0.0117	0.0032	0.0048	0.0058	0.7462
bnn	0.0105	0.0089	0.0073	0.0048	0.0100	0.0160	0.0106	0.0108	0.0132	0.0038	0.0055	0.0067	0.7427
bnn_gnn_loc_meta	0.0055	0.0043	0.0038	0.0024	0.0047	0.0081	0.0057	0.0054	0.0068	0.0020	0.0028	0.0035	0.5715
bnn_gnn_meta	0.0082	0.0074	0.0060	0.0040	0.0086	0.0135	0.0082	0.0089	0.0110	0.0031	0.0045	0.0055	0.7350
bnn_gnn_loc	0.0071	0.0068	0.0059	0.0030	0.0075	0.0127	0.0072	0.0079	0.0100	0.0025	0.0039	0.0050	0.6287
bnn_gnn	0.0107	0.0096	0.0076	0.0053	0.0111	0.0170	0.0109	0.0116	0.0141	0.0041	0.0059	0.0071	0.7595

Table 5.0.5: Results of Training of BNN model with metapaths and random walks using GNN on the synthetically fixed USPT dataset with locations as countries on synthetically fixed dataset to remove skewness with unigram_b negative sampling

	P@2	P@5	P@10	Recall@2	Recall@5	Recall@10	NDCG@2	NDCG@5	NDCG@10	MAP@2	MAP@5	MAP@10	AUCROC
bnn	0.0105	0.0089	0.0073	0.0048	0.0100	0.0160	0.0106	0.0108	0.0132	0.0038	0.0055	0.0067	0.7427
bnn_gnn_loc_meta	0.0105	0.0090	0.0072	0.0050	0.0104	0.0166	0.0105	0.0109	0.0134	0.0038	0.0055	0.0066	0.7694
bnn_gnn_meta	0.0047	0.0042	0.0036	0.0023	0.0050	0.0085	0.0047	0.0050	0.0066	0.0017	0.0025	0.0031	0.6840
bnn_gnn_loc	0.0129	0.0106	0.0081	0.0062	0.0119	0.0177	0.0132	0.0132	0.0155	0.0050	0.0070	0.0082	0.7924
bnn_gnn	0.0132	0.0106	0.0083	0.0066	0.0121	0.0183	0.0140	0.0137	0.0163	0.0056	0.0074	0.0086	0.7948

just by having a homogeneous graph with only skills as nodes is a better performer than a heterogeneous graph of skills and location as nodes. We observe that the graph embeddings, regardless of the case of having location as nodes in the graph, perform better than the vectors learned through sparse matrices. The results of the column bnn in table 3. consisting of only skills denotes the use of sparse vector representations and is shown to perform the worst of all the implemented baselines and our state-of-the-art methods. The graph embedding does improve the recommendation by an average of 5% when compared with just the sparse representations. The difference

Table 5.0.6: Results of Training of fnn model with metapaths and random walks using gnn on the USPT dataset with locations as countrieson skewed and synthetically fixed dataset with uniform negative sampling

		P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
Skewed - Uniform	fnn_gnn_loc_meta	0.0013	0.0014	0.0015	0.0006	0.0017	0.0034	0.0012	0.0016	0.0024	0.0005	0.0008	0.0011	0.5826
	fnn_gnn_meta	0.0013	0.0014	0.0014	0.0007	0.0017	0.0034	0.0013	0.0016	0.0024	0.0005	0.0008	0.0011	0.6744
	fnn_gnn_loc	0.0015	0.0015	0.0014	0.0007	0.0018	0.0032	0.0015	0.0017	0.0024	0.0006	0.0009	0.0012	0.5365
	fnn_gnn	0.0019	0.0018	0.0017	0.0009	0.0021	0.0040	0.0019	0.0022	0.0030	0.0007	0.0011	0.0015	0.5536
Synthetically Fixed - Uniform	fnn_gnn_loc_meta	0.0018	0.0018	0.0018	0.0009	0.0022	0.0043	0.0019	0.0022	0.0031	0.0007	0.0011	0.0014	0.6852
	fnn_gnn_meta	0.0010	0.0011	0.0011	0.0005	0.0015	0.0029	0.0010	0.0013	0.0020	0.0004	0.0007	0.0009	0.6467
	fnn_gnn_loc	0.0114	0.0109	0.0084	0.0052	0.0125	0.0190	0.0117	0.0131	0.0156	0.0044	0.0067	0.0079	0.7967
	fnn_gnn	0.0111	0.0098	0.0078	0.0055	0.0114	0.0173	0.0113	0.0120	0.0145	0.0043	0.0061	0.0073	0.7799

Table 5.0.7: Results of Training of bnn model with metapaths and random walks using gnn on the USPT dataset with locations as countrieson skewed and synthetically fixed dataset with unigram_b negative sampling

		P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
Skewed - Unigram_b	bnn_gnn_loc_meta	0.0102	0.0086	0.0070	0.0048	0.0098	0.0158	0.0102	0.0104	0.0128	0.0036	0.0052	0.0063	0.7682
	bnn_gnn_meta	0.0066	0.0057	0.0047	0.0032	0.0066	0.0107	0.0066	0.0069	0.0086	0.0024	0.0035	0.0043	0.7148
	bnn_gnn_loc	0.0118	0.0105	0.0083	0.0057	0.0119	0.0188	0.0120	0.0126	0.0154	0.0044	0.0065	0.0076	0.7956
	bnn_gnn	0.0123	0.0106	0.0084	0.0060	0.0122	0.0189	0.0123	0.0129	0.0156	0.0045	0.0066	0.0078	0.7962
Synthetically Fixed - Unigram_b	bnn_gnn_loc_meta	0.0102	0.0086	0.0070	0.0048	0.0098	0.0158	0.0102	0.0104	0.0128	0.0036	0.0052	0.0063	0.7682
	bnn_gnn_meta	0.0098	0.0084	0.0068	0.0046	0.0097	0.0153	0.0099	0.0102	0.0126	0.0036	0.0052	0.0062	0.7543
	bnn_gnn_loc	0.0120	0.0105	0.0083	0.0057	0.0119	0.0187	0.0121	0.0126	0.0154	0.0044	0.0064	0.0076	0.7957
	bnn_gnn	0.0123	0.0107	0.0084	0.0059	0.0124	0.0189	0.0125	0.0131	0.0157	0.0046	0.0067	0.0079	0.7975

Table 5.0.8: Results of Training of fnn model with metapaths and random walks using gnn on the USPT dataset with locations as countries on skewed and synthetically fixed dataset with unigram_b negative sampling

		P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
Skewed - Unigram_b	fnn_gnn_loc_meta	0.0015	0.0016	0.0015	0.0007	0.0018	0.0035	0.0015	0.0018	0.0025	0.0006	0.0009	0.0012	0.5776
	fnn_gnn_meta	0.0014	0.0015	0.0015	0.0007	0.0018	0.0037	0.0014	0.0018	0.0026	0.0006	0.0009	0.0012	0.6742
	fnn_gnn_loc	0.0018	0.0016	0.0014	0.0008	0.0019	0.0034	0.0018	0.0019	0.0026	0.0007	0.0010	0.0013	0.5347
	fnn_gnn	0.0016	0.0016	0.0015	0.0008	0.0020	0.0036	0.0016	0.0019	0.0027	0.0006	0.0010	0.0013	0.5581
Synthetically Fixed - Unigram_b	fnn_gnn_loc_meta	0.0018	0.0018	0.0018	0.0009	0.0023	0.0044	0.0018	0.0021	0.0032	0.0007	0.0011	0.0014	0.6867
	fnn_gnn_meta	0.0016	0.0017	0.0017	0.0008	0.0021	0.0041	0.0016	0.0020	0.0029	0.0006	0.0010	0.0013	0.6791
	fnn_gnn_loc	0.0121	0.0109	0.0083	0.0059	0.0126	0.0190	0.0123	0.0132	0.0158	0.0046	0.0068	0.0079	0.7972
	fnn_gnn	0.0121	0.0107	0.0083	0.0059	0.0124	0.0183	0.0122	0.0129	0.0153	0.0045	0.0066	0.0078	0.7927

Table 5.0.9: Results of Training of bnn model with metapaths and random walks using gnn on the USPT dataset with locations as countries+cities on the skewed and synthetically fixed dataset with uniform negative sampling

		P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
Skewed - Uniform	bnn_gnn_loc_meta	0.0018	0.0017	0.0017	0.0009	0.0020	0.0039	0.0018	0.0020	0.0029	0.0007	0.0010	0.0014	0.5778
	bnn_gnn_meta	0.0095	0.0083	0.0068	0.0045	0.0091	0.0147	0.0098	0.0101	0.0124	0.0036	0.0052	0.0063	0.7458
	bnn_gnn_loc	0.0070	0.0066	0.0058	0.0030	0.0070	0.0123	0.0072	0.0077	0.0098	0.0025	0.0038	0.0049	0.6453
	bnn_gnn	0.0131	0.0112	0.0089	0.0063	0.0124	0.0190	0.0135	0.0138	0.0165	0.0051	0.0073	0.0087	0.7981
Synthetically Fixed - Uniform	bnn_gnn_loc_meta	0.0020	0.0019	0.0018	0.0010	0.0023	0.0043	0.0020	0.0023	0.0032	0.0008	0.0012	0.0016	0.5733
	bnn_gnn_meta	0.0061	0.0051	0.0043	0.0032	0.0062	0.0100	0.0063	0.0065	0.0082	0.0026	0.0034	0.0041	0.6929
	bnn_gnn_loc	0.0108	0.0092	0.0075	0.0051	0.0102	0.0161	0.0111	0.0114	0.0138	0.0042	0.0060	0.0072	0.7547
	bnn_gnn	0.0107	0.0087	0.0071	0.0052	0.0097	0.0155	0.0108	0.0108	0.0133	0.0042	0.0057	0.0069	0.7329

Table 5.0.10: Results of Training of fnn model with metapaths and random walks using gnn on the USPT dataset with locations as countries+cities on the skewed and synthetically fixed dataset with uniform negative sampling

		P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
Skewed - Uniform	fnn_gnn_loc_meta	0.0008	0.0010	0.0010	0.0004	0.0012	0.0023	0.0008	0.0011	0.0016	0.0003	0.0006	0.0008	0.5593
	fnn_gnn_meta	0.0019	0.0020	0.0020	0.0009	0.0024	0.0047	0.0018	0.0023	0.0034	0.0007	0.0011	0.0015	0.6783
	fnn_gnn_loc	0.0024	0.0022	0.0020	0.0011	0.0025	0.0045	0.0024	0.0026	0.0034	0.0009	0.0014	0.0018	0.5386
	fnn_gnn	0.0043	0.0038	0.0033	0.0018	0.0040	0.0069	0.0044	0.0044	0.0056	0.0014	0.0023	0.0029	0.5915
Synthetically Fixed - Uniform	fnn_gnn_loc_meta	0.0008	0.0010	0.0010	0.0004	0.0012	0.0025	0.0008	0.0011	0.0017	0.0003	0.0006	0.0008	0.5562
	fnn_gnn_meta	0.0047	0.0040	0.0035	0.0021	0.0044	0.0076	0.0048	0.0048	0.0062	0.0017	0.0026	0.0033	0.6132
	fnn_gnn_loc	0.0087	0.0069	0.0058	0.0041	0.0078	0.0127	0.0089	0.0087	0.0107	0.0033	0.0047	0.0057	0.6829
	fnn_gnn	0.0018	0.0018	0.0018	0.0009	0.0022	0.0043	0.0018	0.0021	0.0031	0.0007	0.0011	0.0014	0.6765

Table 5.0.11: Results of Training of bnn model with metapaths and random walks using gnn on the USPT dataset with locations as countries+cities on the skewed and synthetically fixed dataset with unigram_b negative sampling

		P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
Skewed - Unigram_b	bnn_gnn_loc_meta	0.0018	0.0017	0.0016	0.0009	0.0021	0.0037	0.0018	0.0020	0.0028	0.0007	0.0011	0.0014	0.5697
	bnn_gnn_meta	0.0058	0.0049	0.0042	0.0031	0.0060	0.0098	0.0059	0.0062	0.0080	0.0024	0.0033	0.0039	0.6910
	bnn_gnn_loc	0.0083	0.0070	0.0059	0.0036	0.0074	0.0125	0.0083	0.0083	0.0103	0.0029	0.0043	0.0054	0.6483
	bnn_gnn	0.0124	0.0108	0.0087	0.0061	0.0121	0.0187	0.0129	0.0135	0.0162	0.0051	0.0071	0.0085	0.7975
Synthetically Fixed - Unigram_b	bnn_gnn_loc_meta	0.0018	0.0016	0.0015	0.0009	0.0020	0.0037	0.0018	0.0020	0.0028	0.0007	0.0011	0.0014	0.5723
	bnn_gnn_meta	0.0098	0.0083	0.0068	0.0045	0.0091	0.0146	0.0100	0.0102	0.0124	0.0037	0.0053	0.0065	0.7478
	bnn_gnn_loc	0.0124	0.0107	0.0085	0.0059	0.0116	0.0182	0.0127	0.0131	0.0157	0.0049	0.0069	0.0083	0.7944
	bnn_gnn	0.0124	0.0104	0.0083	0.0058	0.0115	0.0177	0.0127	0.0128	0.0154	0.0048	0.0067	0.0081	0.7825

Table 5.0.12: Results of Training of fnn model with metapaths and random walks using gnn on the USPT dataset with locations as countries+cities on the skewed and synthetically fixed dataset with unigram_b negative sampling

		P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
Skewed - Unigram_b	fnn_gnn_loc_meta	0.0010	0.0011	0.0011	0.0005	0.0013	0.0026	0.0010	0.0012	0.0018	0.0004	0.0007	0.0009	0.5614
	fnn_gnn_meta	0.0012	0.0013	0.0013	0.0006	0.0016	0.0032	0.0012	0.0015	0.0023	0.0005	0.0008	0.0010	0.6590
	fnn_gnn_loc	0.0025	0.0022	0.0019	0.0012	0.0026	0.0044	0.0026	0.0027	0.0035	0.0010	0.0015	0.0019	0.5392
	fnn_gnn	0.0037	0.0034	0.0031	0.0016	0.0035	0.0064	0.0037	0.0039	0.0051	0.0013	0.0020	0.0027	0.5935
Synthetically Fixed - Unigram_b	fnn_gnn_loc_meta	0.0010	0.0011	0.0011	0.0005	0.0014	0.0027	0.0010	0.0013	0.0019	0.0004	0.0007	0.0009	0.5577
	fnn_gnn_meta	0.0050	0.0043	0.0038	0.0024	0.0049	0.0084	0.0051	0.0052	0.0067	0.0019	0.0028	0.0036	0.6198
	fnn_gnn_loc	0.0021	0.0021	0.0021	0.0011	0.0026	0.0050	0.0021	0.0025	0.0036	0.0008	0.0013	0.0017	0.6818
	fnn_gnn	0.0079	0.0064	0.0053	0.0038	0.0073	0.0120	0.0083	0.0081	0.0101	0.0032	0.0044	0.0053	0.6794

Table 5.0.13: Results of Training of bnn and fnn model with metapaths and random walks using gnn on the DBLP dataset with locations as venues with uniform negative sampling

	P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
bnn_gnn_loc_meta	0.0028	0.0024	0.0021	0.0016	0.0035	0.0061	0.0028	0.0031	0.0044	0.0012	0.0018	0.0022	0.6867
bnn_gnn_meta	0.0039	0.0036	0.0032	0.0022	0.0052	0.0093	0.0039	0.0047	0.0066	0.0017	0.0025	0.0031	0.7233
bnn_gnn_loc	0.0054	0.0046	0.0041	0.0031	0.0067	0.0120	0.0054	0.0061	0.0086	0.0023	0.0033	0.0041	0.7606
bnn_gnn	0.0048	0.0050	0.0043	0.0028	0.0072	0.0125	0.0047	0.0062	0.0087	0.0021	0.0034	0.0041	0.7605
fnn_gnn_loc_meta	0.0007	0.0008	0.0008	0.0004	0.0012	0.0022	0.0007	0.0009	0.0015	0.0003	0.0005	0.0007	0.6506
fnn_gnn_meta	0.0009	0.0009	0.0009	0.0005	0.0013	0.0027	0.0009	0.0012	0.0018	0.0004	0.0006	0.0008	0.6587
fnn_gnn_loc	0.0013	0.0013	0.0012	0.0008	0.0019	0.0037	0.0013	0.0016	0.0025	0.0006	0.0009	0.0012	0.5953
fnn_gnn	0.0023	0.0020	0.0018	0.0013	0.0029	0.0053	0.0024	0.0027	0.0038	0.0011	0.0015	0.0019	0.6237

Table 5.0.14: Results of Training of bnn and fnn model with metapaths and random walks using gnn on the DBLP dataset with locations as venues with unigram_b negative sampling

	P@2	P@5	P@10	recall@2	recall@5	recall@10	ndcg@2	ndcg@5	ndcg@10	map@2	map@5	map@10	aucroc
bnn_gnn_loc_meta	0.0027	0.0025	0.0022	0.0015	0.0036	0.0064	0.0027	0.0032	0.0045	0.0012	0.0018	0.0022	0.6819
bnn_gnn_meta	0.0047	0.0040	0.0033	0.0027	0.0057	0.0096	0.0048	0.0053	0.0070	0.0021	0.0030	0.0035	0.7218
bnn_gnn_loc	0.0053	0.0046	0.0040	0.0031	0.0066	0.0117	0.0054	0.0061	0.0084	0.0023	0.0034	0.0041	0.7603
bnn_gnn	0.0047	0.0045	0.0040	0.0026	0.0064	0.0116	0.0046	0.0057	0.0081	0.0020	0.0031	0.0038	0.7576
fnn_gnn_loc_meta	0.0011	0.0012	0.0011	0.0006	0.0017	0.0032	0.0011	0.0014	0.0021	0.0005	0.0008	0.0010	0.5919
fnn_gnn_meta	0.0018	0.0019	0.0018	0.0011	0.0028	0.0052	0.0019	0.0024	0.0036	0.0008	0.0014	0.0017	0.6230
fnn_gnn_loc	0.0010	0.0010	0.0010	0.0006	0.0015	0.0029	0.0010	0.0013	0.0020	0.0005	0.0007	0.0009	0.6583
fnn_gnn	0.0007	0.0007	0.0007	0.0004	0.0010	0.0021	0.0007	0.0009	0.0014	0.0003	0.0005	0.0006	0.6515

of the results for the column of skewed data for uspt-country and uspt-country+city stems from the skewness of the data in terms of location which is present in the dataset. The skewness of the locations where most of the teams are sampled from one country, prohibits the expert recommendation to use the benefit of knowledge learned from location representations. We also observe that inclusion of geographic proximity in some cases does improve upon the results of not considering when the same result is compared against an equal distribution of locations(Table 4.), the results of the graph models give an improvement of about 7% percentage and a 2% increase in the information retrieval metrics. The same is also proved in the case of dblp where the percentage increase in metrics is evident. This gives concrete support to the statement that the inclusion of geographic location in the context of team formation in accordance with the skills leads to a better recommendation of experts. We also explain the variability of the results as seen amongst the reported metrics as a possible reason to what percentage a team is affected by the inclusion of location. This develops into another conclusion, that the effects of location also depend upon the nature of data or the field where the team formation is applied to.

In response to RQ2, whether a heterogeneous graph modeling of teams with meta path-based vector representation for skills and locations improves performance compared to heterogeneous and homogeneous graph modeling with random walk-based, our results indicate that the inclusion of metapaths doesn't necessarily improve the performance of our model in information retrieval and classification metrics. From

table 3 and table 4 show an improvement of around 10-15% in the skewed dataset and only a minor improvement of 2 - 5% for the fixed dataset. The results are synonymous across the three different dataset testbeds we experimented our methodology on. Nearly 10% of increase was shown in reported metrics when random walk based embedding learning was used instead of metpath based walk. This concludes that random walks can also generate coverage of graphs that is at par with the metapaths' approach and sometimes even better. These findings further cement the conclusion made by [23] where the authors states that the application of metapath2vec [14] in a heterogeneous graph setting doesn't necessarily improves the performance of graph based recommender systems. We further reflect on the results of [35] where the authors use a heterogeneous graph setting with nodes of types experts, skills, location and teams fails to outperform our implementation of using just the experts, skills and location as nodes, meaning that the addition of teams' nodes doesn't accentuate the vector representation of graph embeddings and removing them could improve performance of the expert recommendation engine.

In response to RQ3, Decreasing the granularity to cities does have a positive effect on the reported metrics. This stems from the fact that in a given team there would be more variety of cities for experts than countries. For example, a team having 3 experts from one country will just have a single entry for location, but when we go a little more deeper to include cities, all the 3 experts can belong from different cities within the same country. This promotes inclusion of more location specific data for experimentation and a spread out distribution of location among experts.

In response to RQ4 and RQ5, The effect of negative sampling on the experimental results is quite evident from the results tables. The parameter setting with unigram_b as negative sampling outperformed when compared to unfiorm negative sampling when compared across our entire test bed. We also notice that the inclusion of team nodes does not add any substantial knowledge gain to the graph neural networks. This is seen as the performance in the result table 5.0.4 where the first two column represents the graph setting that included teams as nodes.

CHAPTER 6

Conclusion and Future Work

Keeping the outcomes of this scientific work in consideration, we draw a conclusion. A team's needs are described by many factors in play such as skills and location. Realistically speaking, the location and skills does prove highly significant for teams. A typical organization generally oversees these complexities while forming a team. Necessarily, the inclusion of location does improve the overall performance of team.

In this thesis, we proposed a training strategy for the problem of team formation that involves inclusion of location with skills while trying to predict experts for a team. We examine the effect of implementing a heterogeneous graph with nodes of experts, skills and locations and compare its performance against a homogeneous graph with only skills as a node. We performed extensive experimentation on two datasets uspt and imdb to examine how our approach results in two different data domains. Our experiments show that (i) locations give an overall improvement over the data set involving just the expert and location. (ii) We also show that graphs can capture more complex connections between experts and skills, experts and location than a simple neural network.(iii) We experiment with metapaths and random walk training strategies in our setup. Our results show that in this team formation problem, the results of the two training strategies fare evenly against each other, with the random walk training strategy gaining an edge in classification and information retrieval metrics. We aim to extend this research beyond by developing a novel end-to-end graph neural network + deep learning architecture and experiment with different granularity of locations' effects on the outcome.

REFERENCES

[usp] `https://patentsview.org/download/data-download-tables`. [Accessed 14-05-2022].

[dbl] `https://aminer.org/citation`. [Accessed 14-05-2022].

[3] Anagnostopoulos, A., Becchetti, L., Castillo, C., Gionis, A., and Leonardi, S. (2012). Online team formation in social networks. In Mille, A., Gandon, F., Misselis, J., Rabinovich, M., and Staab, S., editors, *Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012*, pages 839–848. ACM.

[4] Awal, G. K. and Bharadwaj, K. K. (2014). Team formation in social networks based on collective intelligence - an evolutionary approach. *Appl. Intell.*, 41(2):627–648.

[5] Baykasoglu, A., Dereli, T., and Das, S. (2007a). Project team selection using fuzzy optimization approach. *Cybern. Syst.*, 38(2):155–185.

[6] Baykasoglu, A., Dereli, T., and Das, S. (2007b). Project team selection using fuzzy optimization approach. *Cybernetics and Systems: An International Journal*, 38(2):155–185.

[7] Borchers, G. (2003). The software engineering impacts of cultural factors on multi-cultural software development teams. In Clarke, L. A., Dillon, L., and Tichy, W. F., editors, *Proceedings of the 25th International Conference on Software Engineering, May 3-10, 2003, Portland, Oregon, USA*, pages 540–547. IEEE Computer Society.

[8] Bursic, K. M. (1992). Strategies and benefits of the successful use of teams in manufacturing organizations. *IEEE transactions on engineering management*, 39(3):277–289.

[9] Campêlo, M. B., Figueiredo, T. F., and Silva, A. (2020). The sociotechnical teams formation problem: a mathematical optimization approach. *Ann. Oper. Res.*, 286(1):201–216.

[10] Craig, M. and McKeown, D. (2015). How to build effective teams in healthcare. *Nursing times*, 111(14):16–18.

[11] Cummings, J. N. and Kiesler, S. B. (2008). Who collaborates successfully?: prior experience reduces collaboration barriers in distributed interdisciplinary research. In Begole, B. and McDonald, D. W., editors, *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work, CSCW 2008, San Diego, CA, USA, November 8-12, 2008*, pages 437–446. ACM.

[12] Dashti, A., Samet, S., and Fani, H. (2022a). Effective neural team formation via negative samples. In Hasan, M. A. and Xiong, L., editors, *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 3908–3912. ACM.

[13] Dashti, A., Saxena, K., Patel, D., and Fani, H. (2022b). Opentf: A benchmark library for neural team formation. In Hasan, M. A. and Xiong, L., editors, *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 3913–3917. ACM.

[14] Dong, Y., Chawla, N. V., and Swami, A. (2017). metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 135–144. ACM.

[15] Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. (2017). Neural message passing for quantum chemistry. In Precup, D. and Teh, Y. W.,

editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR.

[16] Glance, N. S. and Huberman, B. A. (1993). Organizational fluidity and sustainable cooperation. In Castelfranchi, C. and Müller, J., editors, *From Reaction to Cognition, 5th European Workshop on Modelling Autonomous Agents, MAAMAW '93, Neuchatel, Switzerland, August 25-27, 1993, Selected Papers*, volume 957 of *Lecture Notes in Computer Science*, pages 89–103. Springer.

[17] Hamilton, W. L., Ying, Z., and Leskovec, J. (2017). Inductive representation learning on large graphs. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034.

[18] Huberman, B. A. and Hogg, T. (1995). Communities of practice: Performance and evolution. *Comput. Math. Organ. Theory*, 1(1):73–92.

[19] Juárez, J., Santos, C. P., and Brizuela, C. A. (2022). A comprehensive review and a taxonomy proposal of team formation problems. *ACM Comput. Surv.*, 54(7):153:1–153:33.

[20] Kargar, M. and An, A. (2011). Discovering top-k teams of experts with/without a leader in social networks. In Macdonald, C., Ounis, I., and Ruthven, I., editors, *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October 24-28, 2011*, pages 985–994. ACM.

[21] Kargar, M., An, A., and Zihayat, M. (2012). Efficient bi-objective team formation in social networks. In Flach, P. A., Bie, T. D., and Cristianini, N., editors, *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML*

PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II, volume 7524 of *Lecture Notes in Computer Science*, pages 483–498. Springer.

[22] Kargar, M., Zihayat, M., and An, A. (2013). Finding affordable and collaborative teams from a network of experts. In *Proceedings of the 2013 SIAM international conference on data mining*, pages 587–595. SIAM.

[23] Kaw, S., Kobti, Z., and Selvarajah, K. (2023). Transfer learning with graph attention networks for team recommendation. In *International Joint Conference on Neural Networks, IJCNN 2023, Gold Coast, Australia, June 18-23, 2023*, pages 1–8. IEEE.

[24] Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

[25] Kunegis, J., Preusse, J., and Schwagereit, F. (2013). What is the added value of negative links in online social networks? In Schwabe, D., Almeida, V. A. F., Glaser, H., Baeza-Yates, R., and Moon, S. B., editors, *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 727–736. International World Wide Web Conferences Steering Committee / ACM.

[26] Lappas, T., Liu, K., and Terzi, E. (2009). Finding a team of experts in social networks. In IV, J. F. E., Fogelman-Soulié, F., Flach, P. A., and Zaki, M. J., editors, *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009*, pages 467–476. ACM.

[27] Leskovec, J., Huttenlocher, D. P., and Kleinberg, J. M. (2010). Predicting positive and negative links in online social networks. In Rappa, M., Jones, P., Freire, J., and Chakrabarti, S., editors, *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, pages 641–650. ACM.

[28] Li, C. and Shan, M. (2010). Team formation for generalized tasks in expertise social networks. In Elmagarmid, A. K. and Agrawal, D., editors, *Proceedings of the 2010 IEEE Second International Conference on Social Computing, SocialCom / IEEE International Conference on Privacy, Security, Risk and Trust, PASSAT 2010, Minneapolis, Minnesota, USA, August 20-22, 2010*, pages 9–16. IEEE Computer Society.

[29] Li, L., Tong, H., Cao, N., Ehrlich, K., Lin, Y., and Buchler, N. (2015). Replacing the irreplaceable: Fast algorithms for team member recommendation. In Gangemi, A., Leonardi, S., and Panconesi, A., editors, *Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy, May 18-22, 2015*, pages 636–646. ACM.

[30] Majumder, A., Datta, S., and Naidu, K. V. M. (2012). Capacitated team formation problem on social networks. In Yang, Q., Agarwal, D., and Pei, J., editors, *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012*, pages 1005–1013. ACM.

[31] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Burges, C. J. C., Bottou, L., Ghahramani, Z., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

[32] Morrison-Smith, S. and Ruiz, J. (2020). Challenges and barriers in virtual teams: a literature review. *SN Applied Sciences*, 2:1–33.

[33] Qin, P., Xu, W., and Guo, J. (2016). A novel negative sampling based on TFIDF for learning word representation. *Neurocomputing*, 177:257–265.

[34] Rad, R. H., Bagheri, E., Kargar, M., Srivastava, D., and Szlichta, J. (2021). Retrieving skill-based teams from collaboration networks. In Diaz, F., Shah, C.,

Suel, T., Castells, P., Jones, R., and Sakai, T., editors, *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 2015–2019. ACM.

[35] Rad, R. H., Fani, H., Kargar, M., Szlichta, J., and Bagheri, E. (2020). Learning to form skill-based teams of experts. In d'Aquin, M., Dietze, S., Hauff, C., Curry, E., and Cudré-Mauroux, P., editors, *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 2049–2052. ACM.

[36] Sapienza, A., Goyal, P., and Ferrara, E. (2019). Deep neural networks for optimal team composition. *Frontiers Big Data*, 2:14.

[37] Selvarajah, K., Bhullar, A., Kobti, Z., and Kargar, M. (2018a). WSCAN-TFP: weighted SCAN clustering algorithm for team formation problem in social network. In Brawner, K. and Rus, V., editors, *Proceedings of the Thirty-First International Florida Artificial Intelligence Research Society Conference, FLAIRS 2018, Melbourne, Florida, USA. May 21-23 2018*, pages 209–212. AAAI Press.

[38] Selvarajah, K., Zadeh, P. M., Kargar, M., and Kobti, Z. (2019). Identifying a team of experts in social networks using a cultural algorithm. In Shakshuki, E. M. and Yasar, A., editors, *The 10th International Conference on Ambient Systems, Networks and Technologies (ANT 2019) / The 2nd International Conference on Emerging Data and Industry 4.0 (EDI40 2019) / Affiliated Workshops, April 29 - May 2, 2019, Leuven, Belgium*, volume 151 of *Procedia Computer Science*, pages 477–484. Elsevier.

[39] Selvarajah, K., Zadeh, P. M., Kobti, Z., Kargar, M., Ishraque, M. T., and Pfaff, K. (2018b). Team formation in community-based palliative care. In *2018 Innovations in Intelligent Systems and Applications, INISTA 2018, Thessaloniki, Greece, July 3-5, 2018*, pages 1–7. IEEE.

[40] Selvarajah, K., Zadeh, P. M., Kobti, Z., Palanichamy, Y., and Kargar, M. (2021).

A unified framework for effective team formation in social networks. *Expert Syst. Appl.*, 177:114886.

[41] Tang, J., Chang, Y., Aggarwal, C. C., and Liu, H. (2016). A survey of signed network mining in social media. *ACM Comput. Surv.*, 49(3):42:1–42:37.

[42] Thurow, A. P., Abdalla, C. W., Younglove-Webb, J., and Gray, B. (1999). The dynamics of multidisciplinary research teams in academia. *The review of higher education*, 22(4):425–440.

[43] Wang, W., Jiang, J., An, B., Jiang, Y., and Chen, B. (2017). Toward efficient team formation for crowdsourcing in noncooperative social networks. *IEEE Trans. Cybern.*, 47(12):4208–4222.

[44] Wu, L., Chen, Y., Shen, K., Guo, X., Gao, H., Li, S., Pei, J., and Long, B. (2023a). Graph neural networks for natural language processing: A survey. *Found. Trends Mach. Learn.*, 16(2):119–328.

[45] Wu, S., Sun, F., Zhang, W., Xie, X., and Cui, B. (2023b). Graph neural networks in recommender systems: A survey. *ACM Comput. Surv.*, 55(5):97:1–97:37.

[46] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. S. (2021). A comprehensive survey on graph neural networks. *IEEE Trans. Neural Networks Learn. Syst.*, 32(1):4–24.

[47] Ye, Z., Kumar, Y. J., Sing, G. O., Song, F., and Wang, J. (2022). A comprehensive survey of graph neural networks for knowledge graphs. *IEEE Access*, 10:75729–75741.

[48] Zzkarian, A. and Kusiak, A. (1999). Forming teams: an analytical approach. *IIE transactions*, 31(1):85–97.

VITA AUCTORIS

NAME:	Karan Saxena
PLACE OF BIRTH:	New Delhi, India
YEAR OF BIRTH:	1996
EDUCATION:	Amity University, B.Tech Computer Science, Noida, Uttar Pradesh, India, 2018
	University of Windsor, M.Sc in Computer Science, Windsor, Ontario, 2023