The Power of Collaboration: Machine Learning for Team Performance Optimization

CSE 575 – Statistical Machine Learning

Portfolio Report

Shilpitha Gandla
Robotics and Autonomous Systems –Artificial Intelligence
Arizona State University Tempe, Arizona
1224631798
sgandla2@asu.edu

***Abstract*— This report presents a recommendation system for team member replacement, inspired by the works of Liangyue Li et al.'s "Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation."[1] The system utilizes machine learning techniques, including graph kernels, to suggest optimal team configurations based on skill matching and structural compatibility. The algorithm considers factors such as player performance metrics, playing styles, teamwork compatibility, and injury/sickness history. Experimental evaluations using NBA data demonstrate the system's effectiveness and efficiency, outperforming alternative approaches. The proposed algorithms offer a comprehensive solution for identifying suitable replacements, ensuring team performance and continuity.**

Keywords—team member replacement, recommendation system, machine learning, graph kernels, skill matching, structural compatibility, team performance.

I. Introduction

Efficient and effective team members play a vital role in the success of a team. However, unforeseen events such as illness, resignation, or termination can result in the sudden absence of a crucial team member. This absence can have a significant impact on team performance, potentially leading to project delays or disruptions. In such situations, it becomes crucial for project managers and team leaders to find a competent substitute who can perform at the same level as the departing team member.

The process of finding a suitable replacement can be challenging and time-consuming. Project managers and team leaders need to identify individuals who possess the necessary skills and expertise to fill the vacant role. Additionally, they must consider factors like compatibility with the team, adaptability to the existing work environment, and the ability to meet project requirements.

To streamline this process, there is a growing need for an effective algorithm that can quickly and economically suggest viable replacements for important team members. Such an algorithm would assist project managers and team leaders in efficiently identifying potential candidates who possess the required qualifications and capabilities.

The algorithm should incorporate various elements to ensure its effectiveness. It should consider the specific requirements of the role, such as technical skills, domain knowledge, and relevant experience. Additionally, it should account for the team dynamics and cultural fit by assessing the interpersonal and communication skills of potential candidates. Furthermore, the algorithm should be flexible and adaptable to accommodate different scenarios. It should be capable of handling varying levels of urgency and provide options for both short-term and long-term replacements.

By developing an effective algorithm that combines data analysis, role-specific criteria, and flexibility, project managers and team leaders can significantly reduce the time and effort required to find suitable replacements for important team members. This can ultimately mitigate the negative impact of sudden absences on team performance and ensure the continuity of projects and operations.

II. Problem Statement

To address this problem, we have a worked on a system for which our research is motivated by the works of Liangyue Li et al.'s "Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation" [1] and Theodoros Lappas et al.'s "Finding a Team of Experts in Social Networks." [2] These articles have inspired us to develop a recommendation system that utilizes machine learning techniques to suggest optimal team configurations, maximizing strengths and minimizing weaknesses.

The primary objective of our study is to create a robust recommendation system that assists coaches and team managers in making well-informed decisions regarding their team's composition and tactics. By analyzing various factors and leveraging machine learning approaches, we aim to provide valuable insights into team member selection, substitution strategies for injured or sick athletes, and identifying employees suitable for specific roles within business teams.

We have developed a recommendation system that utilizes NBA statistics data [5] as the basis for analysis. By employing machine learning techniques, we extracted valuable patterns and insights from the vast amounts of available data. These techniques enabled us to identify individual player strengths and weaknesses, team dynamics, and compatibility factors that contribute to successful team performance.

The recommendation system considers a range of factors such as player performance metrics, playing styles, teamwork compatibility, and injury/sickness history. By integrating these variables, the system will be able to generate optimal team configurations that maximize overall performance and minimize potential shortcomings.

Our recommendation system is mainly based on the paper: "Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation" by Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin and Norbou Buchler [1].

A. *Overview of the Study*

The authors propose an algorithm for team member recommendation that combines skill-based matching, social network analysis, and machine learning-based modeling. The algorithm compares the skills of potential replacements to the remaining team members using Voronoi diagrams and similarity measures. It also considers the social connections within the team using a modified version of the PageRank algorithm. Additionally, the algorithm uses graph kernels to model the relationships between team members and potential replacements. The authors evaluate the algorithm on NBA data and demonstrate its superior performance compared to baseline methods. Overall, the algorithm offers a comprehensive solution to the challenge of finding suitable replacements for crucial team members. Figure 1 shows the graph before and after the replacement of the team.

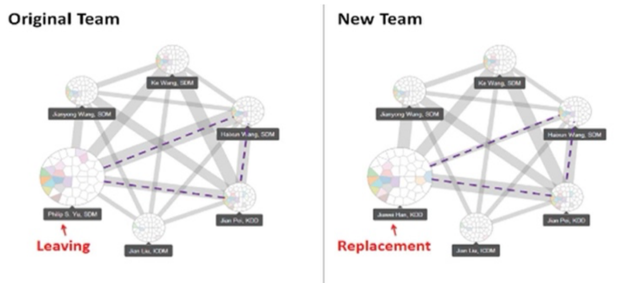

Figure 1: Team graph before and after replacement [1]

B. *Proposed Algorithm*

The algorithm proposed by the authors incorporates three crucial elements: skill-based matching, social network analysis, and a machine learning-based model. The skill-based matching component compares the skill vectors of potential replacements and the remaining team members using a similarity measure based on Voronoi diagrams. This approach ensures that the replacements have similar skills to the departing team member and complement the skills of the existing team members.

The social network analysis component leverages the team's network structure to identify candidates for replacements who are likely to integrate successfully. The authors utilize a modified version of the PageRank algorithm, a popular network analysis algorithm, to compute a measure of each potential replacement's social centrality within the team.

The machine learning-based modeling component utilizes graph kernels to model the relationships between team members and potential replacements. Graph kernels are functions that capture the similarities between graphs. The authors propose several graph kernels that consider various aspects of the relationships, such as skills, positions, and interactions on the court. These kernels enable the computation of a team context-aware similarity measure between the current team and potential replacements.

The algorithm is evaluated on a real-world dataset of NBA players and teams. By comparing it to several baseline methods, the authors demonstrate that their algorithm outperforms the baselines in terms of precision and recall. Additionally, a sensitivity analysis is conducted to showcase how the algorithm's performance varies with different parameter settings.

C. *Proposed Solution*

The paper introduces a similarity measure called the team context-aware similarity measure, which is essential for implementing the proposed algorithm. This measure considers both skill matching and structural matching, as well as their interaction. In mathematical terms, a graph kernel defined on the current and new teams serves as a powerful tool for calculating the team context-aware similarity.

III. ML TECHINIQUES

The paper proposes several machine learning techniques to address the problem of team member recommendation. The key ML techniques discussed in the paper are graph kernels and their application in the context of team context-aware similarity. These techniques aim to capture both skill matching and structural matching, considering the interaction between them.

A. *Graph Kernels*

The authors utilize graph kernels as a tool to measure the similarity between subgraphs in different teams. The idea behind graph kernels is to compare the similarity of sub-graphs between two input graphs and aggregate the results to obtain the overall similarity between the graphs. In the context of team member recommendation, each subgraph represents a specific skill among a subset of team members. By comparing the similarity between two subgraphs, the capability of a new team member to perform a particular sub-task can be assessed. By aggregating these similarities for each task, the overall capability of a potential replacement can be computed. The paper mainly focuses on the use of random walk-based graph kernels, which capture the local structure and connectivity of a graph. These graph kernels effectively model the interactions between individual skills and the team structure, providing a comprehensive framework for team member recommendation.

B. *TeamRep-Basic Method*

The paper introduces the TeamRep-Basic method, which utilizes the random walk-based graph kernel for team member replacement problems. The method involves computing the graph kernel for everyone in the team, which can be computationally expensive, especially for large team sizes. To overcome this challenge, the authors propose two methods: Scale-Up: Candidate Filtering and Speedup Graph Kernel.

C. *Scale-Up: Candidate Filtering*

To reduce the computational complexity, a pruning strategy is employed known as candidate filtering. This strategy removes candidates who do not have any connections with the current team, focusing only on potential replacements who have relevant connections. This pruning strategy significantly reduces the number of graph kernel computations required.

D. *Speedup Graph Kernel.*

The speedup graph kernel algorithms, including TeamRep-Fast-Exact [1] and TeamRep-Fast-Approx [1], aim to address the computation speed challenges. These algorithms leverage effective pruning strategies and approximation techniques to decrease the computational cost. TeamRep-Fast-Exact employs matrix inverse computations and optimization techniques to accelerate the computation of the graph kernel. TeamRep-Fast-Approx uses a sampling-based approach, randomly selecting a subset of random walks instead of considering all possible walks, leading to faster computations with a small approximation error. Figures 2 and 3 show the TeamRep-Fast-Exact and TeamRep-Fast-Approx algorithms.

Algorithm 1: TEAMREP-FAST-EXACT

Input: (1) The entire social network $\mathbf{G} := \{\mathbf{A}, \mathbf{L}\}$, (2) original team members $\mathcal{T}$, (3) person p who will leave the team, (4) starting and ending probability $\mathbf{x}$ and $\mathbf{y}$(be uniform by default), and (5) an integer k (the budget)

Output: Top k candidates to replace person p

Initialize $\mathbf{A}_c, \mathbf{L}_1^{(j)}, \mathbf{L}_2^{(j)}, j = 1, \ldots, l$;
Pre-compute
$\mathbf{Z}^{-1} \leftarrow (\mathbf{I} - c(\sum_{j=1}^{l} \mathbf{L}_1^{(j)} \otimes \mathbf{L}_c^{(j)})(\mathbf{A}_1 \otimes \mathbf{A}_c))^{-1}$;
Set $\mathbf{R} \leftarrow (\sum_{j=1}^{l} \mathbf{L}_1^{(j)} \otimes \mathbf{L}_c^{(j)})\mathbf{x}$; $\mathbf{b} \leftarrow \mathbf{y}^T\mathbf{Z}^{-1}\mathbf{R}$; $\mathbf{l} \leftarrow c\mathbf{y}^T\mathbf{Z}^{-1}$;
for *each candidate q in* $\mathbf{G}$ *after pruning* **do**
  Initialize $\mathbf{s} \leftarrow$ a zero vector of length t except the last element is 1;
  Initialize $\mathbf{w} \leftarrow$ weight vector from q to the new team members;
  Set $\mathbf{E} \leftarrow [\mathbf{w}, \mathbf{s}]$; $\mathbf{F} \leftarrow [\mathbf{s}'; \mathbf{w}']$;
  Set $\mathbf{e}^{(j)} \leftarrow$ a t by 1 zero vector except the last element is 1, for $j = 1, \ldots, d_n$;
  Set $\mathbf{f}^{(j)} \leftarrow$ a $1 \times t$ zero vector except the last element which is label j assignment for q;
  Set $\mathbf{P} \leftarrow [\mathbf{L}_1^{(1)} \otimes \mathbf{e}^{(1)}, \ldots, \mathbf{L}_1^{(l)} \otimes \mathbf{e}^{(l)}]$;
  Set $\mathbf{Q} \leftarrow [\mathbf{I} \otimes \mathbf{f}^{(1)}; \ldots; \mathbf{I} \otimes \mathbf{f}^{(l)}]$;
  Compute $\mathbf{X}_1 \leftarrow (\sum_{j=1}^{l} \mathbf{L}_1^{(j)}\mathbf{A}_1 \otimes \mathbf{L}_c^{(j)}\mathbf{E})$;
  Compute $\mathbf{X}_2 \leftarrow (\sum_{j=1}^{l} \mathbf{L}_1^{(j)}\mathbf{A}_1 \otimes \mathbf{e}^{(j)}\mathbf{f}^{(j)}\mathbf{E})$;
  Compute $\mathbf{Y}_1 \leftarrow \mathbf{Q}(\mathbf{A}_1 \otimes \mathbf{A}_c)$;
  Compute $\mathbf{Y}_2 \leftarrow (\mathbf{I} \otimes \mathbf{F})$;
  Set $\mathbf{X} \leftarrow [\mathbf{P}, \mathbf{X}_1, \mathbf{X}_2]$, $\mathbf{Y} \leftarrow [\mathbf{Y}_1; \mathbf{Y}_2; \mathbf{Y}_2]$;
  Update $\mathbf{M} \leftarrow (\mathbf{I} - c\mathbf{Y}\mathbf{Z}^{-1}\mathbf{X})^{-1}$;
  Compute $\mathbf{r}' \leftarrow \mathbf{Z}^{-1}\mathbf{P}\mathbf{Q}\mathbf{x}$;
  Compute score$(q) = \mathbf{b} + \mathbf{y}^T\mathbf{r}' + \mathbf{l}\mathbf{X}\mathbf{M}\mathbf{Y}(\mathbf{Z}^{-1}\mathbf{R} + \mathbf{r}')$;
end
Return the top k candidates with the highest scores.

Figure 2: TeamRep-Fast-Exact algorithm [1]

Overall, the ML techniques presented in the paper offer a comprehensive framework for team member recommendation. The use of graph kernels enables the assessment of both skill matching and structural matching, providing a holistic view of team dynamics. The proposed methods, such as candidate filtering and speedup graph kernel algorithms, address the computational challenges involved in large-scale team member recommendation tasks. By incorporating these techniques, the algorithm can efficiently and accurately recommend suitable replacements for critical team members, ensuring the continuity and success of a team's performance.

Algorithm 2: TEAMREP-FAST-APPROX

Input: (1) The entire social network $\mathbf{G} := \{\mathbf{A}, \mathbf{L}\}$, (2) original team members $\mathcal{T}$, (3) person p who will leave the team, (4) starting and ending probability $\mathbf{x}$ and $\mathbf{y}$ (be uniform by default), and (5) an integer k (the budget)

Output: Top k candidates to replace person p

Initialize $\mathbf{A}_c, \mathbf{L}_1^{(j)}, \mathbf{L}_2^{(j)}, j = 1, \ldots, l$;
Compute top r eigen-decomposition for $\mathbf{A}_c$: $\mathbf{U}\mathbf{\Lambda}\mathbf{U}' \leftarrow \mathbf{A}_c$;
Set $\mathbf{V} \leftarrow \mathbf{\Lambda}\mathbf{U}'$;
Initialize $\mathbf{s} \leftarrow$ a zero vector of length t except the last element is 1;
Initialize $\mathbf{w}_1 \leftarrow$ weight vector from p to $\mathcal{T}$;
Set $\mathbf{E}_1 \leftarrow [\mathbf{w}_1, \mathbf{s}], \mathbf{F}_1 \leftarrow [\mathbf{s}'; \mathbf{w}_1']$;
Set $\mathbf{X}_1 \leftarrow [\mathbf{U}, \mathbf{E}_1]$, $\mathbf{Y}_1 \leftarrow [\mathbf{V}; \mathbf{F}_1]$;
for *each candidate q in* $\mathbf{G}$ *after pruning* **do**
  Initialize $\mathbf{w}_2 \leftarrow$ weight vector from q to the new team members ;
  Set $\mathbf{E}_2 \leftarrow [\mathbf{w}_2, \mathbf{s}], \mathbf{F}_2 \leftarrow [\mathbf{s}'; \mathbf{w}_2']$;
  Set $\mathbf{X}_2 \leftarrow [\mathbf{U}, \mathbf{E}_2]$, $\mathbf{Y}_2 \leftarrow [\mathbf{V}; \mathbf{F}_2]$;
  Compute $\mathbf{S} \leftarrow \sum_{j=1}^{l} \mathbf{y}_1'\mathbf{L}_1^{(j)}\mathbf{X}_1 \otimes \mathbf{y}_2'\mathbf{L}_2^{(j)}\mathbf{X}_2$;
  Compute $\mathbf{T} \leftarrow \sum_{j=1}^{l} \mathbf{Y}_1\mathbf{L}_1^{(j)}\mathbf{x}_1 \otimes \mathbf{Y}_2\mathbf{L}_2^{(j)}\mathbf{x}_2)$;
  Update $\mathbf{M} \leftarrow (\mathbf{I} - c(\sum_{j=1}^{l} \mathbf{Y}_1\mathbf{L}_1^{j}\mathbf{X}_1 \otimes \mathbf{Y}_2\mathbf{L}_2^{j}\mathbf{X}_2))^{-1}$;
  Set score$(q) = (\sum_{j=1}^{l}(\mathbf{y}_1'\mathbf{L}_1^{(j)}\mathbf{x}_1)(\mathbf{y}_2'\mathbf{L}_2^{(j)}\mathbf{x}_2)) + c\mathbf{S}\mathbf{M}\mathbf{T}$;
end
Return the top k candidates with the highest scores.

Figure 3: TeamRep-Fast-Approx algorithm [1]

IV. RESULTS

The paper conducted experimental evaluations to assess the effectiveness and efficiency of the proposed algorithms for team member replacement. Three datasets, namely DBLP, Movie, and NBA, were used for evaluation purposes.

A. *Effectiveness Results*

Qualitative evaluations included case studies on the datasets, where the algorithm recommended replacements for unavailable team members. The algorithm's ability to consider both skill and structural match was evident, as the authors provided explanations for the suitability of the recommended replacements.

Quantitative evaluations demonstrated that simultaneously considering skill and structural match outperformed considering only one objective. The proposed algorithm consistently outperformed alternative approaches in terms of performance, recall scores, precision scores, and average technique accuracy. Figure 4 compares the performance of the 3 methods. Figure 5 shows the average accuracy as a reflection of budget k.

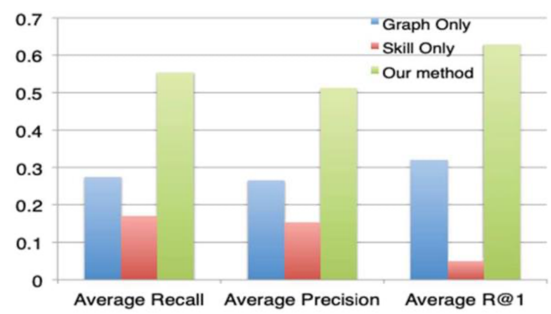

Figure 4: performance of three comparison methods[1]

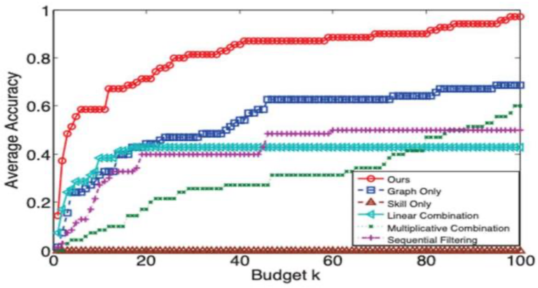

Figure 5: Average technique accuracy as reflection of budget k [1]

B. *Efficiency Results*

The pruning strategy significantly reduced the running time of the TeamRep-Basic algorithm without sacrificing accuracy. Comparisons between TeamRep-Basic and TeamRep-Fast-Exact, as well as Ark-L and TeamRep-Fast-Approx, showed that the proposed algorithms were significantly faster, especially for larger team sizes. Additionally, both algorithms exhibited sub-linear scalability, making them efficient and scalable for large-scale team formation tasks.

In summary, the experimental evaluations confirmed the effectiveness and efficiency of the proposed algorithms for team member replacement. The algorithms excelled in considering skill and structural match, outperforming alternative approaches. They also demonstrated computational efficiency and scalability, making them suitable for various team formation scenarios.

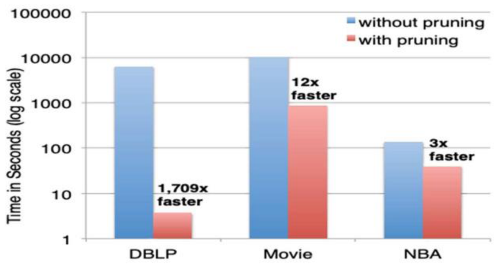

Figure 6: Time comparison before and after pruning

V. Implementation

The NBA dataset [5] was obtained from a website and used for the implementation of the algorithms discussed in the previous paper. A social network was generated using this dataset, where players' names represented the nodes and the number of seasons played together represented the edge weights. The dataset contained statistics of NBA players from 1976 to 2019, including information such as player names, teams played for, positions, ages, and playing status.

In the implementation, two algorithms from the previous paper, namely TeamRep-Fast-Exact and TeamRep-Fast-Approx, were implemented using the NBA dataset. A skill matrix was also generated for all players, with each player having five skills defined, including their primary position (PG, SG, SF, PF, C). Based on the network between players and the skill matrix, the two algorithms were used to identify replacements for a specific player. The algorithms suggested the top five candidates who could potentially replace the player.

To analyze the results, a team of players who had played together was selected. The algorithms were then used to identify replacements for one teammate. The objective was to determine if the algorithms suggested players who had previously played with the remaining set of teammates. The current team array consisted of players such as Tim Duncan, Manu Ginóbili, Tony Parker, Michael Finley, and Matt Boner. Manu Ginobili was chosen as the player to be replaced, and the algorithms were executed to suggest the top five potential replacements.

The results of running the TeamRep-Exact and TeamRep-Approx algorithms provided the top five candidates for replacing Manu Ginobili. The results were evaluated to determine if the suggested players had previously played with the remaining teammates in the team array. Figure 7 shows the screenshot of the NBA Excel data sheet we have used. Figure 8 shows the results after implementation.

1	OKC2018	Álex Abrin	2018	OKC	31	Active	SG	25	5.3
2	PHO2018	Quincy Acy	2018	PHO	10	Active	PF	28	1.7
3	ATL2018	Jaylen Ada	2018	ATL	34	Active	PG	22	3.2
4	OKC2018	Steven Ada	2018	OKC	80	Active	C	25	13.9
5	MIA2018	Bam Adeb	2018	MIA	82	Active	C	21	8.9
6	CLE2018	Deng Adel	2018	CLE	19	Active	SF	21	1.7
7	SAS2018	LaMarcus	2018	SAS	81	Active	C	33	21.3
8	CHI2018	Rawle Alki	2018	CHI	10	Active	SG	21	3.7
9	UTA2018	Grayson Al	2018	UTA	38	Active	SG	23	5.6

Figure 7: NBA Data sheet used [5]

```
Command Window
We need to replace Manu Ginobili ...
Using TEAMREP-FAST-EXACT, the top five candidates are:
Gary Trent
Charles Smith
Ime Udoka
Drew Gooden
David West
time taken    0.2333

Using TEAMREP-FAST-APPROX, the top five candidates are:
Gary Trent
Charles Smith
Ime Udoka
Drew Gooden
David West
time taken    0.0773
fx >>
```

Figure 8: Results after implementing two algorithms.

VI. Individual Contribution

I made significant contributions to this project through my thorough analysis of the suggested publications and projects listed on Canvas. I took the initiative to study these resources and provide a comprehensive analysis of potential project ideas to my teammates. My points were well-received and appreciated by my teammates, demonstrating the value of my contributions in shaping the direction of the project. Furthermore, I took the responsibility of studying the relevant publications related to our project topic, ensuring a deep understanding of the subject matter. I actively shared my views and insights during the project proposal phase, leveraging my research to contribute valuable input that helped shape the overall direction and objectives of our project.

In addition to my contributions in the early stages of the project, I played an instrumental role in writing the final project report. I actively participated in crafting the overview, introduction, ML techniques, experimental evaluations, and results sections of the report. Drawing upon my understanding of the "Fast Algorithms for Team Member Recommendation" paper, I conducted a comparative analysis between the proposed algorithms and contributed valuable insights to highlight the strengths and weaknesses of both models. Furthermore, I actively participated in implementing the fast-approx algorithm, leveraging my technical skills to contribute to the practical implementation aspect of the project. Additionally, I took part in creating the presentation slides, ensuring that our findings and contributions were effectively communicated to our audience.

Overall, my contributions to this project encompassed extensive research, critical analysis, writing, and technical implementation. From shaping the project proposal to actively participating in the implementation and documentation phases, my efforts were geared towards the success and quality of the final project deliverables.

VII. Lessons Learnt

From this project, I have learned several key concepts and gained valuable information. Firstly, I have gained a deep understanding of the importance of efficient and effective team members in achieving successful team outcomes. I have realized that unforeseen events such as illness or resignation can significantly impact team performance and disrupt project timelines.

One of the key concepts I learned was the utilization of graph kernels as a powerful tool for measuring the similarity between subgraphs in different teams. The project explains how graph kernels capture both skill matching and structural matching, providing a holistic view of team dynamics. The various graph kernel techniques discussed, such as random walk-based graph kernels, candidate filtering, and speedup graph kernel algorithms, demonstrated the efficiency and scalability required for large-scale team member recommendation tasks.

Overall, this project has provided me with a comprehensive understanding of team member replacement and the significance of considering skill matching, structural compatibility, and team dynamics. I have also gained practical knowledge of machine learning techniques, particularly graph kernels, and their application in team member recommendation. The insights and lessons learned from this project will undoubtedly contribute to my future endeavors in team management and decision-making.

VIII. Team Members

- Deekshith Reddy Yeruva
- Aasish Tammana
- Sai Charan Raghupatruni
- Sangeetha Ramaswami
- Praveen Sama

Acknowledgment

I would like to express my sincere gratitude to my professor Dr. Moses Boudourides for their guidance, support, and invaluable feedback throughout the development of this report. Their expertise and insightful suggestions have greatly contributed to the quality and clarity of our work. I would also like to extend my appreciation to my dedicated team members for their collaborative efforts and contributions to this project. Their hard work, enthusiasm, and commitment have been instrumental in the successful completion of this report. I am grateful for their valuable input, constructive discussions, and teamwork, which have enhanced the overall quality of our work.

References

[1] Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, and Norbou Buchler. 2015. Replacing the Irreplaceable: Fast Algorithms for Team Member Recommendation. In Proceedings of the 24th International Conference on World Wide Web (WWW '15). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 636-646. https://doi.org/10.1145/2736277.2741132

[2] Theodoros Lappas, Kun Liu, and Evimaria Terzi. 2009. Finding a team of experts in social networks. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '09). Association for Computing Machinery, New York, NY, USA, 467–476. https://doi.org/10.1145/1557019.1557074

[3] E. M. Arkin and R. Hassin. Minimum-diameter covering problems. Networks, 36(3):147–155, 2000.

[4] G. Reich and P. Widmayer. Beyond Steiner's problem: a VLSI oriented generalization. In Proceedings of the fifteenth international workshop on Graph-theoretic concepts in computer science, pages 196–210, 1990

[5] Michael O'Donnell, Social Network Analysis of NBA Teammate Networks https://mikeodonnell.work/portfolio/sna-of-nba/