

EXTRA: Explaining Team Recommendation in Networks

Qinghai Zhou[†], Liangyue Li[†], Nan Cao[‡], Norbou Buchler[‡] and Hanghang Tong[‡]

[†]Arizona State University, [‡]Tongji University, [‡]ARL

[†]{qinghai.zhou, liangyue, hanghang.tong}@asu.edu, [‡]nan.cao@gmail.com, [‡]norbou.buchler.civ@mail.mil

ABSTRACT

State-of-the-art in network science of teams offers effective recommendation methods to answer questions like *who* is the best replacement, *what* is the best team expansion strategy, but lacks intuitive ways to explain *why* the optimization algorithm gives the specific recommendation for a given team optimization scenario. To tackle this problem, we develop an interactive prototype system, EXTRA, as the first step towards addressing such a sense-making challenge, through the lens of the underlying network where teams embed, to explain the team recommendation results. The main advantages are (1) *Algorithm efficacy*: we propose an effective and fast algorithm to explain random walk graph kernel, the central technique for networked team recommendation; (2) *Intuitive visual explanation*: we present intuitive visual analysis of the recommendation results, which can help users better understand the rationality of the underlying team recommendation algorithm.

KEYWORDS

Team Recommendation Explanation; Visualization; Random Walk Graph Kernel

1 INTRODUCTION

The emergence of network science has been significantly changing the landscape of team-based research in recent years. A cornerstone behind team recommendation algorithms for various scenarios (e.g., *team member replacement*, *team expansion* and *team shrinkage*) is random walk graph kernel [2]. For instance, in team member replacement, by applying random walk graph kernel to the team networks before and after replacement, it encodes both the skill match, structure match as well as the interaction between the two during the replacement process [3]. Team member replacement further enables other team recommendation scenarios [4]. However, these existing methods lack intuitive ways to explain *why* the underlying algorithm gives the specific recommendation for a given team optimization scenario.

Purpose and Novelty: In this paper, we present an interactive prototype system, EXTRA, a paradigm shift from *what* to *why*, for the purpose of explaining the networked team recommendation results. The novelties of EXTRA are (1) on the algorithmic side, we propose an effective and efficient algorithm to explain random walk graph kernel, where the key idea is to identify the most influential network

substructures and/or attributes whose removal or perturbation will impact the graph kernel/similarity the most, and (2) on the system side, EXTRA is able to provide informative visual analysis for explaining various team recommendation scenarios from a variety of perspectives (e.g., edges, nodes and attributes). For example, given a candidate for team member replacement, we are able to tell (a) what the key connections are between the candidate and the existing team members, and (b) what the key skills the candidate has that might make him/her a potentially good replacement.

2 OVERVIEW OF THE SYSTEM

In this section, we present (1) the main functionalities of EXTRA to explain three different team recommendation scenarios and (2) key algorithms that support the main functionalities.

2.1 System Functionality

The system provides intuitive visual explanations for the team recommendation results through the lens of the underlying network. Table 1 summarizes the main functionalities, where the rows represent explanations from three different perspectives (e.g., edges, nodes and attributes) for three team recommendation scenarios.

A – Explaining Team Replacement: EXTRA identifies a few key (1) edges (the relationship between the candidate and other team members), (2) nodes (other existing team members) and (3) attributes (the skills of the candidate) that make the candidate and the departure member most similar. In this way, it could help the end-user (e.g., the team leader) understand why the underlying replacement algorithm thinks the given candidate is potentially a good replacement.

B – Explaining Team Expansion: it provides the explanations from the following aspects, including (1) what are the unique new skills s/he brings to the team (i.e., attribute); and (2) what are the key collaborations the new team member might establish (i.e., edges) and with whom (i.e., nodes).

C – Explaining Team Shrinkage: the prototype system flags the *absent* skills and connections with existing team members that makes the candidate most insignificant to the current team. In other words, we want to understand why the dismissal of this particular candidate would impose the least negative impact on the team.

2.2 Overview of Algorithm

A – Graph Kernel for Team Recommendation: random walk graph kernel provides a natural way to measure the similarity between two graphs [1] (e.g., the two team networks before and after a replacement) and turns out to be the core building block behind a variety of team recommendation scenarios. For example, in *team replacement*, the objective is to find a *similar* person m to replace an existing team member r in the context of the team itself. A good replacement m should have a similar skill set as well as a similar collaboration structure with the existing team member. With

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

RecSys '18, October 2–7, 2018, Vancouver, BC, Canada

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-5901-6/18/10.

<https://doi.org/10.1145/3240323.3241610>

	Team Replacement	Team Expansion	Team Shrinkage
Edges	important common collaborations shared by the candidate and the departure member	new collaborations that the new member might establish	the most important but lacking collaborations the candidate should have
Nodes	key existing team members both candidate and departure member collaborate with	key existing team members the new member will work with	key existing team members that the candidate should have collaborated with
Attributes	common and important skills shared by the candidate and the departure member	the unique skills the new team member brings that are critical to the team's new need	the most important skills that the candidate lacks

Table 1: Summary of system functionalities.

graph kernel, we aim to find a candidate that makes the new team most similar to the old team. For details regarding other scenarios (i.e., team expansion and shrinkage), please refer to [2, 4].

B – Explaining Team Recommendation: we seek to understand the team recommendation results via the influence of various graph elements to the corresponding graph kernels. We define the influence score of one specific element as the *rate* of the change in graph kernel $k(\mathbf{G}, \mathbf{G}')$. Given random walk graph kernel for labelled graph [3], the influence of a graph edge W_{ij} can be computed as

$$\mathcal{I}(W_{ij}) = \frac{\partial k(\mathbf{G}, \mathbf{G}')}{\partial W_{ij}} = \mathbf{c} \mathbf{q}_x^T \mathbf{R} \mathbf{L}_x \times (\mathbf{J}^{i,j} + \mathbf{J}^{j,i}) \otimes \mathbf{W}' \mathbf{R} \mathbf{L}_x \mathbf{p}$$

where $\mathbf{R} = (\mathbf{I} - \mathbf{c} \mathbf{W}_x)^{-1}$ and $\mathbf{J}^{i,j}$ is a single-entry matrix with one at its (i, j) -th entry and zeros everywhere else. In the scenario of team member replacement, an ideal candidate should have the key collaborations (with high influence score) as the departure member does, indicating their similarity in terms of how they collaborate with existing members.

The node influence of the i^{th} member is defined as the aggregation of the influence of all the edges incident to this node, i.e., $\mathcal{I}(i) = \sum_{j|(i,j) \in E} \mathcal{I}(W_{ij})$. Existing team members with the highest node influence scores are expected to be key members in the team.

Likewise, to compute the influence of a team member's attributes on the graph kernel, we take the derivative of the graph kernel w.r.t. the member's skill. Denote the k^{th} skill of the i^{th} team member in graph \mathbf{G} as $L_k(i)$, the attribute influence can be calculated as

$$\mathcal{I}(L_k(i)) = \frac{\partial k(\mathbf{G}, \mathbf{G}')}{\partial L_k(i)} = \mathbf{q}_x^T \mathbf{R} \left(\frac{\partial \mathbf{L}_x}{\partial L_k(i)} \right) (\mathbf{I} + \mathbf{c}(\mathbf{W} \otimes \mathbf{W}') \mathbf{R} \mathbf{L}_x) \mathbf{p}_x$$

where $\frac{\partial \mathbf{L}_x}{\partial L_k(i)} = \text{diag}(\mathbf{e}_i) \otimes \text{diag}(\mathbf{L}'(:, k))$ and \mathbf{e}_i is an $n \times 1$ vector with one at the i^{th} entry and zeros everywhere else. Skills with the highest influence scores are the key skills that (1) the candidate and the departure member have in common in team replacement, (2) the new team member might bring in team expansion, and (3) the dismissal member might lack in team shrinkage.

3 SYSTEM DEMONSTRATION

Figure 1 presents the user interface of EXTRA and an example of visualization of influence analysis for team member replacement on a co-authorship network¹, which suggests that Dr. *Jiawei Han* is the best replacement. The influence scores of all edges for both graphs are calculated by our proposed algorithm in Section 2.2 and the width of edge is proportional to the influence score. The top-4 most influential edges are those that connect Dr. *Yu* with the 4 existing members, *Lin*, *Zhao*, *Chen* and *Sun*, all of whom are considered as the key members and they also have strong collaborations with Dr.

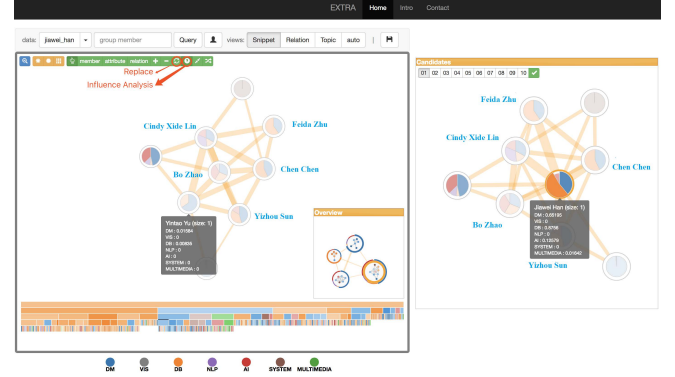


Figure 1: An illustrative example of influence analysis.

Han. After replacement, the top-5 most influential edges with Dr. *Han* overlap with those of Dr. *Yu* in the same ranking order. The only exception is that no edge exists between Dr. *Yu* and Dr. *Zhu* because there is no prior collaboration between them. In addition, the system can provide explanations from the attribute perspective. The key skills (represented in pie chart) shared by Dr. *Han* and Dr. *Yu* are *databases* and *data mining* in this case, which makes the team replacement recommendation more understandable.

4 CONCLUSIONS AND FUTURE WORK

In this demo, we present an interactive prototype system, EXTRA, as the first step towards explaining team recommendation through the lens of the underlying network where teams embed. The system can provide intuitive visual explanations from different perspectives for various team recommendation scenarios. Future work includes applying the proposed algorithm to other networked recommendation scenarios, e.g., user-product networks to explain why a certain product is recommended to a given user.

ACKNOWLEDGMENTS

This work is supported by NSF (IIS-1651203, IIS-1715385, CNS-1629888 and IIS-1743040), DTRA (HDTRA1-16-0017), ARO (W911NF-16-1-0168), DHS (2017-ST-061-QA0001) and NSFC (61602306).

REFERENCES

- [1] Karsten M Borgwardt, Cheng Soon Ong, Stefan Schöner, SVN Vishwanathan, Alex J Smola, and Hans-Peter Kriegel. 2005. Protein function prediction via graph kernels. *Bioinformatics* 21, suppl_1 (2005), i47–i56.
- [2] Karsten M Borgwardt, Nicol N Schraudolph, and SVN Vishwanathan. 2007. Fast computation of graph kernels. In *NIPS*. 1449–1456.
- [3] Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, and Norboub Buchler. 2015. Replacing the irreplaceable: Fast algorithms for team member recommendation. In *WWW*. International World Wide Web Conferences Steering Committee, 636–646.
- [4] Liangyue Li, Hanghang Tong, Nan Cao, Kate Ehrlich, Yu-Ru Lin, and Norboub Buchler. 2017. Enhancing team composition in professional networks: Problem definitions and fast solutions. *TKDE* 29, 3 (2017), 613–626.

¹A demo video of the system is available at <https://youtu.be/D4gcI-QHtPs>. System website: <http://144.202.123.224/system.html>.