AN APPROACH FOR FORMATING TEAM ON SOCIAL NETWORKS CONSIDERING STATUS RELATIONS

George Christodoulou
Computer Science & Engineering
University of Ioannina
gchristodoulou@cse.uoi.gr

Aristidis Chronarakis
Computer Science & Engineering
University of Ioannina
achronarakis@cse.uoi.gr

ABSTRACT

A model for recommending a team formation with required skills and good collaboration between team members from a social network of people connected with status relations is vital. The main target of our work is to propose an algorithm for forming teams which fulfill both the above requirements: (1) cover the defined skills set (2) and the status relations between team members respect the status theory. Together with the proposition of this algorithm, we provide an evaluation of this, based on experiments with a number of datasets, as well some ideas about the future work on this problem.

KEYWORDS

Social Network, Team Formation, Status relations.

1 INTRODUCTION

A common issue on social network is the formation of teams with people of this network who cover a predefined set of skills. A common problem of this approach of the team formation problem is that the selected people who form the team may not collaborate effectively together. In order to deal with this problem we can gain information about the status of each person on this network to form the team. Thus, we consider the problem of designing teams in a social network of people with diverse skills and taking advantage of their status. The design of a team like this not only ensures that the members meet the skill criteria, but also can communicate and effectively work together. The set of skills is known in advance and must be covered by the members in the team. The requirement about the good collaboration can be derived by the status relations between the team members. A status relation between two people indicate who has the upper or the lower status between the two. This status relation is common because they agree on it. The team that will be created must contain members with status relations that will not lead to conflicts, making the team incompatible. For example, if we have formed a decision team in which person A has upper status than person B, person B has upper status than person C and person C has upper status than person A, then there is not someone who can take the final decision as the person with the dominant status and the team is incompatible. If the created team has status relations between them that ensure that the team is compatible then we claim that the team respects the status theory. We propose an algorithm which finds a team with the requested set of skills and also respects the status theory. Worth noting is that our implemented algorithm does not handle the status relations as transitive information. Using different datasets we did an experimental analysis of our algorithm an propose some ideas about future work.

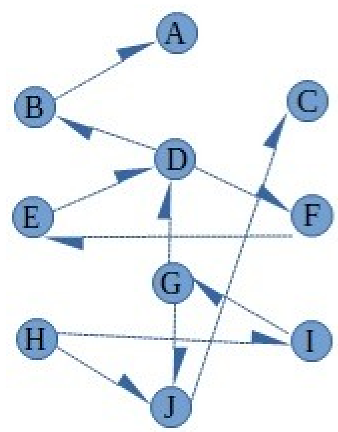

Figure 1: A Social Network in which each node has a set of skills and each edge refers to a status relation between the two nodes.

2 PROBLEM & ALGORITHM DEFINITION

2.1 TEAM FORMATION WITH STATUS AS A GRAPH THEORY PROBLEM

The need for team formation with considering status relations on social networks can be mapped to a graph theory problem. In this problem the social network's people are the nodes of a directed graph where the edges represent the status relations between the people. If a node has a directed edge to an other node, the person which is represented by the first node considers the second person as a person with upper status. On the directed graph each node has one or more skills of the whole skill set. We can see a graph like this in Figure 1. Our desire is to find nodes (people) in this graph to complete a team, in which the team members (nodes/people) respect the status theory and their skills cover the needed skills set. The requirement for respecting the status theory, is to avoid status relations between nodes that create conflicts. This is fulfilled by avoiding the cycles that the suggested team members may create. In other words a team that fulfills our requirements is a team which covers the needed skills set and the corresponding graph does not contain cycles.

2.2 PROBLEM'S FORMAL DEFINITION

In a formal definition of the problem we define as $S=S_i, i=1,2,\ldots,s$ all the skills that are available. The team requires a subset RS of the n available skills. We consider a set of people $P=P_i, i=1,2,\ldots,p$. Each person possesses a subset $K=K_i, i=1,2,\ldots,k$ of the skills. The initiate set of the skills in the team will be $G_0=\{\}$ and the desired set of skills in the team is defined as $RS=RS_i, i=1,2,\ldots,n$. So, when the P_i person is added to the team, the skills set of the team will be $K\cup G$. In other words, a skill is added to the team coverage if there exists at least one member who possesses that skill. We aim to design a team that covers the specified skills while accomplishing a non-conflicting status pattern inside the team. We define status as S=(+,-), where (+) means 'upper status' and (-) means 'lower status'. If an edge with (+) exists, starting from a node A and pointing to a node B, it means that node A has upper status than B. Because of the agreement on the status between two nodes, a pointing edge with (-) from a node X to a node Y is equal with a pointing edge with (+) from Y to X. In order to, make a compatible team, there must not be any conflict of status between the members. A team is incompatible when status edges can form a cycle. This leads to undefined status structure inside the team. Concluding, a team consists of a set of nodes $X=X_i, i=1,2,\ldots,t$ and is considered acceptable if: (1) The skills of the nodes-members form a superset of the skills set required. (2) The status edges between nodes-members do not form a cycle.

2.3 PROPOSED ALGORITHM

We present an algorithm which takes as input the graph (social network with status relations and skills), and returns a set of nodes (team) as output which satisfies the requirements. The input graph has both directed edges with upper (+) or lower (-) status relations. Because of the fact that the nodes agree to the status relation between the all the edges with the lower (-) status are converted to upper (+) edges of the opposite direction. The algorithm is based on a traversal approach on the given graph. Firstly, we remove nodes from the graph with skills that don't intersect with the needed skills set. If a node hasn't skills in common with the needed skills set, we don't benefit from the addition of this node in the team. We also find the rarest skill from the needed skills set. We use nodes with this skill as starting nodes on graph traversal because a node with the rarest skill will always be a member of the team and may be discovered difficultly, if we start the traversal from a random node. Our traversal is similar to the BFS traversal. Starting from one of the nodes with the rarest skill we add him to the team and then we explore his neighbors with incoming and outcoming edges. A neighbor and his skills are added to the team and the team skills set, if the addition of him in the team does not create cycles in the graph, constructed using the edges (status relations) between team members.

Algorithm 1 Team formation with status

Input : Social Network with Status Relations as SN
Requested Skillset as RS

1.Remove from SN nodes which $nodeSkills \cap RS = \emptyset$
2.Find from SN nodes which have the rarest skill from RS as RSN
3.While $TeamSkills \subset RS$ and $RSN \neq \emptyset$
Select node N where $N \in RSN$ as starting node
Traverse nodes on SN like BFS starting from N.
If the addition of new explored node X to Team respects Status Theory
then ADD skills of X to TeamSkills and X to Team
4.Remove from Team nodes which do not contribute unique skills to the Team only if they are not a bridge to nodes with unique skills.

Output : Team that covers the RS and respects the Status Theory

After the exploration of the neighbors of the node with the rarest skill we explore the next level like BFS, the neighbors of the neighbors of starter node. On this point, we have to make an observation. Nodes can be added to the team while they contribute skills that are already acquired. This leads to the second phase of the algorithm in which we check the formed team for nodes which when removed (1) the required skills set is still covered (2) the team remains a connected component. With this check we aim to remove nodes from the team that do not contribute unique skills on team skills set, only if they are not a bridge to nodes with needed skills. This algorithm terminates when the team skills set matches the desired one or when we can not find an acceptable team using different starting nodes.

3 EXPERIMENTAL ANALYSIS

Overview

In order to check if our proposed approach for team formation with status works, we implement it. We made our implementation using Python language and NetworkX library [1] for the operations on graphs. The full code of the proposed algorithm along with the input datasets which are presented in **§ 3.2** can be found on [2] and [3].

3.1 Metrics

We used 2 different characteristics to check the output teams of our algorithm on two datasets: (1) the diameter of the team (2) the bridge nodes taking part in the team. We seek the right balance between spreading the status respect and keeping it effective. Thus, the diameter must be kept short. Also, the bridge nodes are used in order to keep a vital path between nodes that are needed in the team. Although, if there is a big amount of bridge nodes in the team, they would lose their effectiveness. The role of the bridge nodes in the team is assistive and are needed only for connecting nodes with important skills.

3.2 Datasets

We used two social network datasets to test our methods. Both are provided as part of the Stanford Large Network Dataset Collection. We used also a skill-oriented dataset from Kaggle Datasets. The networks we used have both positive and negative edges. We used data taken from a cryptocurrency network and football statistics to create the first dataset for our experiments. The cryptocurrency dataset represents the Bitcoin Alpha web of trust network. It is a directed and signed network based on trust relations among people in the network. The dataset consists of 3.783 nodes and 24.186 trust connections between them. The second dataset is created from the official FIFA rankings on players in 2017. FIFA gives a ranking to all the players for all their skills based on their real life performances. The dataset consists of more than 17.000 players with their skills. It is created to be used in the FIFA 18 game. So, the final synthetic dataset is a combination of two real datasets. The second synthetic dataset is based on a network of trust relations among users at a technology news site named Slashdot. At Slashdot, users can rate each other as friend or foe. These ratings are converted to negative and positive links between users. The dataset contains 77.357 nodes and 516.575 edges. The second synthetic dataset was made by assigning skills from the FIFA dataset to the Slashdot users.

Dataset	Nodes	Edges	Comments
Fifa	17.000	-	Football statistics
Slashdot	77.357	516.575	User ratings
Alpha Bitcoin	3.783	24.186	Bitcoin web of trust

Table 1: Dataset details.

3.3 Results

For each run of our algorithm we used different demands on skills needed to form a team. Also we differentiated the rareness of skills existence among users. We accomplished that by tuning the percentage of a skill needed to confirm that a user owns a particular skill. For example, if 75% of a skill is needed to confirm that a user possesses it, there will be more users possessing the same skill than if 85% is needed to confirm the possession of a skill.

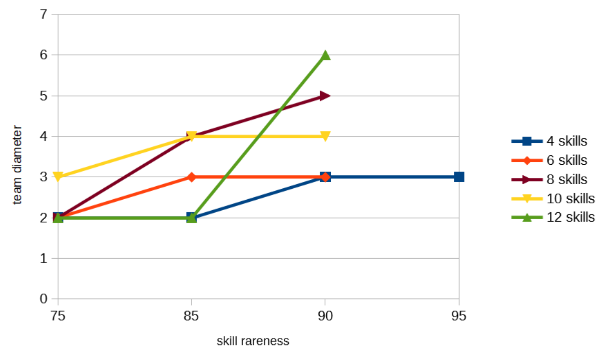

Figure 2a: The team diameter of teams depending on skills demands and rareness.(Slashdot dataset)

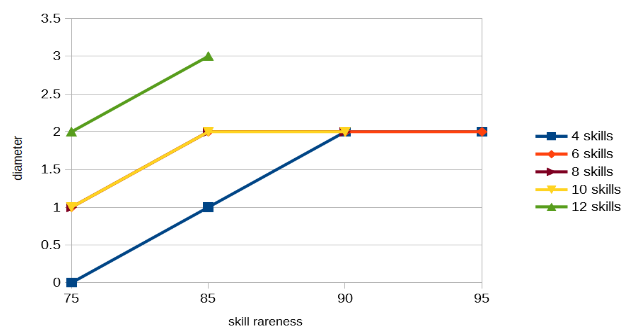

Figure 2b: The team diameter of teams depending on skills demands and rareness.(Alpha Bitcoin dataset)

In figure 2 we see the evolution of the team diameter. In figure 2a, we applied the algorithm to the Slashdot dataset. The diameter of the team raises as the rareness of a skill evolves but remains stable. In figure 2b, the results of the Alpha-Bitcoin network leads us to the same conclusion. The team diameter is raising again with the evolution of skill demanding and rareness. The higher value of the diameters differs depending on the dataset, although in general the diameter remains in low levels. The difference is caused by the nature of the Slashdot dataset which is larger and more complicated than the Alpha-Bitcoin. In conclusion, we can describe the output-teams as teams with quality because the diameter is short, so the status respect is kept effective and direct.

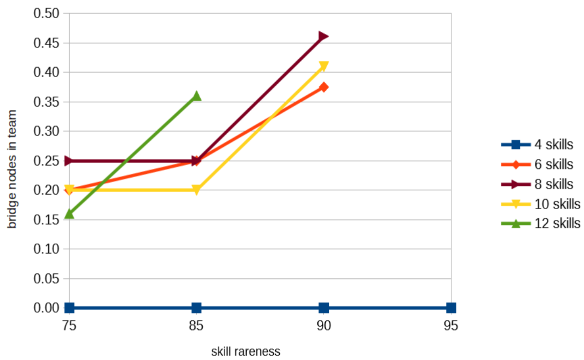

Figure 3a: The bridge nodes percentage of teams depending on skills demands and rareness.(Slashdot dataset)

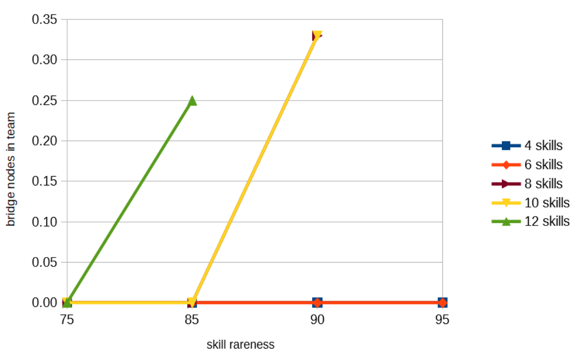

Figure 3b: The bridge nodes percentage of teams depending on skills demands and rareness.(Alpha-Bitcoin dataset)

In figure 3 we observe the percentage of bridge nodes needed to form teams. Bridge nodes are important in joining nodes that have critical skills to complete the demanding skillset. Also, the bridge nodes have skills needed in the team, so they remain relevant. In figure 3a, our algorithm is applied on the Slashdot dataset. The dataset has clearly users with critical skills that are not directly connected so the bridge nodes are needed to construct a team. In figure 3b, the bridge nodes are used less in the Alpha-Bitcoin dataset, because of the direct status connections. The percentage of bridge nodes is growing -as expected - as the skill demands and rareness evolve. In general, the participation of the bridge nodes is small even in strict conditions like high skill demands and rare skill existence in the dataset. In conclusion, the bridge nodes constitute an economic cornerstone in team construction.

FUTURE WORK

Currently, we consider a generic threshold for every skill to approve the possession of it by a node. In the future, we can add exclusive thresholds for the skills that depend on the requirements of the team designer. For example, the team can require excellent knowledge of a certain skill (high threshold) and intermediate knowledge of another skill (low threshold).

CONCLUSION

In this paper, we contributed a new algorithm for designing teams with respect to status relations between the members. We approached the problem in a greedy way so that a team will be created even in difficult situations where users with the compatible skills are not directly respecting each other status. The teams are not long in terms of diameters and the bridge-nodes are taking a small-but critical- part in the team. We tested our method with two different synthetic datasets ,composed with real data. Also, we stretch-tested the algorithm with high skill rareness and skill demands. Finally, we contributed a good algorithm in terms of effectiveness by choosing to use bridge-nodes in teams when needed.

REFERENCES

[1] NetworkX https://networkx.github.io/

[2] Kaggle Datasets https://www.kaggle.com/datasets

[3] Stanford Datasets https://snap.stanford.edu/data/

[4] T. DuBois, J. Golbeck and A. Srinivasan, "Predicting Trust and Distrust in Social Networks," *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, Boston, MA, 2011, pp. 418-424.