Graph Mining to Characterize Competition for Employment

ABSTRACT

In this paper, we discuss a novel application of graph analytics to characterize competition in the workforce. We propose a methodology that relies on finding communities in a graph representing prospective employees (with edges connecting people who interviewed for the same job) and communities in a graph representing available jobs (with edges connecting jobs that interviewed the same person). We then apply the proposed methodology to a real dataset corresponding to co-operative internships offered to undergraduate students at a North American post-secondary institution, illustrating the benefits of our approach.

ACM Reference format:

. 2017. Graph Mining to Characterize Competition for Employment. In *Proceedings of ACM Network Data Analytics conference, Raleigh, North Carolina USA, May 2017 (NDA'17),* 7 pages.

DOI:

1 INTRODUCTION

Many applications involve relationships that can naturally be expressed as graphs: friend/follower relationships in social media, hyperlink relationships in the World Wide Web, chemical structure and protein interactions in bioinformatics, etc. Graph mining techniques such as clustering and community detection can identify interesting structure in such relationships.

We discuss a novel application of graph mining in the context of the workforce to understand competition for employment. This is an increasingly important application domain: the job market has become very competitive due to forces such as globalization and technological change [1]. As a result, employers are becoming more data-driven in their hiring processes [3]. Current data analysis efforts focus on measuring time to hire, hiring manager satisfaction, and performance of a hire [6], as well as measuring the effectiveness of employee benefits programs [3]. However, there remain critical gaps in the job market that employers alone cannot tackle. For example, employers may not have a good understanding of the available talent pool and may not be allocating their recruiting resources effectively. Likewise, employees may not be aware of the extent of competition for various types of jobs and therefore they may not know which jobs are realistically within their reach.

LinkedIn is perhaps the most common example of the potential of graph analytics in the context of the job market: its users establish connections to other users, thereby creating professional networks whose properties have been investigated [9, 10]. LinkedIn data have been used to develop graph-based skill extraction algorithms that are useful for hiring [7] and have inspired link recommendation algorithms [11]. Furthermore, a natural example of network effects is employee referrals, which have been shown to generate higher quality hires with better retention and which LinkedIn offers as a service [6].

In this paper, we go a step further and perform an end-to-end analysis of an entire job sub-market. Our analysis is enabled by a unique dataset from a North American post-secondary institution, corresponding to 4100 undergraduate students competing for 2000 co-operative internships. The dataset includes information about every job interview and hiring decision that took place during the summer 2016 season. The questions we want to answer include:

(1) Are there natural clusters of employers and employees, and if so, what are the defining characteristics of each cluster?
(2) Can we rank employees and jobs into tiers based on the corresponding networks?
(3) Which employers attract good prospective employees and which prospective employees obtain interviews for sought-after positions?

To answer these questions, we propose a graph-oriented methodology to characterize workforce competition. Our methodology is based on finding communities in graphs induced by the job market. We construct an employee graph by connecting two employees if they interview for at least one job in common. Similarly, we construct a job graph by connecting two jobs that interview at least one employee in common. We then perform community detection on both graphs, describe each community using its unique characteristics, and identify relationships among the community structures in the two graphs.

To summarize, we make the following three contributions:

(1) We apply graph mining to a novel application domain of employment/workforce.
(2) We propose a methodology to characterize competition for jobs and for prospective employees using graph mining (using community detection in the corresponding networks).
(3) We apply our methodology to a unique dataset consisting of undergraduate students as prospective employees and employers offering co-operative internships. We show that our techniques lead to actionable insight for the benefit of employees and employers.

The remainder of this paper is organized as follows. We discuss related work in Section 2; we describe our dataset in Section 3; we present our methodology in Section 4 followed by our results in Section 5; and we conclude in Section 6.

2 RELATED WORK

This paper is related to two bodies of work: graph mining and workforce/employment analytics. In the context of graph mining, there are several standard analysis techniques such as clustering and community detection, which have been widely successful in

NDA'17, Raleigh, North Carolina USA

DOI:

Table 1: Job and student data

Table	Attributes
Students	Student ID, Academic Year, Academic Group
Jobs	Job ID, Job Title, Employer, Location, Industry
Interviews	Job ID, Student ID, Hired

understanding complex networks [4]. We also use standard analysis techniques but we apply them to a novel application domain.

In the context of employment data analytics, we are not aware of any similar work since it is difficult to obtain a dataset of interviews across many competing employees and employers. Social media services like LinkedIn are becoming increasingly used by employers for recruiting and by employees for job-hunting, and the importance of network effects such as employee referrals are becoming increasingly important to employers [6]. Employers recognize the importance of understanding competition, and there has been prior work on analyzing the LinkedIn graph [7, 9–11]. However, we are not aware of any previous work on characterizing competition among employees and employers in a large job market.

Finally, we remark that there has been prior work on mining co-operative employment data [reference omitted for double-blind review], but competition in the co-operative workplace has not been studied at the employee and employer level as we do in this paper.

3 DATA

The dataset used for this analysis consists of 4100 undergraduate students from a North American post-secondary institution. These students competed for nearly 2000 jobs from 700 distinct employers over a two-month period in summer 2016. On average, each student had 3.5 interviews, and each job interviewed 7 students. Some jobs hired multiple students, as discussed later. Approximately 50 percent of students were hired by some employer and 75 percent of jobs were filled with at least one student. This is indicative of a highly competitive environment.

The information we have for students and jobs is summarized in Table 1, with primary keys underlined. We have the year of study (first year through fourth year) and academic group of each student. For each job opening, we have the job title, employer name, location, and an industry label which indicates whether the job is related to Information Technology (IT). The industry label was manually assigned by the data providers and contained incorrect and missing values, which we will treat through community detection.

For each interview, we have information about the job, the participating student, and whether the student was hired for the job.

Hiring statistics for the four academic groups in the dataset are shown in Table 2. For each group, we show the total number of students, the number of students hired, and the percentage of students hired. Table 3 shows the same information for each academic year. In both tables, there is no obvious group of students that are in highest demand, although senior students (year 3 and especially 4) are more likely to be hired than junior students (years 1 and 2).

Some student records were missing the academic year and/or the academic group, which explains why the total number of students in Table 2 and Table 3 does not sum up to 4100.

Table 2: Student academic group sizes and hire rates

Academic Group	Students	Students Hired	Hired (%)
Arts	300	155	52
Computing	900	470	52
Engineering	1575	785	50
Sciences	525	240	46

Table 3: Student academic year sizes and hire rates

Academic Year	Students	Students Hired	Hired (%)
1	375	180	48
2	1450	695	48
3	975	495	51
4	675	370	55

Table 4: Most common job types and hire rates

Job Type	Jobs	Students Hired
Software Engineering	575	890
Analytics	225	270
Finance	125	190
Manufacturing	125	180
Education	100	170
Administrative	125	160
User Design	75	90
Civil Engineering	50	70
Medicine	50	70
Accounting	25	50

Several jobs interviewed many more than the average number of students interviewed per job (7) and hired multiple students for the same position. The implications of this will be illustrated in Section 5 when examining the communities. Table 4 shows common job types, with the total number of jobs and students hired across these types tabulated. Some job types overlapped, such as Software Engineering and Analytics.

4 METHODOLOGY

Algorithm 1 summarizes our graph mining methodology. We begin by constructing job and employee graphs. Edges are undirected and unweighted in both graphs. The job graph is formed by connecting jobs that interviewed at least one student in common. The student graph is formed by connecting students who have at least one interview in common.

Table 5 is an example set of interviews across 9 students (labeled 1-9) and 8 jobs (labeled A-H). The corresponding job and student networks are illustrated in Figures 1 and 2 respectively; we will discuss the communities shortly.

With a job and student graph in hand, we run community detection on both graphs. We use the Louvain Method, implemented in the Networkx Python package [5], which aims to optimize connections within communities while minimizing connections between

ALGORITHM 1: A graph mining methodology to analyze the job and student graphs

Data: Student Graph S and Job Graph G
Result: A labeled set of student and job communities

(1) Perform community detection on S and G separately;
(2) Using domain knowledge, rank the job communities in G into tiers: top, mid, low, and other;
(3) Label students in S who were interviewed (or hired) with each tier of job;
(4) Describe the unlabeled student communities based on the state of their nodes as determined by (3).

Table 5: Example table of interviews

Job ID	Student IDs
A	1, 2
B	1, 2
C	2
D	1, 3, 4, 5
E	5, 6
F	6, 7, 8, 9
G	7, 8, 9
H	7

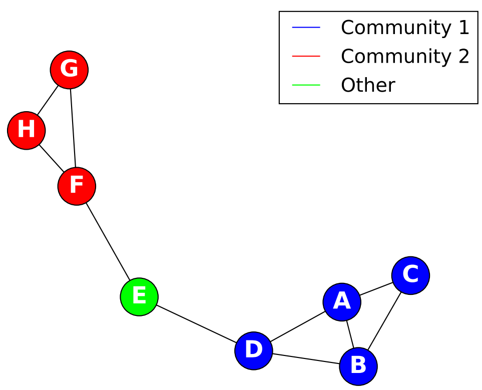

Figure 1: A job network based on the data from Table 5, colored by job community

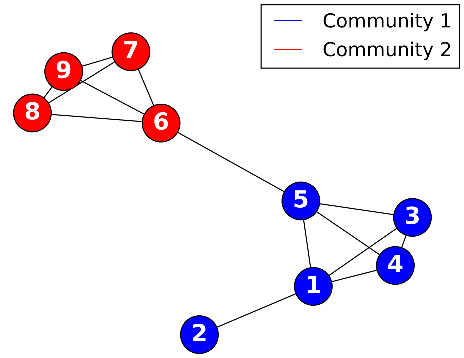

Figure 2: A student network based on the data from Table 5, colored by the job communities from Figure 1

communities [2, 8]. We then inspect the characteristics of students and jobs in each community to identify representative features. This step helps handle missing data and noisy signals since students and jobs that are similar tend to compete with each other. For example, we observed that a community of jobs labeled as Information Technology jobs were actually oriented around design and project management (details in Section 5).

The community features of the job network were easily interpretable given our domain knowledge of co-operative education. To scope down the manual labeling process, we focused on the largest industry in the dataset: Information Technology (IT). As seen in Table 4, the most common job types were software engineering and analytics. Given the institution's specialty in computing, internships offered by IT companies in California and Seattle were deemed the "best". Such companies include Amazon, Apple, Facebook, Google, LinkedIn, Microsoft, Snapchat, Uber and Yelp. As it turns out, these employers competed in a single community of top jobs, as discussed in the next section.

This domain knowledge-driven labeling step made it possible to rank the communities of jobs by their quality. In return, this new information was used to categorize communities in the student network. This was done by inspecting the distribution of job quality that students in a given community were competing (interviewing) for. Labeling can also be done based on where students were hired, although analyzing either distribution yielded similar results.

Recall Figures 1 and 2. The example job network is colored based on communities detected. The communities in the example student network are then colored based on the job communities in which the students had the most interviews. For example, student community 1, containing students 1 through 5, is colored blue because these students interviewed for jobs in job community 1 which is colored blue.

To zoom in on important nodes in each community, we extend our approach by performing centrality analysis in the job network. Centrality can be used to identify the most important nodes in a graph. In our case, central jobs are typically multi-disciplinary roles and interview a diverse set of students. While communities allow us to characterize competing groups of jobs and students, looking at central nodes gives us a more concrete view into a community. We find that central nodes complement community detection by acting as representatives for their communities. This provides us a chance to validate the results of the Louvain Method which is influenced highly by central nodes.

Table 6: Summary of the job communities

Community	Community Type	All Jobs	Edges	Density (%)	Diameter	Avg. Degree	IT Jobs (%)
Full Network		1970	25700	1.3	6	26	29
A	Physical Engineering	390	3325	4.4	6	17	12
B	Mid Tier IT	365	4150	6.3	5	23	60
C	Finance	280	2550	6.4	7	18	9
D	Research & Teaching	280	2225	5.6	6	16	9
E	Top Tier IT	240	4200	14.5	4	35	60
F	Civil Engineering	200	1550	7.6	6	15	0
G	Low Tier IT	185	1425	8.4	5	15	42
H	User Experience	30	150	36.5	4	10	0

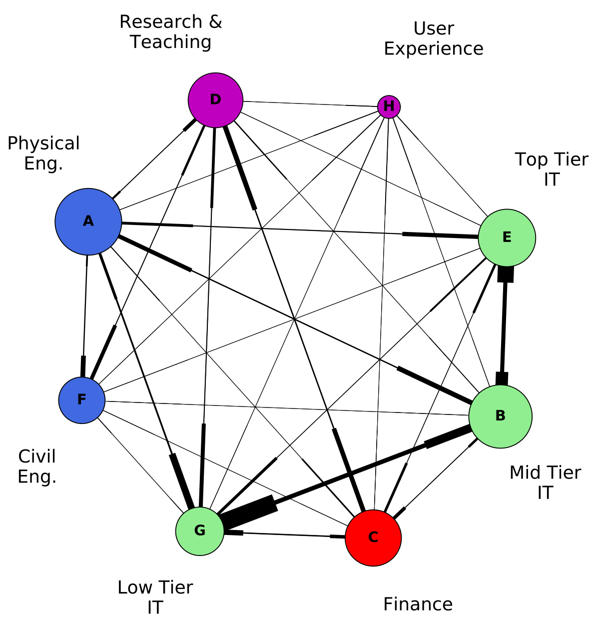

Figure 3: The labeled job community network

5 RESULTS AND DISCUSSION

5.1 Job Communities

In the job network, the Louvain Method found eight communities, three of which we labeled "IT communities" based on manual judgment of their jobs. IT was the only discipline with several communities, which naturally invites us to rank them. Table 6 summarizes properties of each community, including the percentage of jobs in IT, and our manual label for each community. The community network, where nodes are the communities detected, is shown in Figure 3. The thickness of an edge visualizes the overlap of students interviewed between communities (in a directed way), as discussed in Section 5.1.4. Node sizes correspond to the number of jobs in a community.

5.1.1 ***Job Network Properties.*** With the exception of Communities H and E, it is not obvious how the communities differ from each other based on their network properties shown in Table 6. Community H is small and is mostly composed of a single job type, as discussed later. Community E is dense, has the lowest diameter for its size, and has a high average degree per node. This establishes Community E as the most competitive community.

5.1.2 ***Job Community Labeling.*** As it turns out, Community E is special: it contains all of the best IT jobs at top companies such as Amazon, Bloomberg and Facebook. A large majority of students interviewing for these jobs were in their fourth (final) year of study, with the remaining students mostly in their third year of study. A subgraph containing the top 20 employers with the most interviews from this community is shown in Figure 4. This subgraph is a near-clique, with a density of 89% and an average degree of 17.

We ranked Community B second in IT (Mid Tier). It contained small and often local IT companies who interviewed more junior students, mostly in their second year. The final IT community, Community G, had mostly quality assurance and software testing roles, which is perceived by students as less desirable work (Low Tier). Most students competing for these jobs were in their first year and had little prior work experience.

Community H exclusively contained user experience roles. Community A had primarily mechanical engineering, manufacturing and hardware jobs, which we label as "Physical Engineering". Community C was primarily composed of financial firms. Although Communities A and C were non-IT communities, their most desirable jobs were IT-like. For example, the most competitive roles in the finance community were in data science.

5.1.3 ***Job Centrality Analysis.*** To zoom into each community, we extracted the top ten nodes with the highest closeness centrality (that is, the jobs with the smallest average shortest path length to other jobs). In the finance community, central nodes were data analyst jobs at a consulting firm and bank trading floor. In the physical engineering community, the central node was a manufacturing job from a large automotive company. In the top IT community, the data scientist role at Facebook was the most central role. These central jobs are interdisciplinary, competing with many types of jobs and interviewing a diversity of students such as younger students. An advantage of this in co-operative education is that it enables students to change industries, say from software engineering to data science. Furthermore, finding central jobs allows us to explain interesting phenomena, such as lower-year students competing for top jobs.

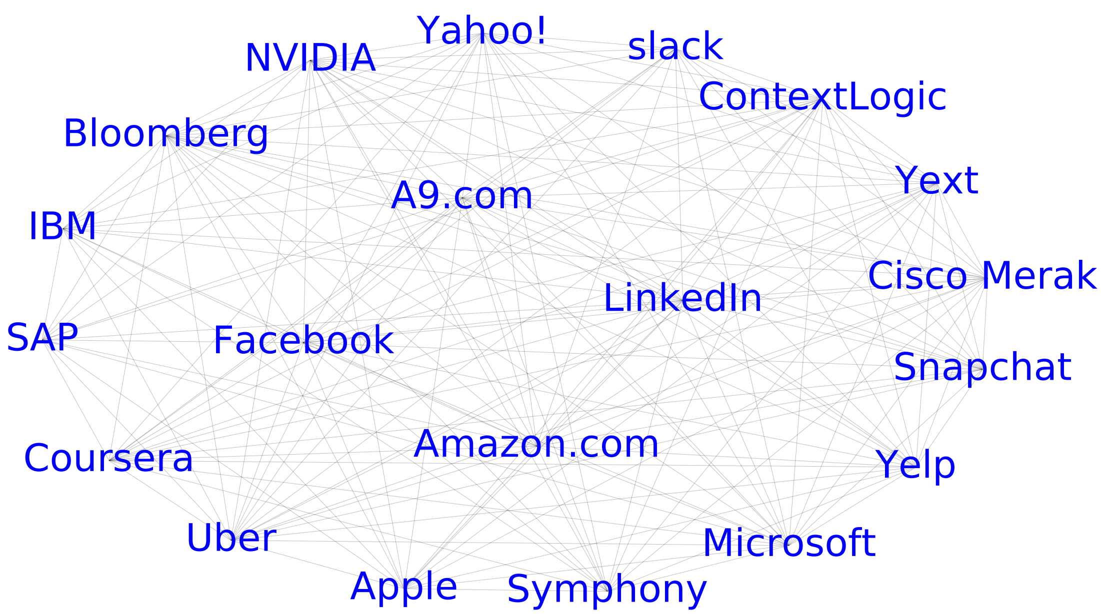

Figure 4: The top 20 employers in the top tier IT community by most interviews

5.1.4 ***Job Cross-Community Competition.*** The job community network (Figure 3) is a clique, which was not expected due to the large differences between communities and the exclusivity (discussed in greater detail in Section 5.2) of top tier IT jobs. Defining S() as the set of students who interviewed for at least one job in a given job community, we calculate the directed edge weight between communities X and Y as the conditional probability of a student interviewing for a job in X given that he or she interviewed for a job in Y.

$$Weight(X, Y) = |S(X) \cap S(Y)|/|S(Y)|.$$

In Figure 3, the edge weights are visualized, with thicker lines corresponding to larger weights.[1] We find that the low tier IT community interviewed many students who are also seeking jobs from other communities, especially from the mid tier IT community. Likewise, there are connections between the top and mid tier IT communities, where top students could be applying to mid tier jobs as a safety net. Validating our ranking, there are few connections between the top and low tier IT communities.

5.1.5 ***Job Outlier Detection.*** We now zoom into the top-tier IT community and inspect top jobs that did not hire any student.

The students these jobs interviewed were hired by other top jobs. Manual inspection of these unfilled "top" jobs showed that they were outliers: they were interesting enough to attract students, but their actual quality was lower. It was common to find IT start-ups competing against the top jobs from large employers in California. Thus start-ups are not fully aware of the extent of competition, creating missed opportunities for hiring. In addition, manual extraction of job descriptions showed that some of these jobs exaggerated the role a student would be hired for (e.g., "Big Data Hacker").

[1] The edges leaving Community H were made thin since many students who interviewed for these jobs also interviewed elsewhere due to its small size.

Table 7: Hiring statistics of the different job communities

Community	Interviews per Job	Students Hired per Job
Full Network	7	1.10
A	7	1.12
B	7	1.08
C	6	1.09
D	6	0.90
E	11	1.43
F	6	1.18
G	6	0.95
H	5	0.79

5.1.6 ***Employer Resource Imbalance.*** In section 5.1.4 we learned that lower tier IT jobs struggle to compete with top tier ones. Through outlier detection, we discovered start-ups are interviewing students who end up joining large IT companies. We now investigate this in more detail. Table 7 shows the number of interviews per job and number of students hired per job in each community. Top tier jobs (Community E) are able to commit more resources to interviewing and hiring students. Community E had the most hires per job among all communities, while Community G, the lowest tier, had the lowest hiring rate among IT.

Table 8: Student communities in the network

Community	Job Interview Distribution	All Students	Edges	Density (%)	Diameter	Avg Degree	Jobs	Interviews	Hired (%)
Network		4120	80075	0.9	7	39	1970	14275	52
1	Mid/Low Tier IT (55/26%)	1005	11125	2.2	6	22	860	3725	56
2	Finance (83%)	640	7225	3.5	6	23	470	2075	48
3	Research (81%)	625	5475	2.8	8	17	440	1850	47
4	Phys Eng. (93%)	555	8250	5.4	6	30	370	1625	47
5	Civil Eng. (93%)	415	2975	3.5	7	14	260	1200	55
6	Top Tier IT (90%)	410	27025	32.4	4	132	380	2125	60
7	Phys Eng. (65%) & IT (Mixed)	325	2825	5.4	5	17	340	1175	52
8	Top/Mid Tier IT (64/30%)	85	1450	42.3	4	35	145	275	53
9	Product/Research (52/23%)	60	275	14.8	8	9	75	225	53

5.2 Student Communities

There are nine student communities in the network derived from the Louvain Method, which are visualized in 5. We proceed to use the information from the job communities to label the student communities. Table 8 shows the job community makeup of each student community in the "Job Interview Distribution" column, as per step four of our algorithm. Immediately we find that several student communities (2, 3, 4, 5, 6) interviewed mostly with a single job community. This allows us to instantly understand these student communities based on their corresponding job communities. Based on this information, there are four clear IT student communities.

Before we discuss this further, we look at information we would have about student communities if we did not perform community detection on the job graph. The number of jobs students competed for, the number of interviews they had, and the percentage of students hired for any job are tabulated on the far right of Table 8. It is not obvious how the student communities are different if we only considered their success rates. For example, Community 1 has a higher success rate than Community 8, although the tiers of IT interviews are lower in Community 1.

5.2.1 ***Student Network Properties.*** Network properties of each student community are shown in Table 8. We see immediately that Community 6 is extremely dense and has a low diameter. In particular, the average degree between students in the community is 132. Certainly this student community is exceptionally competitive. Community 8 is also dense and has a low diameter which is partially due to its small size.

5.2.2 ***Student Community Labeling.*** It is not obvious from the success rates that student Community 6 had nearly all (90%) students interviewed by top employers and thus is the best student community. Despite not having access to student grades in our dataset, manual extraction of resumes showed these students had exceptional academic performance and extracurricular involvement compared to students in other IT communities.

Community 8 contained several students who interviewed with IT start-ups. This community consisted mostly of second year students, which shows that IT start-ups do not compete for the most experienced students. Larger companies who interviewed students in Community 8 were Yahoo!, NVIDIA, and IBM. We note that these employers, despite appearing in the top IT job community, mostly hired from student Community 8. These companies struggled to acquire the highest tier of IT students.

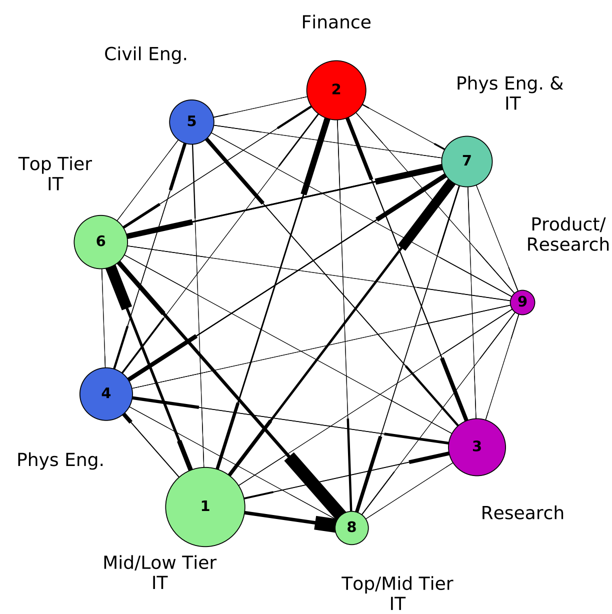

Figure 5: The student community network with communities labeled using the job communities

Community 1 mainly interviewed for mid to low tier jobs at smaller companies, often involving software testing. These students were in their first and second year of study. This demonstrates that smaller IT companies are unable to compete for the same batch of students as the top IT companies.

Notably, Community 7 had engineering students who competed for both physical engineering and IT jobs. These students, as well as students in the finance community, showed a noticeable interest in IT jobs based on their distribution of interviews. This reveals that hardware and finance talent is trending towards IT. Likewise, we have noticed an increase of employers in these industries hiring students for IT jobs in data science and quantitative modeling. On the other hand, certain types of engineering students, especially civil engineering, have remained less interested in IT.

5.2.3 *Student Cross-Community Competition.*

The student community network, where nodes are the communities detected, is shown in Figure 5. The thickness of an edge visualizes the overlap of jobs competed for between communities.[2] We calculated the edge weights and node sides using the same process as in Section 5.1.4. That is, defining J() as the set of jobs that interviewed at least one student in a given student community, the directed edge weight between student communities X and Y is the conditional probability of a job interviewing a student from community X given that it interviewed a student from community Y:

$$Weight(X, Y) = |J(X) \cap J(Y)|/|J(Y)|.$$

The student community network is not a full clique, since the Civil Engineering community did not compete for any jobs that the Top/Mid Tier IT community competed for. This strengthens the notion that Civil Engineering students are not trending into IT as much as other students. On the other hand, the Finance community has a strong connection to Mid/Low Tier IT, which indicates a shift in the industry. The strongest discovery is that many Physical Engineering students are trending heavily towards IT, due to their connections with all tiers of IT. We see that this Physical Engineering Community (7) is very different than the other one (4) due to their low connection.

Community 8, which is the mid tier of IT students, reveals the discovered effect of start-ups and companies like Yahoo! losing hires to top tier jobs. Likewise, we see that students in this community are not as sure of themselves and thus interviewed with more lower tier jobs than students in the top tier IT community. In fact, a job interviewing a student from the lower tier IT community had a 75% conditional probability of interviewing a student from the mid tier IT community.

We also find results that strengthen our findings from cross-community competition in jobs (Section 5.1.4). Low tier IT students are generally unable to compete for the same jobs as top tier students in IT. However, top tier students could interview for lower tier IT jobs as a back-up plan.

5.2.4 *Student-Job Asymmetry.*

We conclude by analyzing what happened to the best students who were not hired. It turns out that there is an asymmetry in the causes of unfilled students compared to unfilled jobs. Manually extracting student records after the fact, we found many top students were hired outside of the institution's internship system. These students were so exceptional that they could find employment on their own. This was not the case for other classes of students, who indeed were unable to find a job. The unmatched top jobs (recall Section 5.1.5), on the other hand, were outliers in the other direction due to having lower quality.

6 CONCLUSIONS

In this paper, we proposed a graph mining methodology for workforce data and demonstrated its effectiveness on a large co-operative internship dataset. We uncovered the nature of competition in the network, including the top employee and employer communities. We successfully found the types of jobs that compete with each other, showing that start-ups, for example, often cannot compete with the largest Information Technology companies in the world. We also showed that natural clusters in the job network can be utilized to rank prospective employees who are otherwise equal. We were able to overcome data limitations to understand each community of employees better and form better-justified tiers of employees and jobs.

Our dataset contained many students who were not hired and jobs that did not hire any student. We discovered several types of jobs that were unaware of the level of competition for students, including small companies and start-ups. Thus we identify several missed opportunities in the hiring process between employees and jobs, which can be naturally uncovered through network analysis. In particular, future work can be done to form recommendations via link prediction.

[2] The edges leaving Community 9 were made thin since many students who interviewed for these jobs also interviewed elsewhere due to its small size.

REFERENCES

[1] Daron Acemoglu, David Dorn, Gordon H Hanson, Brendan Price, and others. 2014. *Import competition and the Great US Employment Sag of the 2000s*. Technical Report. National Bureau of Economic Research.

[2] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (2008), P10008.

[3] Thomas H Davenport, Jeanne Harris, and Jeremy Shapiro. 2010. Competing on talent analytics. *Harvard Business Review* 88, 10 (2010), 52–58.

[4] Santo Fortunato. 2010. Community detection in graphs. *Physics reports* 486, 3 (2010), 75–174.

[5] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. 2008. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*. Pasadena, CA USA, 11–15.

[6] Stephanie Howell. 2015. Global Recruiting Trends 2016. (September 2015). business.linkedin.com [Online; posted 22-September 2015].

[7] Ilkka Kivimäki, Alexander Panchenko, Adrien Dessy, Dries Verdegem, Pascal Francq, Cédrick Fairon, Hugues Bersini, and Marco Saerens. 2013. A graph-based approach to skill extraction from text. *Graph-Based Methods for Natural Language Processing* (2013), 79.

[8] Renaud Lambiotte, J-C Delvenne, and Mauricio Barahona. 2008. Laplacian dynamics and multiscale modular structure in networks. *arXiv preprint arXiv:0812.1770* (2008).

[9] Pasquale Lops, Marco de Gemmis, Giovanni Semeraro, Fedelucio Narducci, and Cataldo Musto. 2011. Leveraging the linkedin social network data for extracting content-based user profiles. In *Proceedings of the fifth ACM conference on Recommender systems*. ACM, 293–296.

[10] Matthew A Russell. 2013. *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More.* " O'Reilly Media, Inc.".

[11] Zhijun Yin, Manish Gupta, Tim Weninger, and Jiawei Han. 2010. LINKREC: a unified framework for link recommendation with user attributes and graph structure. In *Proceedings of the 19th international conference on World wide web*. ACM, 1211–1212.