Revisiting an Educator’s Dilemma: Using Natural Language Processing to Analyze the Needs of Employers and Inform Curriculum Development*

Nathan Green, Xiang Liu, and Diane Murphy
Marymount University
Arlington, VA, 22207
{ngreen,xliu,dmurphy}@marymount.edu

Abstract

Employers in technology are constantly changing their job advertisements and it is a challenge for academic programs to produce graduates who can meet these job requirements and employer expectations. The Information Systems (IS)/Information Technology (IT) educators have been facing the conundrum of whether an emerging technology is a “game-changing” development or something more transient so as to avoid “bloating” the curriculum. This study examines the body of knowledge as represented in our IT/IS program course syllabi and the recent job postings in five of our specialization areas including data science, computer science, healthcare IT, information systems and cybersecurity using natural language processing techniques. One of our goals is to identify the major overlaps and gaps between the two entities systematically by employing quantitative methods. The major contributions of this study lie in that it demonstrates how such data-driven analysis and mining approach informs clarifications to the wording of existing course syllabi, modifications to existing course contents, or the introduction of new courses into the curriculum. Lastly, the future research directions are delineated which this knowledge base can be applied to enhance university graduates’ employability by analyzing students’ resumes and presenting which jobs most closely match their knowledge, skills, and abilities.

*Copyright ©2018 by the Consortium for Computing Sciences in Colleges. Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the CCSC copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Consortium for Computing Sciences in Colleges. To copy otherwise, or to republish, requires a fee and/or specific permission.
1 Introduction

College graduates entering the workplace today are expected to be equipped with a gamut of skills including technical, subject matter-related skills along with practical experience and essential competencies like problem-solving and leadership [13]. However, there have been growing concerns among business leaders across industries, policymakers as well as higher education administrators and educators about the skills gap in the current workforce [11].

Moreover, a multitude of latest surveys and studies on the job vacancies and workforce readiness corroborate such concerns. For example, CareerBuilder’s latest studies on the effects of the skills gap on the U.S. labor market revealed that 68 percent of 2,380 employers surveyed currently have open positions for which they cannot find qualified candidates [2]. What makes the situation more worrisome is that the gap between the number of jobs posted each month and the number of people hired is growing larger as employer’s struggle to find candidates to fill positions at all levels within their organizations. According to the CareerBuilder report, two in three employers (67 percent) are concerned about the growing skills gap and 55 percent of the employers have seen a negative impact on their business due to extended job vacancies.

A research program undertaken by LinkedIn and Capgemini examined the digital talent gap that affects all areas of the business [6]. 50 percent of the organizations participating in this study acknowledged hampering impacts of the widening digital talent gap on their digital transformation programs and agreed that a shortage of digital talent has cost them a competitive advantage. Another example is the latest research conducted by Udemy, which disclosed that the nearly 80 percent of Americans feel the U.S. is facing a skills gap, and 35 percent state that it affects them personally in the face of constant changes [14].

However, no consensus has been reached as for what underlies this mismatch or what causes a widening shortage of skilled workers in the marketplace [4]. Different stakeholders have experimented various approaches to close the skills gap. For instance, some argue that companies should develop their own talent pipeline program as the remedy [12]. The government has recognized the need to connect individuals with careers from an early age and is trying new approaches such as the Department of Labor’s Registered Apprenticeship Program. On the other hand, some propose to develop labor-market intermediaries such as employment agencies or trade associations, employer relationships with technical colleges or other institutions, and employer-provided training to bridge the supply and demand side [15].

How to prepare workers for the future has also largely focused on what educators can and should offer through the academic programs. Colleges and universities have been striving to close the gap using different approaches and
strategies such as the co-op, partnership program run jointly by the employers and university and the connected program or curriculum programs using connections between the various stakeholders throughout a student’s educational journey [5, 17]. However, such efforts mainly address the soft employability skills gap instead of hard employability skills. As university faculty, we are facing challenges to ensure students have both technical skills and soft skills that employers will eventually want and to prepare students to be career-ready and competitive in a global economy. Therefore, this study aims to tackle the challenge from a different angle by examining the body of knowledge as represented in our IT/IS program course syllabi and the recent job postings using natural language processing (NLP) techniques.

The rest of the article is organized as follow. The next section revisits an IT/IS educator’s dilemma in curriculum development. The third section focuses on research methodology and presents the detailed steps involved in data collection, data cleaning, and data analysis using several information retrieval and data mining tools and techniques. The discussion section explores how the result informs clarifications to the wording of existing course syllabi, modifications to existing course contents, or the introduction of new courses into the curriculum. The last section presents our conclusions along with several future research directions.

2 Motivation: Resolving an Educator’s Dilemma

The skills gap is particularly conspicuous in the technology field due to the many fast-paced, disruptive innovations. From voice assistants to self-driving cars to robot caregivers, automation and artificial intelligence (AI) are becoming more and more pervasive. Such phenomenon engenders a stream of controversial debates and conversations regarding the impact of those emerging technologies on the labor market [3, 7]. It is estimated that graduates from the disciplines such as computer science, IT, and IS today will find those skills they have learned from school out of date within six years [8]. There is a pressing need for university educators to examine the current curriculum to identify and adjust those that do not match industry demand to adequately prepare students for the work of the future [1].

However, given the fast rate of change in the employment landscape, the IT/IS curriculum can easily become bloated. An educator’s dilemma is to find a balance between accommodating new material from the discipline, teaching fundamental concepts and maintaining a curriculum with the fixed number of credits in accordance with accreditation requirements. A holistic model was proposed to tackle such dilemma [16]. The authors [16] came up with a conceptual framework consisting of two models: the strategic model for ”when” to
incorporate new technology topics into the curriculum and the tactical model for "how" to insert new courses into the existing curriculum. The major limitation of this framework is that "when" and "how" decisions are subjective and to some extent "ad hoc" instead of "data-driven". Further, no formal assessment of the effectiveness of this framework has been conducted. Thus, this study attempts to tackle the dilemma using a more flexible and agile approach. One of our goals is to identify the major overlaps and gaps between the two entities (i.e., course syllabi and job advertisement) systematically by employing quantitative methods.

3 Methodology

This section covers details of data curation, data cleaning and analysis techniques that were used to conduct an initial explorative study.

3.1 Data Curation

The process for conducting our study requires two data sets: a set of job postings for an industry and the current syllabi used by the program or department for the relevant majors. To get the required job skills for an industry, we created a web scraper for one of the leading online job seeking and recruiting websites, indeed.com. This scraper retrieved all textual information on a job posting while removing all html and JavaScript code. Doing this we lost the structure of the post but retain all text in the posting. To scrape selected jobs, we restricted the search to a location, entry level jobs only, and a search term which is required to be in the post. For instance, our scraper can pull the first 100 pages of a search with 15 jobs per page (1500 jobs) for selected parameters such as Entry Level Jobs in Washington DC for "Information Technology". The results for each one of these data scrapes was then saved for further analysis.

To collect what skills the university currently covers in its curricula, we collected the last syllabus used for each course, 99 in total. Using data extraction techniques with Python scripting, we extracted all the textual information for the syllabi. The extracted text from each syllabus was saved in its own file so each individual class could be compared to industry needs.

3.2 Data Cleaning

An average job posting from indeed.com contains many boilerplate sentences, which is irrelevant for the goal of our analysis to see if the university’s courses contain all words that the job posting contains. For instance, basic language and frequent words (the, is to, a, an) are not particularly useful for our analysis.
These words are often called “stop words” and these stop lists come with NLP kits. We remove all stop words from the job postings using the Natural Language Toolkit’s (NLTK) stop word list for English [9]. With the basic stop words removed, most job postings still contain some common job-related terms. Most of these deal with employment and Human Resources such as Equal opportunity, LLC, headquartered, experienced, etc. To remove these terms from our evaluation, we hand curated a list of more than two hundred common employment terms. After removing both the stop words and the common employment terms, we obtain a list of terms that mostly relate to job skills needed to fill the position.

Many of the data issues associated with the indeed.com data are present in our syllabi data as well. For example, syllabi also come with boilerplate information as the template is given by the university. We conducted the same data cleaning procedure to remove all stop words from the syllabi. The ultimate goal for analysis is to get rid of noise information about school policies (late assignments, quizzes, test, drops, and withdraws) and extract the skills taught by each course. We crafted a list of 200+ academic terms as stop words and removed those from each syllabus as well. In the end we were left with only the key concepts in each syllabus.

3.3 Analysis Techniques

To compare the skills taught in a class with industry needs, we created a process that would rank the relevancy of each course to a specific industry. First, we extracted the key terms that represent an industry and represent each syllabus respectively. We conducted this extraction automatically using the common information retrieval technique of term frequency–inverse document frequency (tf-idf). Tf-idf ranks the importance of a term to the target.

![Figure 1: Flowchart of the Research Methodology](image_url)
Having a value for each term, we turned each document into a vector of values representing the terms in the document. To find out if two documents were similar we took the cosine similarity measure between the two vectors. If the cosine value is equal to 1, they are the same document, if the cosine value is 0, the documents share no terms in common [10]. We conducted cosine similarity analysis on both vectors and generated one file for each industry. In each file the courses are ranked by how closely they match the industry job postings based on their cosine similarity value. For instance, if we want to inquire about the top five courses offered in our program that cover the skills matching the Data Science industry job needs most closely, we would see: IT390, IT385, MSC385, MSC325, IT820. These courses accurately reflect the data science components of our undergraduate data science specialty in the IT major as well as a course in the doctorate in cybersecurity program. Figure 1 illustrates the steps involved in our research methodology.

In addition, we had the goal of seeing what we should be teaching that industry needs and what we are currently teaching that is not relevant to industry. These features are created as a byproduct of the above tf-idf analysis. Industry needs not currently covered in a class can be discovered by taking the set difference between terms in the keywords in industry and the keywords in the syllabi, represented by A in Figure 2. Topics covered in courses but not mentioned in job postings are covered by the reverse, the set difference between terms in the syllabi and terms in industry, represented by B in Figure 2. The code and analysis for this research has been scripted using Python. To give a high-level view of how we are doing as a school each semester, we calculate a coverage ratio. This is defined as the percentage of terms in industry that we cover in the syllabi. We calculate this in order of rank, so we can see how we do on the top 10 industry terms, as well as top 25, 50, 100, and 1,000. If all terms are examined, regardless of rank, this would be the ratio of C/A in Figure 2.

Figure 2: Venn Diagram showing the relationships between industry needs and course topics
4 Result Analysis and Discussion

In this section several different analyses on the preliminary results are presented. The purpose is to examine gaps and overlaps between our curriculum and industry job needs from various angles in order to achieve a holistic picture.

4.1 Analysis 1: Concepts in Syllabi Not in Job Advertisements

Figure 3 is a list of the top 50 terms that were identified as frequently occurring in our syllabi but not appearing in the job description.

| programmed | portfolios | rfid | globalization, globalized |
| literacy | recursion, recursive | tracer, ipconfig | outsourcing |
| webpage | traversing | deterministic | polynomial |
| macintosh | watermarking | cybercriminals, cyberterrorism, cybercrime cyberstalking | backdoor |
| ciphers | authenticated | elliptic permutation | asymmetric |
| randomness, pseudorandom | cryptanalysis, cryptosystem | trapdoor | decryption |
| sha md5 | symmetric | higherperformance | traversed |
| asynchronously, synchronizing | hotspot, threads | cyberspace, physical, usenix | |
| enigma | multithread, threads | deadlock | pentester |
| cyberinsurance | motherboards motherboard microcomputer | webapps storyboard | tuples |
| conditionals | concurrency | cyberattacks | honeynets honey pots |
| ransomware | unmasking | infections | backdoors |
| singleton | antispypware | corpora cordova | tokenization |

Figure 3: Top 50 Terms in Syllabi Not in Job Advertisements

Most of these terms cover fundamental concepts in the core areas of computer technology (e.g., motherboards), software development (e.g., threads), networking (e.g., synchronizing), databases (e.g., tuples), and cybersecurity (e.g., ciphers). These reflect the depth of coverage in our syllabi. RFID is the only technology that is mentioned in the syllabi and not in the job advertisements selected. Some terms need to be edited in the syllabi, such as “pentester” to “penetration tester”, webapps to “web application” and “honey pots” to “honeypots”.

4.2 Analysis 2: Skills Required by the Industry, but Not Covered by Each Concentration

While tf-idf and cosine similarity gives us a ranking of our courses per industry, it still does not let us know if our curriculum is missing any key concepts. To analyze this, we also generate a list of key terms that occur in the indeed.com postings that do not occur anywhere in our syllabi. The top 5 terms in each specialization are shown in Figure 4.
These are the unfiltered top 5 terms. Geospatial, GIS and geographic terms occur multiple times in three specialties and would appear to be a possible addition to our program. Our coverage in cybersecurity appears more than adequate and the highly occurring terms are more descriptors than skills. However, “genetics” as a term in cybersecurity will be researched further as this might be an emerging area. The use of “developer” rather than “programmer” will result in a review of our software development syllabi. Similarly, the use of “analysts” will be reviewed. Open source intelligence (“osint”) is covered in the cybersecurity syllabi but we will further investigate its use in data science, outside of cybersecurity.

The Healthcare IT words are not technology related but show how Healthcare IT has migrated outside the medical offices and hospitals. It is now being implemented in chiropractors, rehab clinics, spine clinics and sports facilities.

### 4.3 Analysis 3: Coverage Ratios

The coverage ratio represents the percentage of top keywords that exist in a course. Figure 5 shows ratios for the top 10 to 1000 keywords in each of the five specialties.

<table>
<thead>
<tr>
<th>Concentration</th>
<th>10</th>
<th>25</th>
<th>50</th>
<th>100</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEALTHCARE IT</td>
<td>0.6</td>
<td>0.8</td>
<td>0.86</td>
<td>0.88</td>
<td>0.726</td>
</tr>
<tr>
<td>INFORMATION SYSTEMS</td>
<td>1</td>
<td>0.92</td>
<td>0.88</td>
<td>0.8</td>
<td>0.711</td>
</tr>
<tr>
<td>DATA SCIENCE</td>
<td>0.9</td>
<td>0.92</td>
<td>0.92</td>
<td>0.94</td>
<td>0.726</td>
</tr>
<tr>
<td>CYBERSECURITY</td>
<td>1</td>
<td>0.96</td>
<td>0.92</td>
<td>0.94</td>
<td>0.811</td>
</tr>
<tr>
<td>COMPUTER SCIENCE</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.99</td>
<td>0.865</td>
</tr>
</tbody>
</table>

Figure 5: Coverage Ratios by Specialty

For the top 10 words in Information Systems, Data Science, and Cybersecurity specialty there are at least one course that mentions each of them (100%). For Data Science we have 90% (9 of the 10 keywords) and for healthcare 60%. Further analyses show that this number changes as the number of keywords
increases. Computer Science is fully covered for the first 100 keywords. Cybersecurity coverage is also good through 100 keywords. For Healthcare the value goes up as we go down in keywords, so the syllabi are covering many of the mid-term words. Further analyses will include coverage for when a keyword happens in at least 2 courses, 3 courses, and so on.

4.4 Analysis 4: Programming Languages Requested in Job Advertisements

Learning to program is part of the IS/IT curriculum, and academics must select one or more languages to teach basic and more advanced concepts of software development. Figure 6 shows the five most common programming languages for each specialty.

<table>
<thead>
<tr>
<th>Healthcare IT</th>
<th>Information Systems</th>
<th>Data Science</th>
<th>Cybersecurity</th>
<th>Computer Science</th>
</tr>
</thead>
<tbody>
<tr>
<td>java</td>
<td>java</td>
<td>python</td>
<td>python</td>
<td>java</td>
</tr>
<tr>
<td>cobol</td>
<td>javascript</td>
<td>java</td>
<td>java</td>
<td>python</td>
</tr>
<tr>
<td>javascript</td>
<td>spark</td>
<td>spark</td>
<td>shell</td>
<td>javascript</td>
</tr>
<tr>
<td>puppet</td>
<td>bash</td>
<td>ruby</td>
<td>perl</td>
<td></td>
</tr>
<tr>
<td>python</td>
<td>javascript</td>
<td>perl</td>
<td>ruby</td>
<td></td>
</tr>
</tbody>
</table>

Figure 6: Most Common Programming Languages in Job Advertisements

Java and Python are the primary languages requested across all specialties. This reinforces our curriculum decisions where Java I and II are recommended for the Computer Science specialty and Python (introduced in 2016) for the Data Science and Cybersecurity specialties. Web development, including the JavaScript and Perl languages, is also offered and recommended for the Information Systems specialty. Apache Spark is an important addition for the Data Science and Information Systems specialties and should be added to the programming environment for these options. Cobol was unexpected and indicates the reliance on older software in the healthcare field.

5 Conclusions

This was an initial analysis of the two datasets: curriculum scope and job advertisements in the IT/IS field. While there was substantial overlap, the initial analysis showed that there were some gaps. In several cases this can be overcome by a careful review of the syllabi to ensure that they reflect the current terminology in the field and that variations on terms are used (e.g., analysts not just analysis). In other cases, a new course might be needed. In this initial review, geospatial/GIS seems a good addition to the curriculum and will be researched further.
There are many additional analyses possible now that we have the datasets in place. We can run these analyses every semester to identify changes in the job advertisements as well as any curriculum changes that we have made. We will also do additional analyses looking for other gaps over time. For example, break down the syllabi by level (undergraduate, masters, and doctorate) and tie this into job advertisements and their education requirements. Further research will also be performed with a third data set, applicant resumes, to determine whether it is possible to use these data to promote certain job positions for individuals in the program as part of our job readiness activities.

References


[16] X., L., and D., M. Tackling an is educator’s dilemma: A holistic model for “when” and “how” to incorporate new technology courses into the is/it curriculum. SAIS.

[17] X., L., and D., M. Constructing a connected program in it/is: A non-r1 university case. SAIS.