INTRODUCTION
For any individual interested in finding a professional career to pursue, one of the most compelling problems they face is considering which potential job best aligns with their skillset. Without a doubt, this knowledge will give an individual a better, more realistic sense of the potential opportunities that they will be well-equipped to pursue. But even beyond gaining a better idea for a set of potential career opportunities, having a clear idea of which skills are desirable for which fields is extremely valuable, especially in a highly competitive job market. That way, individuals can also think about which skills they might like to acquire or better hone in order to make them better suited for certain careers.

There were three key aspects of the job and skillset matching process that were particularly compelling to us. These included:

1) **Skill Matching.** We wanted individuals to acquire a good idea of the jobs that they could have based on the skillsets that they possessed or hoped to acquire.

2) **Aptitude.** We wanted our users to understand how their aptitude levels at various skills affected their career possibilities. This knowledge would indicate how certain skills are valued for certain careers, which would then allow the user to determine what new skills they might like to acquire or become better at.

3) **Salary and Job Demand.** One of the most important factors an individual considers when deciding which career to pursue is the salary of that career [5]. Thus, we also wanted users to be able to compare desirability of different jobs based on salary level, a key motivator for choosing a career. Furthermore, we wanted users to be able to easily see which skills were in the highest demand based on the number of jobs that those skills were connected to.

Thus, we wanted to create an effective, compelling visualization that matched users to a desirable set careers based on an input set of skills and aptitudes. Our visualization, Dexio, hopes to answer this problem by leveraging dynamic graph filtering techniques that allow users to instantly compare the marketability of individual skills, the strength of the match between different skillsets and jobs, and the desirability of a job based on expected income.

RELATED WORK
An extensive exploration of existing job-skill matching visualizations revealed that the while the visualizations had extensive databases containing very useful information on careers and skills, all the information was statically displayed, requiring the user to progress through several screens to finish a lengthy, unexciting quiz (Figure 1). Furthermore, when viewing the results of the quiz, the user is again presented with a static, tabular list of potential careers, but the results require extensive parsing and careful reading (Figure 2).
Figure 2: Results of the skill quiz are statically displayed in a tabular format.

Research performed by Robert L. Morgan of Utah State University indicated the usefulness of Dexio’s approach to job and skill visualization [3]. In his study, Morgan used a career-skill assessment that matched individuals to careers based on how proficient the users were determined to be at each of the list pre-determined skills. His results showed the potentials of an assessment that takes into consideration the level of aptitude at particular skills that the individual has. However, Morgan’s tool had the same shortcomings as the aforementioned career surveys: his web-based assessment still required to user to parse through several static pages of listed skills and aptitudes, generating a multi-page report of the results from the test.

Finally, research performed by Christopher Ahlberg & Staffan Truvé of the Chalmers University of Technology also explored the career matching domain as well [4]. Their research explored the efficacy of dynamic querying and tight-coupling of the information to the visualizations on the screen and confirmed the effectiveness of this sort of interface. While Ahlberg and Truvé’s interface misses many of the relationships that we hoped to visualize with Dexio, such as relevant skills being linked to multiple jobs, the guiding principles we learned from their research informed our decision to incorporate dynamic querying and filtering of graph nodes in our visualization.

METHODS

Data Set:

Both datasets came from government entities and provided broad coverage of the national labor market.

Our data set comes from multiple sources including O*NET, or The Occupational Information Network, which is a government sponsored initiative to collect job data based on surveys of workers from each occupation. The data set from O*NET includes 974 occupations and 33 general skills as well as the importance level and required aptitude level for each relevant skill associated with a job. Our second dataset comes from the Bureau of Labor Statistics and contains the median salaries of some of the occupations listed in the O*NET dataset from a May 2015 survey.

Visualization Design Decisions:
We chose to build our visualization as a force directed graph because the graph links provide an clear way for users to associate skills to different jobs. Force directed graphs also allow users to see how closely related different skills are based on the size of their intersecting sets of job nodes. While force directed graphs ensured that nodes would be laid out in such a manner that all nodes are easily visible, there were a number of tradeoffs we considered when choosing this format. One tradeoff was the loss of position as an encoding method for data because force directed graph node positions are fluid and calculated based on the position of all the other graph nodes. While position is ranked highest on Mackinlay’s chart of effectiveness, we decided that giving up control of position was a worthwhile trade because it did not hinder users’ abilities to understand node relationships and also laid out the nodes in a way that was easy for users to explore the nominal data that included salaries and required additional skills. Lastly, force directed graphs are designed to position nodes with more connections in the center area of the graph, which generates a natural grouping for nodes of best fit.

We used d3.js, an extensive JavaScript library for data
visualization to create our visualization. We began with a force-directed graph to encode the relationships between jobs and related skills, and extended the functionality so that the visualization adequately addressed our use cases. These extensions included:

1) Dynamic filtering of the jobs and skills in real-time
2) Increasing the opacity and visibility of the skill nodes based on the number of out-links that it has so users can quickly identify jobs they are best suited for
3) Changing the size of the nodes based on salary ranges of the job so users can have a sense of what skills lead to the most lucrative and in-demand careers
4) Toggling the skill level a user has for each particular skill and then matching the user to skills based on those skill levels
5) Providing information in popup boxes upon hovering over job nodes that describes the median salary and additional required skills and skill levels to be qualified for a particular job.

Dynamic filtering: We want users to see only relevant subsets of the overall job and skills graph so our application filters out irrelevant job nodes and links based on the user inputted skill. We filter jobs by two criteria. The first criteria was deciding whether or not the entered skill was in the top 5 most important skills associated with the job. The second criteria is whether or not the user inputted aptitude level is above the level required for the job. Once the data is filtered, the resulting nodes and links are appended to the existing graph.

Increasing opacity based on outlinks (degree): We thought a lot about the topics covered in the beginning half of the class such as the different principles of presenting ordinal and quantitative data. Based on Mackinlay’s ranking of the effectiveness of encoding quantitative data and the constraints of our force directed layout, we first chose to test area as a way to encode the degree of a node. In one of our early iterations, we scaled a job node’s radius according to its degree so that a job node with two outlinks would have twice the radius of a job node with one outlink. While this area encoding was effective in highlighting strength of match, we found that area would be better suited to encode the ratio of salaries associated with jobs. Therefore, we went further down Mackinlay’s ranking and experimented with saturation, or opacity in CSS. We linearly scaled the opacity of job nodes based on the number of connections multiplied by 0.2. Because each job node is associated only with the top 5 required skills, a job node with 5 connections is a 100% match and therefore has full opacity. Our user testing showed that people could accurately compare different opacities between job nodes and that users immediately focused their attention on nodes with higher opacity.

Scaling job node size based on median salary: Our research on the motivations of people searching for jobs indicated that salary was the most important factor when considering the desirability of a job. We chose to use node area to encode salary because it is easy to compare salary through areas. Because cognitive research has shown that people are terrible at estimating ratios of areas, especially of circles, we scaled the node radii rather than the area so that a job with a $100,000 salary has twice the radius of a job with a $50,000 salary. Our user testing showed that people naturally hovered their mouses over nodes with larger areas, and that scaling areas by made higher paying jobs stand out significantly more, which aligned with our user motivations to seek out the highest paying jobs.

Toggling skill levels: Dexios allows users to gradually increase individual skills and to see dynamically how higher skill level increases the size of their career options graph and how many high paying jobs enter the graph. The goal of this functionality is to help users prioritize which skills to acquire or improve on and to what degree they should try to increase their aptitude levels for those particular skills.

Popup Boxes: To aid job seekers in considering different jobs, we provide job metadata including the job title, median salary, and list of additional required skills and skill levels to be fully qualified for the job. Because this nominal data is considerable in length, we chose to display only the label for one job at a time based on the node the user hovers the mouse over. When the mouse enters a node, the popup box appears with this information and then disappears as soon as the mouse exits the node so that users can quickly browse an entire cluster of jobs.

RESULTS
Our development process required multiple iterations and tests on sample users. Below we discuss the result of our
technical implementation and the feedback we received from user studies with ten different subjects.

**Legend**

- = Skill
- = Job

**Strength of Match:**
- (lowest to highest)

**Median Salary:**
- (lowest to highest)

**Skill Level Scale:**

- Novice
- Moderate
- Expert

Figure 3: The legend summarizes the visual design decisions made to encode the job metadata.

![Image of legend]

Figure 4: An example of the job graph of a user with level 5 aptitude in 4 different skills.

![Image of job graph example]

Figure 5: An annotated version of a graph that we used to test on sample users.

During our user tests regarding Figure 5, the majority of users—seven out of ten—immediately moved their mouse to hover over Node A, which had the largest area and highest opacity. Afterwards, again seven out of ten users next examined the nodes in Cluster B, which suggested that our encodings of size and opacity effectively highlighted jobs of best fit and greatest desirability to users. Our testers rarely explored nodes in the periphery, such as those in Cluster C at the bottom left hand corner of the graph.

Our users also expressed a general curiosity with the dynamic properties of the graph and reported that they were extremely pleased with the ability to learn about new, high paying jobs that they did not know about before, which were displayed to them based on their skill sets. Users also said that they enjoyed seeing the marketability of different skills based on the graph sizes generated from different skill nodes. For an example of a highly marketable skill, see Figure 6 below, which show the two skills—‘Writing’ and ‘Reading Comprehension’—with the largest job graphs (Figure 6).

Figure 6: A graph generated with Level 5 ‘Writing’ and ‘Reading Comprehension’.

Users also reported that the clustering of the nodes in the graph made it easier for them to choose different sections of the graph to explore. For example, in Figure 7, the graph breaks itself up into seven clusters despite containing hundreds of nodes.

Figure 6: A graph generated with Level 5 ‘Writing’ and ‘Reading Comprehension’.
However, users complained that the variety of skills were not specific enough, which was reflective of the general nature of the data set.

A final aspect of our user studies concerned the general efficacy of Dexio in comparison to existing job-skill visualizations. Half of the users were randomly selected to use Dexio first, and then the Education Planner skill matcher second, and the remaining half of the users were selected to use the Education Planner skill matcher first and Dexio second. They were then asked to rate their satisfaction, on a scale of 1 to 5, with 5 being “Extremely Satisfied”, with their experiences with both tools. Dexio had an average user satisfaction rating of 4.7, while the Education Planner skill matcher had an average user satisfaction rating of 2.4. Every one of our subjects rated their overall satisfaction with Dexio higher than their overall satisfaction with the Education Planner. We also measured the time it took users to understand the results presented by both of the visualizations. On average, it took users around five minutes to read through and understand the results presented by the Education Planner tool after inputting three skills, compared to Dexio, where on average it took users around one minute to read through and understand the results. Finally, we asked users whether they preferred a tool like Dexio or the Education Planner, and again found that users unanimously preferred Dexio over the Education Planner job-skill matcher.

DISCUSSION
The results from our user studies showed that the user satisfaction ratings for Dexio were unanimously higher than those for the traditional job-skill matcher and that Dexio was unanimously preferred over the existing job-skill matchers. As reported by our users, a significant reason for this result is the fact that Dexio supported dynamic querying of skills. This way users were able to update in real-time the skills that they possessed and immediately see the results of their inputs. This result speaks to the importance of dynamically updating the visualization based on new information added in the same way Dexio immediately adds new nodes to the graph based on new skills that the user adds, and how instantaneous feedback increases both the usability and efficacy of a particular interface.

In addition, Dexio provided an immediately compelling, intuitive visualization. On average, the users in our study were able to parse through and understand the results presented by Dexio five times faster than the tabular lists of results presented by the existing job-skill visualizations. Users, upon first glance at the resulting graph of jobs and related skills, were able to more quickly recognize how popular--and by extension, how important--certain skills were based on the number of children that each skill node had. In addition, as the user added more and more skills, having the ability to view the intersection of the various jobs relating to each skill node was enlightening as well. These two crucial features were highlighted by a majority of our users as the reasons why Dexio was much more intuitive than traditional job-skill visualizations. This result points towards the promising use of force-directed graphs in contexts such as job-skill matching where understanding the different relationships between the data represented on the screen is a crucial component to overall understanding and parsing through the results.

Throughout our user studies, we also uncovered that users were drawn to hover over the larger and more opaque nodes. This implies the success of using size and opacity to effectively encode the best fit and most desirable jobs for our users, which is especially relevant for future visualizations which employ the usage of force-directed graphs.
Finally, another important insight Dexio provides with regards to visualization is the importance of node clustering. As mentioned in our results section, users were very pleased by the fact that related nodes were clustered together, and one of our biggest pieces of feedback was to further group related nodes by industry, which we will explore in the following section.

NEXT STEPS

We believe that dynamic filtering and clustering of the nodes based on skill aptitude and job salary are incredibly important in helping users quickly understand their job prospects. Our presented visualization verifies the effectiveness of such a system over methods that currently exist. However, there are still interesting areas where future work can be done:

1. Extended user studies to quantify the efficacy of our visualization for providing insights into how good of a fit the jobs suggested to the user by Dexio were. This would potentially include a longitudinal study of how satisfied the users were in the jobs that they ended up choosing based on Dexio, in addition to an algorithm that ranked user fit for the various jobs suggested to them.

2. Node clustering based on job similarity. While our current visualization displays careers based on skill relatedness and salary, it would certainly be more compelling if the job nodes were clustered based on how similar they were (i.e., jobs from a similar industries, etc).

3. Link to live job listing databases: While our visualization shows the median salary of different careers, it does not show the demand of the market for those careers. Users would benefit from being able to tell how many job openings are in each career area to better guide them of which careers to pursue.

4. Improved control over increasing and decreasing aptitude levels for individual skills: Users would benefit from being able to adjust the aptitude levels of their entered skills to be able to see how improving certain skills will affect their job opportunities. This added functionality would help users understand which skills to focus on improving to best increase their desirable job opportunities.

5. Modifying link-length as a way to encode skill aptitude level: In our current visualization, link length does not encode any data. There is potential to use link length to show the aptitude level a job demands. For example, a job that requires higher levels of ‘Writing’ could have a shorter link and be positioned closer to the skill node than a job that requires lower levels of ‘Writing’.

REFERENCES


