University of Glasgow | School of Computing Science

Are matrix-based or node-link graphs more readable when representing causal relationships for social and health data?

Kristina Lazarova

School of Computing Science
Sir Alwyn Williams Building
University of Glasgow
G12 8QQ

Level 4 Project — March, 2017

Abstract

We show how to produce a level 4 project report using latex and pdflatex using the style file l4proj.cls

Education Use Consent

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic format. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Name: ____________________ Signature: ______________________

Contents

Chapter 1

Literature Review

1.1 Data Visualisation

Data visualisation has been described as a technique that makes use of computer-supported, interactive illustrations to deepen human's understanding of a dataset [4]. Information visualisation is necessary when dealing with increasingly large and complex data.

Information visualisation systems are most helpful when a set of data is being explored [10]. Usually this occurs not when someone is looking for a specific answer to a question, but when a deeper understanding of that data set is sought. It was suggested that the value of information visualisation is not in understanding a specific question, it is about developing and deepening one's insights of a set of data [10]. Data visualisation is able to facilitate this process because there are a number of cognitive benefits associated with it. A large visual cue that illustrates data becomes a single point of reference for human cognitive processes. Visuals become external cognition helpers in facilitating human memory by providing a bigger working set for analysing data. Furthermore, visual architecture and design applied at company and department levels have been reported to be successful due to the low cognitive burden for visualization reading [22].

Data visualisation has been been used in a number of fields. For example, it is suggested to be successfully used in Bank of America Chicago Marathon, which is one of the biggest marathons in the world [18]. Large amounts of live data are gathered to establish some of the main principles for event management information visualization. Some of the main benefits from these practices are keeping track of the progress of the participants, communicating information between the agencies organizing the marathon efficiently, and improving medical preparedness. Moreover, astronomical researchers have been suggested to benefit from data visualization techniques [13]. One such technique is linked views in which a user is able to select, highlight and include or exclude points from display and analysis from various data visualisation dimensions. For example, an important visualisation for astrologists can be interactive exploration of relationships between data points in statistical graphs and locations in live 3D space. Another instance of beneficial data visualisation is ”Geographic Information System” tools which are used in demographics and geography. For example, Engage3D provides the functionality of exploring layers of maps in a linked-view systems.

Inventions in this field have been continuously made in order improve the visualisations and benefit the target public. For example, a heatmap represents a graphical illustration that evaluates conical distribution values around data points based on a respective data value that has associated respective data point [5]. Various implementations of a heatmap were developed and discussed since the original heatmap was created, in order to enable easy future work with heatmaps suited for a particularly needed area.

There have been data visualisation issue related to Big Data [14]. First, visual noise is the phenomenon of having so many data points on the screen that the user cannot see them as separate points which leads to visibility loss. Second, large images are dependent on the device resolution abilities. However, even if add together many devices for partial data visualisation to represent a more detailed illustration or a larger amount of data, human perception will eventually reach its limit. Therefore, after a certain point people will not be able understand the representations or their analysis. A third problem is Information Loss which is caused by the previous issues because of data filtration and aggregation. These techniques are likely to hide information and mislead people analysing the data. Another problem is High Performance Requirements especially in dynamic visualization. In addition, the visualisation might be accompanied by high rate of image change and lead to people being unable to react to the number of data changes.

A Big Data visualization method is suggested to be TreeMap [14] which is a hierarchical way of representing data by rectangles. Some advantages of this method are that the hierarchical matching clearly shows data relations and extreme outliers are visible. The disadvantages related to this method are that the data must be hierarchical which makes it unsuitable for examining time patterns, for example.

A well known issue with data visualisation is that sometimes it is challenging for people to understand it. A study aimed to examine familiarity of museum visitors with different visualisation techniques [3]. They included charts, maps, graphs, and networks to reveal how familiar people are with them. It was found that even though most participants were interested in science and art they have difficulties naming and reading the visualizations. It was concluded that people are interested in visualisation techniques, but have significant difficulties in naming and understanding them.

In order to solve this issue, another research area has focused on understanding how information visualization novices think and the approaches that can facilitate their learning. A study used sales data to find the barriers for novices when reading iterative visualisation construction and the way they think about visualisation specifications [15]. They found that the biggest barriers were interpreting questions into data factors, visual mappings, and understanding the visual representations. It was found that there is a need for instruments that suggest possible visualizations, facilitate help with learning, and are integrated with tool support for the whole analytic process. Furthermore, recent research acknowledged that individual differences between people will have influence on the way they interpret graphical representations [28]. They suggested that visualization performance can be improved by personalising visuals according to one's needs, abilities and preferences.

Very little is known about the way users understand and read data visualisations and how they interact with different layouts [9]. It is interesting to explore if there are layouts which provoke better task performance and better response time. Etemadpour and colleagues [9] conducted a research to find out more about performance in similarity based layouts that are generated by multidimensional projections. The results suggested that projection performance is task-dependent and depends on the nature of the data. Therefore, they concluded that the same data layout can have different performance on different tasks and that performance will also be influenced by the characteristics of the data.

1.2 Causal Relationships

Causal relationships are of great interest for scientist who examine influence of different factors on each other. Particularly, research regarding health, social and behavioural sciences aim to investigate questions about causal rather than associative relationships. With the help of statistical analysis associations among different factors can be inferred. Associations are relationships that can be observed in joint distributions of factors such as regressions and correlations. Causal analysis, in addition, is the practice that aims to infer probabilities under factors that are changing [26]. These could be, for example, changes influenced by factors such as drinking, childhood issues or applied treatments. Causal relationships cannot be defined from the distribution alone - for example, effect, confounding, disturbance.

A study looking to identify factors influencing blog design used the Decision Making Trial and Evaluation Laboratory method (DALMATEL) which is used to illustrate the relationships between factors and allows causal relationships to be shown [19]. Some of the causal relationships they found were that color arrangement directly impacts simplicity of layout, colour arrangement directly impacts font arrangement, and color arrangement impacts itself.

ReView is a tool for finding causal relationships in anomalies in network traffic. It has been suggested to facilitate better understanding of network representations [33]. One of its features is minimizing the detailed information while showing the causal relationship. ReView can also quickly navigate the user through networks with a large number of requests and levels of abstractions.

Causal relationships are thought to be perceived directly if they are accompanied by animation [31]. The researchers followed Michotte's perception of causality principle which illustrates a causal event with a billiard ball hitting another ball and causing its motion. Ware and colleagues [31] introduced a visual causal vector method that shows a causal relationship between representations using animation. They found that the perception of causality depends on the simultaneous occurrence the visual causal vector and the change in the graphical representations.

1.3 Node-Link Graphs

Node-link graphs consist of nodes which are connected by edges. Large amount of work has been dedicated to visualizing those structures in the most readable way. However, the larger the network that is being illustrated, the higher the possibility that the graph becomes dense and unreadable. This is especially true when the direction of the edges is important [8]. When representing causal relationships with node-link graphs if node n is causing node m, then the edge between those two nodes will point towards node m. In order to avoid the complexity of the large network some researchers have tried to add interaction with the graph [11], while others' intention was to create visuals that do not change the structure of the network and does not require interaction [8]. An innovative interaction technique was designed by Abello et al [1] where a user is able to navigate through a hierarchically clustered base of the graph. In the non interactive technique researchers decreased the complexity of large directed graphs by replacing single edges with edges which connect to groups of nodes [8].

Data driven journalism is also concerned with difficulties in showing directed relationships in large datasets [25].

Nodes have also been used to represent "visual programming language" or workflow and processing functions [30]. The system translates the user input into "a sequence of data language", which will then be transformed into "service commands" that will be executed. The dependencies between the nodes and data flows are illustrated as interconnections between the vertices on the graphical user interface.

A collider "C" is well represented by an "inverted fork", $A- > C < -B$, in which the arrow shows a direct link from the tail factors to the head factor [16]. Stratifying on the collider is likely to induce bias and influence the the association between A and B. In addition, a similar pattern found in directed acyclic graphs, called a confounder, can be illustrated by a "causal fork" - $A < -C- > B$. Greenland [16] suggested that any change in the A-B link upon C-stratification will induce bias. Statistical adjustment for collider bias is suggested to lead to as much bias as not making adjustments for a confounder [20].

Directed acyclic graphs (DAGs) are increasingly used in representing causal relationships in modern epidemiology [29]. The factors in DAGs are connected by arrows which can never create a closed loop [17]. A path is defined by a sequence of arrows between the factors of interest. The DAG method is valuable as it is found to show underlying relations explicitly and easily identifies sources of confounding. Confounder is known to be a variable that is related with the exposure and the outcome but does not exists in the causal path between them.

For example, chronic kidney disease and mortality are often caused by age. Confounding occurs when reading the causal relationships between chronic kidney disease and the outcome mortality.

1.4 Matrix-based Graphs

An Adjacency matrix is frequently used to represent a network [24]. If a network consists of n nodes, the matrix will consists n x n grid of cells. This is considered to be an unambiguous way of representing data. However, some of its disadvantages are that the area increases quadratically and as large networks are sparse there will mainly empty space on the matrix. Matrices can be used to represent both directed and undirected graphs.

A new technique called Compressed Adjacency Matrices was introduced in 2012 for visualising gene regulatory networks [7]. As those directed networks have specific structural traits, standard representations such as adjacency matrices, and node-link diagrams are unable to depict all traits. Compressed Adjacency Matrices cut and rearrange adjacency matrix so that no space is wasted in case of sparse network. There are specific structures which represent sub networks. This is how scientists came up with a new data structure in order to fit the characteristics of the data they analyse.

Furthermore, PathwayMatrix is another visualisation tool that represents specific relations between proteins in a pathway [6]. The implementation of the tool consists of adjacent matrices that interact with each other. Additional features were added to facilitate the data analysis. This visualisation software received positive feedback in the specific area of representing relations in proteins pathways. Consequently, there is no one best representations technique. Depending on the dataset specifications, the complexity and size of the data there might be many or only few sufficient ways to visualise it.

A performance comparison between square and triangular matrices has been conducted to measure performance speed and accuracy [23]. It was found that performance is influenced by the matrix juxtaposition type which lead to the creation of a new matrix visualisation called TileMatrix. It represents a large amount of matrices and is effective at analysing networks that are multi-faceted and time-varying. With TileMatrix it is easy to see differences in matrices across time and facets. However, triangular matrices can work only in non-directed networks.

Chapter 2

Introduction

Brain connectivity visualisations are in the form of weighted graphs which are node-link graphs in which each edge is given a numerical weight. Alper and colleagues [2] compared augmented adjacency matrix with node-link visualization by conducting a controlled experiment. They found that matrix-based graphs outperform node-linked graphs. It was concluded that for weight graphs, node-link representations are less readable and more error-prone when compared to matrices. On the other side, node-link graphs are adjustable to a specific spacial representation which might also be insightful when reading the graph. Overall, matrix-based graphs had higher accuracy.

Moreover, another paper suggested that adjacency matrices are superior to node-link graphs in representing dense graphs because they are more compact and easier to look at [27]. However, they also stated that node-link graphs are better for path finding as a path can easily be followed if the arrows are not too tangled.

Adjacency matrices and node-link diagrams have been compared in another study [21] to examine which is more suitable graphical representation for the general public. This research was provoked by the fact that matrices are mainly use in engineering area, while node-link graphs are generally a more popular way of visualisation. The research questions being examined were related to the attributes of the connectivity model influencing readability and which representation is more suitable for particular tasks. In addition, they filled the gap in previous research that did not take into consideration participants' familiarity with the data sets. It was found that error rates and response time are highly influenced by size and density. They confirmed Ghoniem et al () findings that matrix-based graphs outperform node-link graphs for dense and large graphs, and node-link graphs are more readable for small and sparse graphs. The only exception found is finding the path between two nodes in a graph. Furthermore, experience with the data set showed to has effect on performance.

The main advantages of a matrix-based graph are that it has no overlapping and can be ordered. Therefore, it was predicted that tasks involving link finding and node finding will be better performed in matrix-based graphs rather than node-link graphs. Furthermore, node-link and matrix-based graphs were compared [12]. They also predicted that counting nodes should be equally difficult on both, while counting links and finding the most connected node should be easier in matrix-based representations. Node-link graphs should be better when building a path between two nodes. The reason for this is that matrix based graphs have the nodes represented twice which introduces extra complexity. Ghoniem and colleagues [12] also hypothesised that node-link graphs will be easier to work with when dealing with graphs with a small amount of graphs. In their experiment they had three sizes of graphs - 20, 50, and 100 nodes. They tested 36 participants and measured their performance on various tasks such as finding paths, neighbours, nodes and links between nodes. They found that performance in node-link graphs decreases as the size of the graph increases. This pattern was confirmed for all of the tasks except for finding path, where node-link graphs regardless of the size and density showed better results.

The present research aims to compare readability in different layouts in node-link and matrix-based graphs representing causal relationships. Following the literature review it is intriguing to explore which representations will show better performance and whether this will be modulated by factors such as layout, question domain, size and task.

Chapter 3

Implementation details

3.1 Software tools and technologies

Given the opportunity to chose any tools and technologies for the development of this web application was a very exciting task. However, I had to be certain that the right decisions are made. After a research period followed by a trial-error week it was decided that the backend of the application will built with Node.js and JavaScript combined with the web application framework Express. Node.js was chosen on the grounds of being event-driven, non-blocking I/O model which contributes to a very efficient and lightweight software. A Node.js JavaScript engine is also used in the Google Chrome browser. JavaScript servers have incredible performance due to their asynchronous I/O. Node.js appears to be single-threaded from a developer's point of view, as there is no thread management involved in the development process. However, behind the scenes Node.js handles threading, file system events, implements the event loop, feature thread pooling etc. Coming from a Java background, the Maven equivalent in Node.js is NPM. By using NPM commands the developer is able to install variety of different modules to help the implementation process. NPM executes the function of a package manager. Express is the standard server framework for Node.js. It is usually described as a minimal and flexible Node.js web application framework. Many popular frameworks such as KeystoneJS, Kraken and Sails, are built on Express.

AngularJS 1 was chosen for management of frontend functionality. Even though there is a newer version of the product, the lack of documentation and support online, was a sufficient reason for using the older AngularJS. It uses HTML as a template and enables the developers to extend it to express the application's components clearly. AngularJS supports features such as data binding and dependency injection which decreases the amount of code that a developer would usually write to implement them.

The database system chosen for the project is PostgreSQL. Considering the size of the project an object-relational database was chosen. In addition, it decided on PostgreSQL in particular because it is open source and has gained a reputation of a reliable database system. Also previous experience with PosgreSQL from developers point of view made the decision easier.

* maybe database schema will be added here *

3.2 Development Process

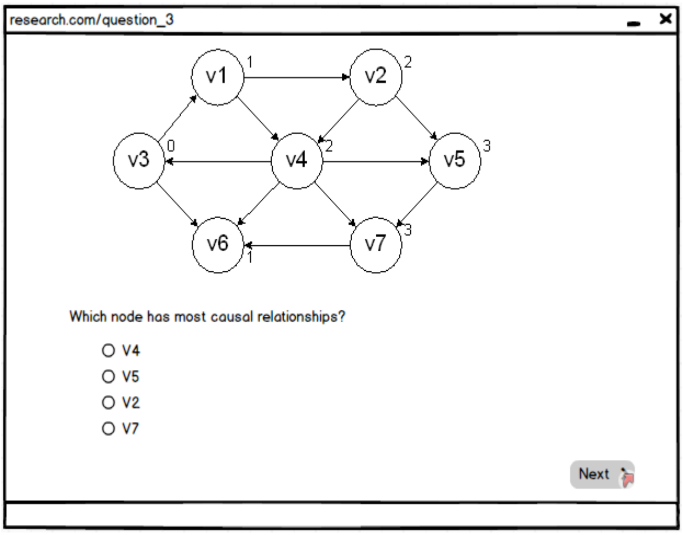
3.2.1 User Stories

The first step of the implementation was understanding the requirements and creating user stories. Some of the most important user stories are:

- As a researcher, I want to be able to see participant's answers to questions, so that I can analyse the data.
- As a participant, I want to be able to see a graph and a questions at a time, so that I can complete the experiment.
- As a researcher, I want to keep participant's scores anonymous, so that the experiment comply with ethics requirements.
- As a participant, I want to be unable to go the next question before completing the present question first, so that I am certain that I haven't missed a question.
- As a researcher,

3.2.2 Wire-frames

After the requirements gathering analysis, development of wire-frames followed. Balsamiq Mockups 3 is the software used for the creation of wire-frames. An example of the research question page can be found in figure 3.1.



Which node has most causal relationships?

○ V4
○ V5
○ V2
○ V7

Figure 3.1: Research Question Wire-frame

After discussing the wire-frames it appeared that some important features are missing. One of those features was a participants training session. The aim of the experiment is to test which graph is more readable for people who do not have regular exposure to such data visualisation. Therefore, it is important to make the participants aware of how to read each graph before the actual experiment. This way, the requirements specification became an iterative process during which a better understanding of the product evolved.

3.2.3 System Design

Designing the system was the next stage in the process.

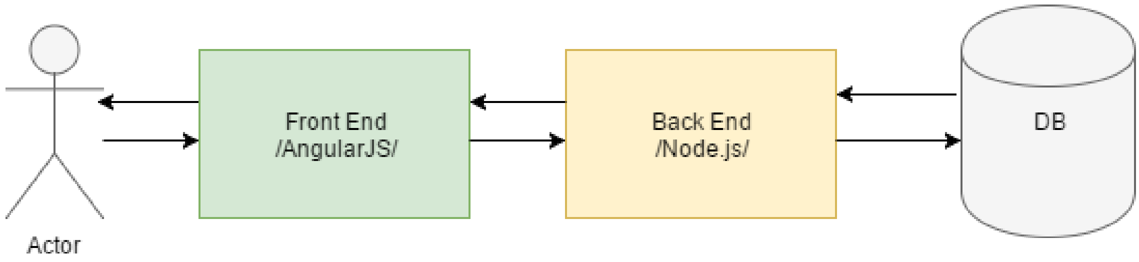


Figure 3.2: An abstract representation of the system design

Figure 3.2 shows an abstract view of the system design. There is an Actor who will either be a participant in the study or a researcher. They will interact with the front-end which will be in the form of a web application in a browser. The front end will communicate with the back-end which will be implemented in Node.js. The back-end will make requests to the database to retrieve and send information.

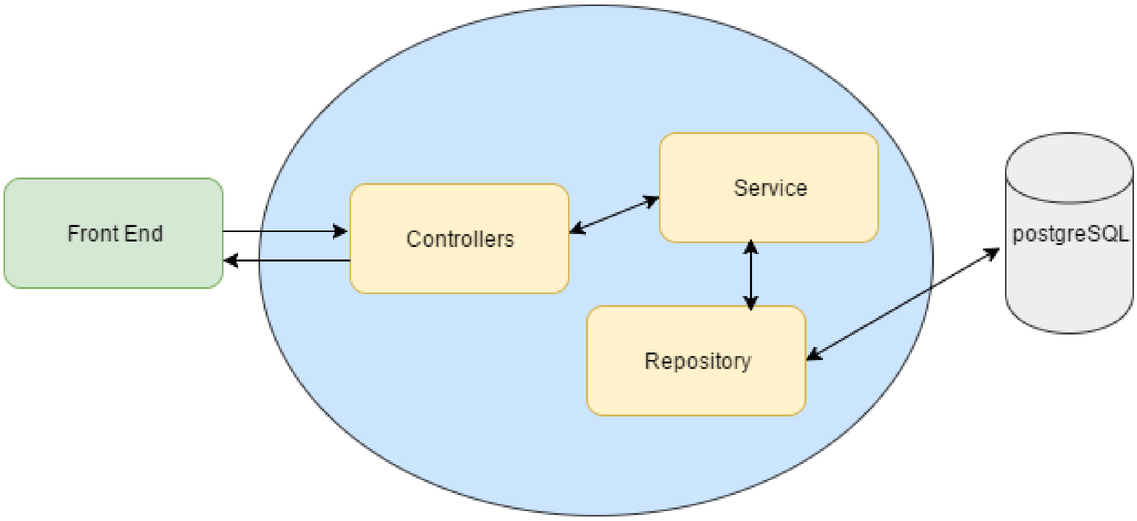


Figure 3.3: A more specific representation of the system design

Figure 3.3 displays a more detailed version of the system design. This particular design has been implemented to separate the different concerns in this specific system. When the Actor interacts with the application, the front-end will send information to the Controllers. There are many controllers because there is a controller for each page with front-end functionality. The Controller component decides what should the next action be according to the user input. It has control over the front-end logic and sending requests to the service if information from the database or the server is requested. For example, the answers to all questions are kept in the front-end until a ”Submit” button is clicked on. This action triggers a request to the Service. The Service component works with the back-end logic. It can send and retrieve data from the database and keep the information in the Repository. The Service also deals with the requests for the different web-pages. Also it ensures that the project dependencies are loaded.

3.2.4 Implementation

The development process was split into front-end and back-end. Without using any frameworks, the front-end Html pages were created following the wireframes. Bootstrap was added to the html to improve the UI design and make it look more appealing for the participant. Furthermore, a hierarchical page set up was implemented by having all html pages extending one layout file. This also helped avoid repeated code as needed libraries were imported only once and all other files inherited them. A considerable amount of time was spent on developing the graphs and converting node-link graphs into matrix-based graphs. The different sizes were evolving from each other, which means that the information contained in the small graph was present in the medium one. Close to piloting the experiment, the front end had to be drastically changed in order to fit the large graphs on one screen.

By this time, it was clear that Node.js will be used to create a server so the next step was to implement it. The decision to use Express as a framework with Node.js followed and the html pages were mapped to a handlebars or hbs files.

The following couple of weeks were dedicate on work on the database system: creating database schema, and tables, and work on connecting it with the server.

3.3 Challenges

```
Spring idea failed
changed to Node.js
Angular compatibility with Node.js
```

In the beginning of this project the Java framework Spring was going to be used in the implementation as it is among the most widely used frameworks in industry [32]. This decision was supported by extensive previous experience with Java from developer's point of view and the applicability of the skills to be acquired. However, one of the reasons why Spring is used in industry is because of the large and complex systems that exist there. The Spring framework works on a very high level of abstraction where you can easily write configuration files to add dependencies from different project. Therefore, it is considered rather unfriendly for small independent projects and developers with limited Spring experience. The reasoning behind this conclusion was provoked after a couple of unsuccessful attempts to set relative paths to different CSS and JavaScript files. The issue was found to be in the web application configuration file. This is how the very simple task of reading a css file turned to be a long tedious debugging process after which the realisation that Spring is unnecessary abstract and complex for this project occurred.

A new research for web-application frameworks followed. Node.js backend was chosen because of its event-driven, non-blocking I/O model which creates an efficient and lightweight server-side of the application. Another challenge appeared when trying to incorporate AngularJS with Node.js. Usually in AngularJS one uses curly braces to reference data structure from the AngularJS controller. However, Node.js also uses curly brackets to reference information from the backend in the frontend. After a long research it was found that Node.js overrides the use of curly braces and the application is not displaying Angular data as it expects it come from the backend. Unfortunately, an appropriate error message does exist and it all had to be discovered during the development process. Instead of using curly brackets one can also use ”ng-bind” and achieve the same result. This approached solved the issue until ”ng-bind” information was need in ”ng-src” to display the appropriate graph image. It is not possible to use ”ng-bind” inside ”ng-src” so the present solution at the time was no longer solving the problem. Therefore, the Angular configurations had be altered to use a different symbol. Implementing this solved the problem entirely.

Another issue was the size of the graphs. During the implementation process it was necessary that a scroll bar appears in order to show the real size of the graphs. This was inconvenient for reading the graphs, and participants were going to need more time to read these graphs. Therefore, this was going to be a confounding variable in the experiment. In order to avoid this issue, the front-end pages had to be restructured. The bootstrap navigation bar had to be removed and the question had to be displayed on the left instead of below the graph. This also meant that this particular sequence of pages had to extend a different layout page, which does not contain the navigation bar and has a different bootstrap grid.

3.4 Software reliability testing

Chapter 4

Experiment

4.1 Design

This is a within subject design experiment with two conditions. In the first condition participants answered questions related to causal relationships in node-link graphs, while in the second one they were asked to answer questions on matrix-based representation. Each condition had four levels - size, layout, question type and question domain. The dependent variables were time taken to answer each question and correctness of the answer. The participants were also given a questionnaire which required subjective answers about their graph preferences and how they enjoy solving logical problems.

4.2 Stimuli

Figure 4.1 illustrates the pattern which was used create the questions and the graphs. The domain questions were healthy and unhealthy gym behaviour, drinking issues, and student exams. The three different sizes were small (10 factors), medium (20 factors) and large (30 factors). There were three different layouts for each type of representation. The matrix-based layouts were alphabetical, in degree descending and out degree descending. The node-link layouts were hierarchical, orthogonal, and series parallel. There was a direct relationship question and an indirect causal relationship question asked for each of the created graphs. The first six questions of the experiment were training questions which did not count towards the final results. The characteristics of the graphs in these questions were chosen to be different from the graphs in the experimental questions. They were added to the beginning of the experiment which generated a total of 42 questions.

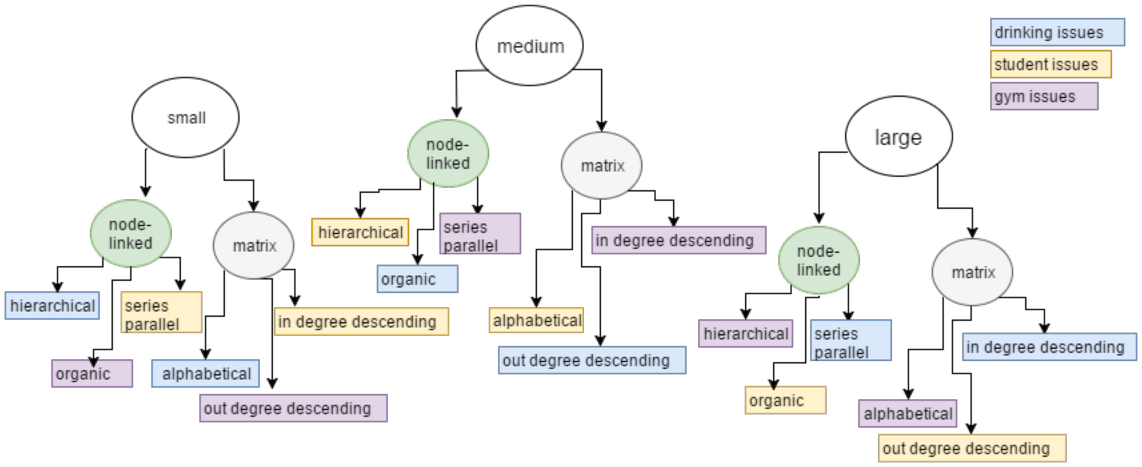


Figure 4.1: Experimental Design

The specified sequence of questions with matched layouts, question themes, and different sizes was likely to influence the results of the experiment because of the exposure effect of that particular sequence. Therefore, to ensure that all participants receive the questions in a different order, Latin square /citewiner1962latin randomization was used. The sequence of questions was different for each participant and the possibility of the questions sequence affecting the results was eliminated.

Node-link graphs C.1 were created using yEd and the equivalent matrix-based representations were generated using software which converts XGML node-link graphs into matrix-based graphs C.2. As long as the graphs are from the same domain question and size, they represent the same information. However, all questions require different information to be read from the graphs to avoid learning effect. The representation of identical information ensures that the complexity of the data is the same. Both types of representations were displayed in their real size on the left side of the screen and the corresponding question was on the right. A special attention was paid to the letter's font size and retaining the original size of the image on one screen. Participants were not able to go to the next question before they answered the current one. In addition, to measure answer time more accurately, there was no "Next" button. Once a participant clicked on an answer it was automatically submitted and the next question was loaded on the page. Figure C.3 in Appendix C shows an example of a trial question.

A Google Form questionnaire was used to ask participants to reflect back on their experience. Two weeks after the testing has started the participants were sent an email with the subjective questionnaire about their graph preference.

4.3 Pilot

Conducting a pilot was an essential part of this study. It brought light into how people who have never been exposed to the graphs and the software before interact with them. It was important to learn whether the tasks are clear and the software is easy to use. Three volunteers took part in the pilot. They were asked to sit on a chair in a silent environment and complete the experiment starting from the information sheet and finishing with the debrief form. At the end the participants were asked questions regarding the training, the clearness and enjoyment of the tasks. All participants said that the training in the beginning of the experiment is sufficient at explaining how to read the graphs. One of the participants, however, required confirmation that the correct direction of reading the graphs is from left to top. This is why, it was decided to make the first three questions trials in which the participants can ask as many questions as they need to understand how to read the graphs. The second three questions (4,5,6) were also going to be practice questions but they were not going to be explicitly informed about it. Consequently, during the first six tasks the participants were expected to fully learn how to interpret the graphs and their answers were not recorded in the results.

An interesting problem regarding the nature of the trial questions was also considered. Initially, it was suggested that the trial instances are always the first questions of each sequence. However, as those questions would not be accounted for in the results and the Latin Square randomizes each sequence, this was going to lead to uneven number of answer for each type of graph. If this was to happen, the results of the experiment were going to be negatively influenced. In order to avoid this, 6 new questions were created using a combination of layouts, sizes and domains that has not been used for the original 36 questions. A mixture of different sizes and types was included to ensure that the participants have exposure to the main challenges of the experiment during the practice trials. These 6 questions were going to be added at the beginning of each sequence so that all participants are exposed to the same trial questions.

All participants in the pilot said that the tasks are clear, the graphs' font size is readable, and different layouts and sizes are appropriately displaying the data. One participant mentioned that in the node-link organic layout, the label is on the arrow which makes it hard to read. The reason for this is that both labels and arrows are in black colour. Their note was taken into consideration but if changes are made to align the arrow and the label differently this would alter the organic yEd layout. Consequently, the results will provide information about a manually created layout similar to the organic yEd layout. Another solution of this issue was to change the label names to individual letters like ”A” instead of actual factor labels such as ”depression”, and ”pass exams”. However, this was going to influence the complexity of the experiment and the participant's level of interest. Thus, no changes were made to the organic layout.

Furthermore, the pilot brought light into how entertaining the tasks are. It is vital to have the participants engaged with the tasks to ensure that they are concentrated and not easily distracted by third factors. Two participants said that the experiment was interesting and were eager to answer the questions. They said that the topics were easy to understand and it was interesting to find the causal relationship. One participant said that they did not consider the task to be very interesting, but they were in a hurry for another meeting and likely to look for an excuse to leave earlier.

One volunteer in the pilot mentioned that sometimes it is hard to find the label you are looking for and they guessed the answer to the question by logical reasoning. To avoid this problem in the experiment, a note was be made to tell participants that there is no logical relationship between the causal relationships and they should not attempt to guess the correct answer. This is supposed to encourage them to look for the labels rather than guess the correct answer. It was also noted, that when a particular node has more than one causal relationship, it is much harder to follow a path, than in a complex matrix-based graph with many relationships. It will be interesting to see whether this observation will be confirmed by the experimental results.

The pilot was also extremely helpful for spotting technical issues. If it was not conducted and some of these problems were not accounted for, they were going to have catastrophic influence on the results. First, it was found that in one question the correct answer was being evaluated to incorrect because of a typo in the csv file that was used to populate the database. In addition, the answers were not written in a consistent way, which meant that the participants might had been influenced to choose the answer that is written differently from the rest of the answers. In order to fix these problems, the database table with the questions had to be altered.

Another technical issue found was concerned with the time recorder. It was found that the timing was starting and stopping when required, but the record of the time taken for a particular question was wrong. The issue was found to be caused by a wrong startTime variable during the timeTaken calculation. This was a scope issue, which was fixed for the real experiment. Moreover, in order to submit their answers the participants had to click a ”Next” button. The timer used to stop once this button is clicked, record the time taken for the current answer, and start the timer for the next question. However, it was found that this extra click influences the recorded time and it would be more appropriate to have the time start and stop ones the participants choose an answer. This led to the decision to completely remove the ”Next” button and submit the answers when one of the radio buttons has been selected. On that click the next graph and question would be loaded. It was not expected from the participants to expect a submission of their answer on their first click so thorough instructions about this feature were added in the information sheet. In addition they were to have six trial questions to get used to that functionality.

In general, the pilot helped in identifying small issues regarding the graphs layout and the manner in which questions were asked. Other bigger issues, that were going to influence the results, were also found such as typos leading to wrong evaluation of correctness and inaccurate time recorder. The pilot also inspired a new way in which the trial questions should be created and accounted for. The present experiment would not have been as accurate and precise if the pilot was not conducted. If the changes that it provoked implementing were to be found in the process of testing participants, the testing had to be started again and new participants had to be recruited.

4.4 Participants

There were 30 participants who took part in the experiment, aged between 20 and 29 years. The excluding criteria restricted people specialized in the subjects of Maths, Engineering and Computing Science from participating. The reason for this is that these fields are likely to include preparation in reading graphs. In addition, findings are expected to be representative for Health and Social sector employees, who are not likely to have background in those degrees.

4.5 Procedure

In order to have a more environmentally valid experiment, the study was conducted only on the experimenter's laptop in their presence. The participants were asked to sit on a chair and complete the experiment in a calm environment. They were recruited either in Glasgow University library or by posts in social media websites. The experiment was conducted in GU library in quiet group study areas. The participants had to read through an information sheet, explaining what is to follow in the next 30 minutes, a consent from, asking them to give their written informed consent, and a training showing them examples of how to solve the upcoming 42 research questions. At the end of the experiment participants were shown a short demographic questionnaire and a debrief form. Two weeks after the testing has started they were sent another questionnaire asking about their graph preferences.

4.6 Results

4.6.1 Data Analysis Methodology

Correct data analysis is vital for the successful completion of a research based project. This is why it was decided to complete a pilot of the data analysis methodology with fake data before all participants were tested. In order to achieve this the results of 20 participants were faked. As this is a repeated measures experiment the data layout had to be in wide format with each participant's data represented on each row. However, the way the database was being populated was in long format. Therefore, data wrangling had to applied before the data was in a format able to be statistically analysed. The data analysis was done with R Studio which provides a package for data wrangling. However, as there were a number of different factors defining each answer, and thousands of lines of raw data, it was decided that writing SQL queries would be a more appropriate way to deal with the data layout.

For each type of comparison, there was a query written to aggregate the needed information in a format ready to be inserted in R Studio. Then each group of data was tested for outliers. It was decided that if outliers were to be found, it is important to find out whether there is a reasonable explanation for their existence. If so a decision to remove or keep them was to follow.

The Anderson-Darling normality test was conducted to check whether the time data was normally distributed. Additional reading suggested that R Studio has other normality tests which are more popular and have better reputation. One such test is Shapiro-Wilks Normality Test. Both tests showed a p-value of less than 0.05 for the fake data which rejected the Null hypothesis that the data is normally distributed. However, for the real data only Shapiro-Wilk test was going to be used. Histograms were plotted using the hist() function. This was found to be helpful to visualise the distribution of the values. If the data values were normally distributed then a repeated measures t-test was going to be run to see whether there is a significant difference between the time taken to read the two types of graphs. However, as the data was found to be not normally distributed, a non-parametric test had to be used. The non-parametric equivalent of repeated measures t-test is Wilcoxon. R Studio has a library containing the test so one command had to be executed to see whether there is a significant difference between node-link and matrix-based graphs.

The type of representation is the main aspect being investigated. However, there are also other research questions that need to be answered. For example, the different sizes and layouts of each type of graph can be compared. Therefore, a comparison between the three layouts for each size could also be done. This means that there will be three factors and a different statistical test needs to be used. For non-parametric data this test is Friedman followed by post-hoc comparison in case of significant difference. The data was faked so that a significant result was found and there was a need to investigate which are the pairs of comparisons significantly different from each other. It was found that there is a function, friedmanmc(), which is no longer built in in R Studio but performs this calculation. This is why, the implementation of the function was found and imported in the working directory. However, this function returns True or False for each pair and not the actual p-value. As the p-value was essential for the experimental results, a new function was found from the PMCMR package which calculates it. An issue which arises here was that the two functions were reporting different results with the second being more likely to calculate significant p-value. A deeper investigation in the way two functions work was made and it was found that the second function is not calculating the adjusted p-value which was leading to a difference in results. In order to account for this Bonferroni adjustment was added as a parameter to the calculation. This is how the process for calculating significance between 3 non-parametric factors was finalised. Furthermore, there is a measurement of correct answers and this is a different type of data - boolean. For example, in the fake data there were 45 percent error rate in the matrix-based graphs and 16 percent errors in the node-link graphs.

The fake data analysis took longer than expected due to the unforeseen circumstances of data layout difficulties, inexperience with R Studio, and statistical analysis of large amounts of data. However, because of this process, a clear layout of the analysis for the real data was prepared. All steps of the process were clear beginning with SQL queries to aggregate data, to inserting data in R Studio, and running all statistical tests.

4.6.2 Experimental Results

The data analysis was first done for direct questions and then the same scrip was used to do the analysis for path questions. Therefore, the results section also consists of two different blocks of analysed data relating to each type of question.

Direct Questions

The research nature of the experiment allowed for thorough data analysis of different factors in the graphs. The results showed that regardless of the graph layout and size the matrix-based graphs (M = 14.15) have better performance than node-link graphs (M = 16.85) for direct questions (Fig4.2) with p-value = 0.0128 (Table4.1).

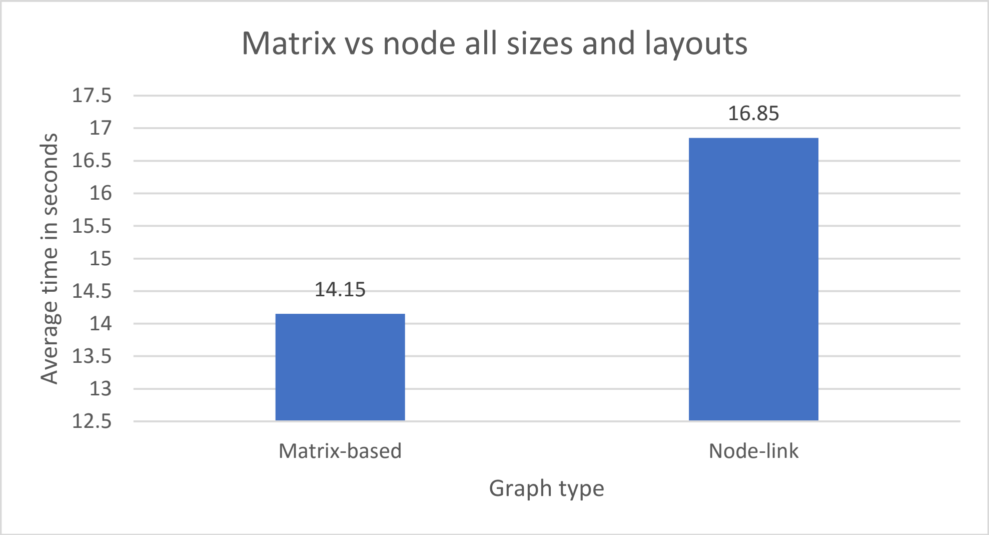


Figure 4.2: Direct Question: Matrix vs Node-link

When different sizes of each graph type were compared significant difference was found only for the large graphs comparison with matrix-based large graphs (M = 16.03) being easier to read than node-link graphs(M = 24.76). Figure4.3 shows the mean values of each each representations. Wilcoxon test showed a p-value of 5.633e-07.

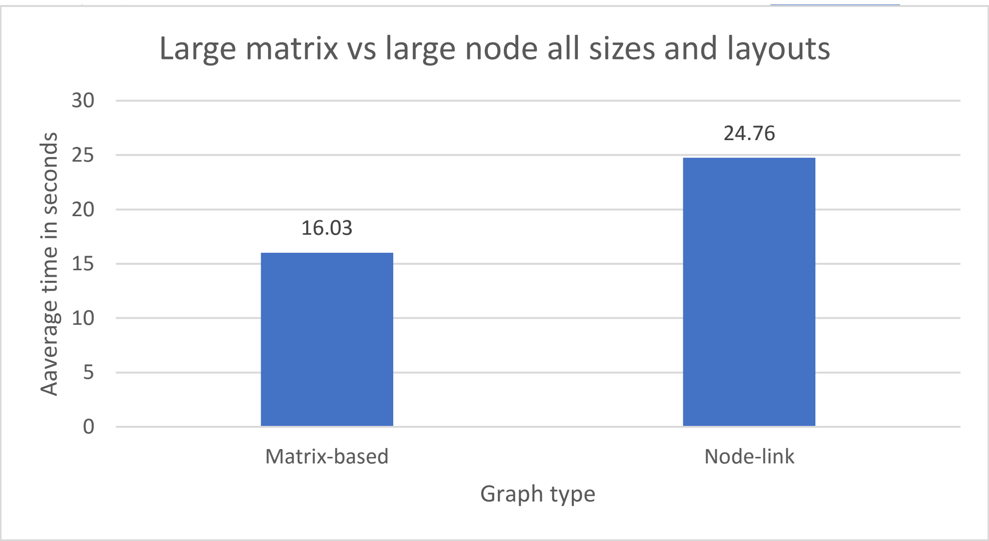


Figure 4.3: Direct Question: Large Matrix vs Large Node-link

Node-link layouts were compared in each size group. Figure4.4 illustrates the difference in mean values in all node-link layouts and sizes. In small size the parallel layout was found the have the worst performance (M = 14.20) and hierarchical layout was found to have the best performance (M = 9.65). Friedman test calculated a p-value of 0.0013* with significant post-hoc tests between all pairs (Table4.1). In the medium size measures, the parallel layout had the worst performance time of 15.62 on average, but when Friedman test was conducted no statistical difference between the three layouts was found with p-value = 0.6575. The node-link large size performance time for hierarchical (M=17.96), parallel (M=38.86), and organic (M=17.46) layouts 4.4 was significantly different and has p-value = 3.261e-06*.

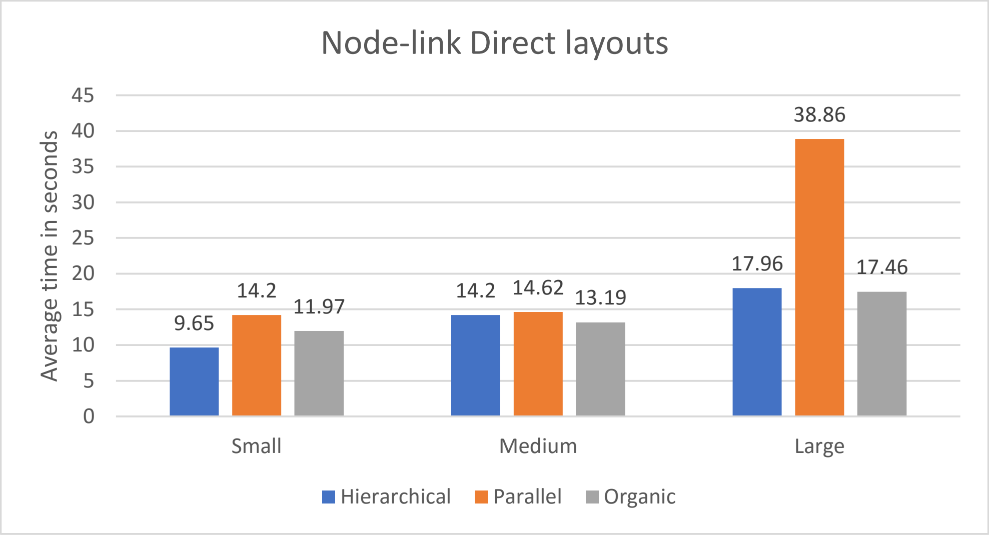


Figure 4.4: Direct Question: Node-link layout comparison in all size groups

The matrix-based based graphs were not found to have consistent performance differences between the various layouts when direct question was asked. Significant difference was found in the medium size with out degree descending layout (M=12.70) outperforming in degree descending (M=14.64) and alphabetical layouts (M=14.86). As far as the small and the large sizes are concerned p-value was shown to be insignificant.

Table 4.1: Direct Question comparisons

<table>
<tr><th colspan="3">Matrix</th><th colspan="3">Node-link</th><th>P-value</th></tr>
<tr><td colspan="3">14.15</td><td colspan="3">16.85</td><td>0.0128*</td></tr>
<tr><td colspan="3">small 10.78</td><td colspan="3">small 11.92</td><td>0.1055</td></tr>
<tr><td colspan="3">medium 14.06</td><td colspan="3">medium 14.01</td><td>0.8776</td></tr>
<tr><td colspan="3">large 16.03</td><td colspan="3">large 24.76</td><td>5.633e-07*</td></tr>
<tr><td colspan="3" rowspan="12"></td><td colspan="4">small</td></tr>
<tr><td>H 9.65</td><td>P 14.20</td><td>O 11.97</td><td>0.0013*</td></tr>
<tr><td>H</td><td colspan="2">P</td><td>&lt;2e-16*</td></tr>
<tr><td colspan="2">P</td><td>O</td><td>1.8e-09*</td></tr>
<tr><td colspan="2">H</td><td>O</td><td>3.5e-05*</td></tr>
<tr><td colspan="4">medium</td></tr>
<tr><td>H 14.20</td><td>P 14.62</td><td>O 13.19</td><td>0.6575</td></tr>
<tr><td colspan="4">large</td></tr>
<tr><td>H 17.96</td><td>P 38.86</td><td>O 17.46</td><td>3.261e-06*</td></tr>
<tr><td colspan="2">P</td><td>O</td><td>&lt;2e-16*</td></tr>
<tr><td>H</td><td colspan="2">P</td><td>&lt;2e-16*</td></tr>
<tr><td>H</td><td colspan="2">O</td><td>0.001*</td></tr>
<tr><td colspan="3">small</td><td colspan="3" rowspan="10"></td><td></td></tr>
<tr><td>inDD 10.35</td><td>outDD 9.25</td><td>alpha 12.75</td><td>0.07939</td></tr>
<tr><td colspan="3">medium</td><td></td></tr>
<tr><td>inDD 14.64</td><td>outDD 12.70</td><td>alpha 14.86</td><td>0.03943*</td></tr>
<tr><td>inDD</td><td colspan="2">outDD</td><td>2.4e-10*</td></tr>
<tr><td colspan="2">outDD</td><td>alpha</td><td>7.5e-07*</td></tr>
<tr><td colspan="2">inDD</td><td>alpha</td><td>0.12</td></tr>
<tr><td colspan="3">large</td><td></td></tr>
<tr><td>inDD 15.34</td><td>outDD 14.76</td><td>alpha 18.00</td><td>0.355</td></tr>
</table>

H - hierarchical
P - parallel
O - organic
inDD - in degree descending
outDD - out degree descending
alpha - alphabetical

Path Questions

The path question results showed different pattern. The Wilcoxon comparison between the matrix-based (M=47.37) and node-link (M=39.51) representations of any size and layout showed a better performance for node-link graphs with p-value=0.02847*. In the small group size of this question type, matrix-based representations (M=23.09) were outperformed by node-link graphs (M=27.83). Wilcoxon test reported a p-value of 0.03997* (Table4.2). No difference was found in the medium size graphs, but in the large size group matrix-based graphs (M=47.38) showed worse performance when compared to node-link graphs (M=39.51). Wilcoxon test found a p-value of 0.02847*. The node-link layout comparison showed that for small sizes the hierarchical layout (M=27.31) has worst performance, followed by the parallel layout (M=22.43) and the organic layout (M=19.62). Friedman test found a significant difference between the layouts with p-value=5.943e-06*. However, in the medium size group the layouts showed different results (Figure4.5). The organic layout (M=48.55) performed significantly worse than the parallel (M=26.44) and the hierarchical (M=23.54) 4.2. These results were consistent with the large group size layouts where the organic layout (M=47.88) showed worse performance than parallel (M=39.20) and hierarchical (M=31.44) layouts.

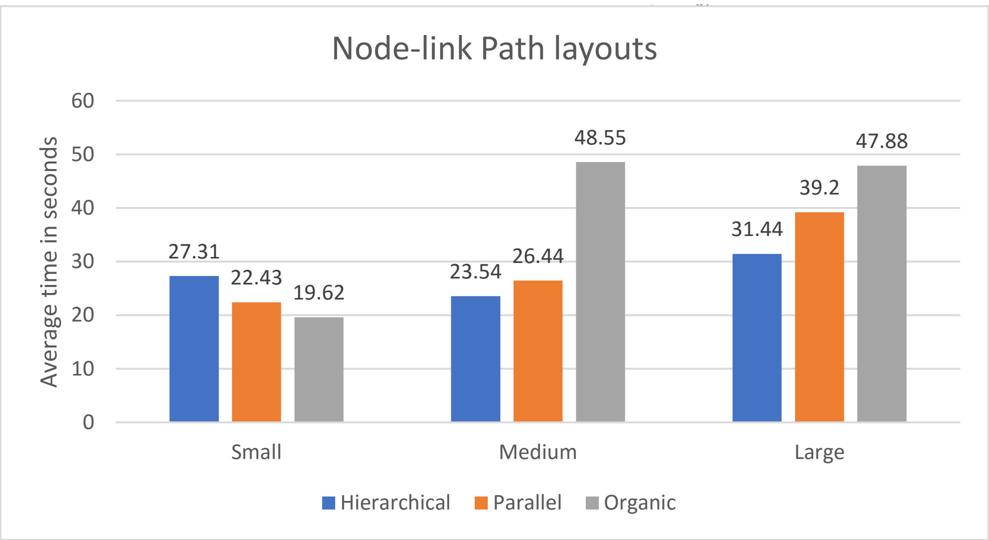


Figure 4.5: Path Question: Node-link layout comparison in all size groups

Among the matrix layouts significant difference was found in the small and large size groups. When the size is small, alphabetical layout (M=23.79) outperformed out degree descending (M=29.50) and in degree descending (30.20) layouts 4.6. A Friedman test found a significant difference between the three layouts with p-value=0.0003245*. In the medium size group, the alphabetical layout was confirmed to have best performance (M=27.29) when compared to in degree descending layout (M=29.07), but not statistical difference was found when compared to out degree descending layout (M=28.68) with p-value= 0.26. No significant difference was found among the layouts in large size. All p-values for path question can be found in Figure 4.2

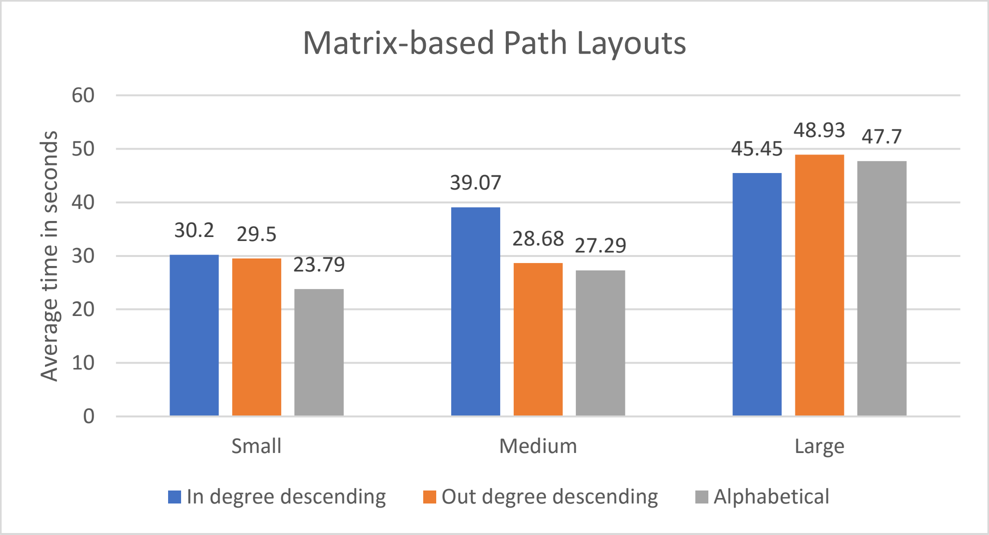


Figure 4.6: Path Question: Matrix-based layout comparison in all size groups

Table 4.2: Path Question comparisons

Matrix			**Node-link**			**P-value**
47.37			39.51			0.02847*
small 27.83			small 23.09			0.03997*
medium 31.67			medium 32.85			0.2665
large 47.38			large 39.51			0.02847*
			small			
			H 27.31	P 22.43	O 19.62	5.943e-06*
			H	P		4.5e-12*
			P		O	4.5e-12 *
			H	O		<2e-16*
			medium			
			H 23.54	P 26.44	O 48.55	3.08e-09*
			H	P		2.40e-13*
			H		O	<2e-16*
			P		O	<2e-16*
			large			
			H 31.44	P 39.20	O 47.88	0.0273*
			P		O	1.1e-09*
			H	P		1
			H		O	1.1e-09*
small						
inDD 30.20	outDD 29.50	alpha 23.79				0.0003245*
inDD	outDD					1.0e-09*
outDD		alpha				<2e-16*
inDD	alpha					1.7e-07*
medium						
inDD 39.07	outDD 28.68	alpha 27.29				0.001454*
inDD	outDD					4.5e-13*
inDD		alpha				7.7e-16*
outDD		alpha				0.26
large						
inDD 45.45	outDD 48.93	alpha 47.7				0.356

H - hierarchical
P - parallel
O - organic
inDD - in degree descending
outDD - out degree descending
alpha - alphabetical

Chapter 5

Discussion

5.1 Conclusion

Appendices

Appendix A

Database Schema

```
participants_answers(question_id, participant_id, answer, time)
participants_info(participant_id,participant_name, email, uni_degree, age)
questions(question_id, question, one, two, three, four, correct, image)
```

Appendix B

Running the Programs

An example of running from the command line is as follows:

```
> java MaxClique BBMC1 brock200_1.clq 14400
```

This will apply $BBMC$ with $style = 1$ to the first brock200 DIMACS instance allowing 14400 seconds of cpu time.

Appendix C

Graphs

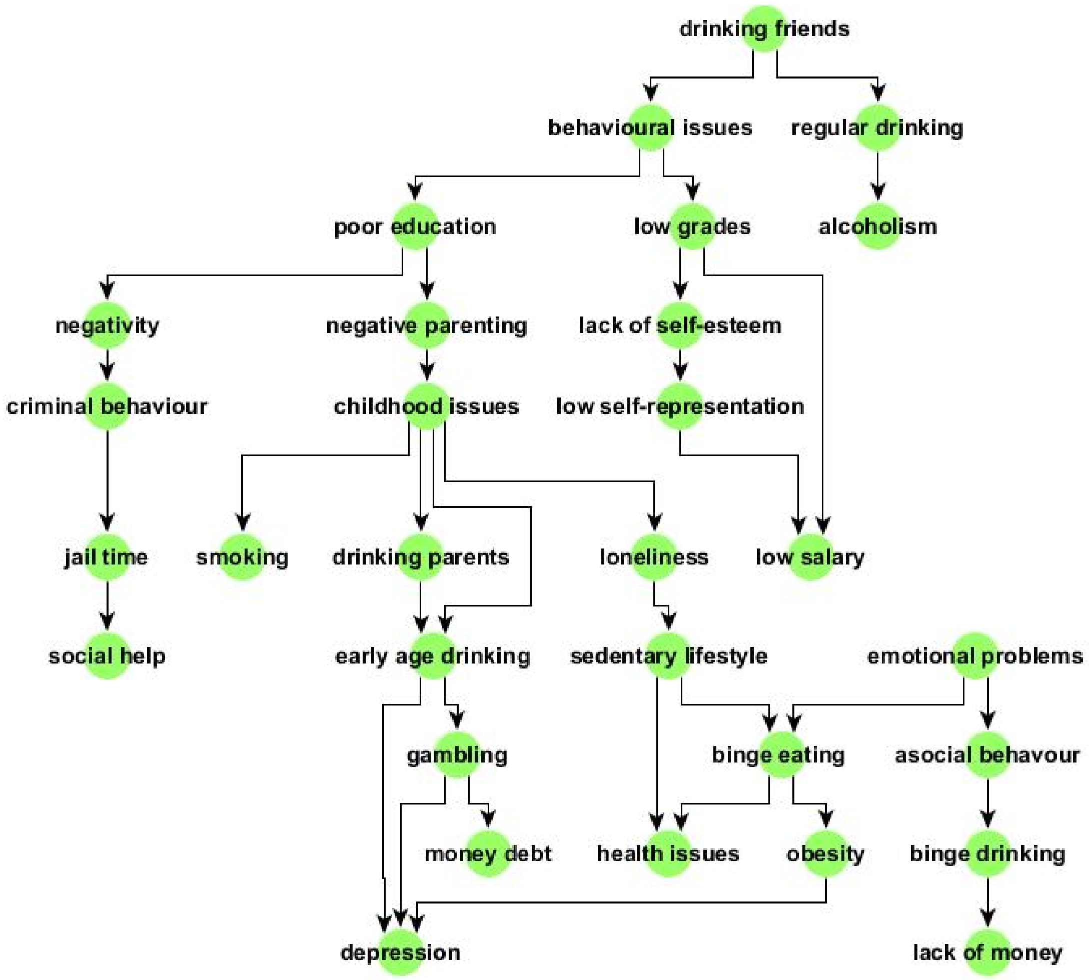


Figure C.1: Large node-link graph in hierarchical layout

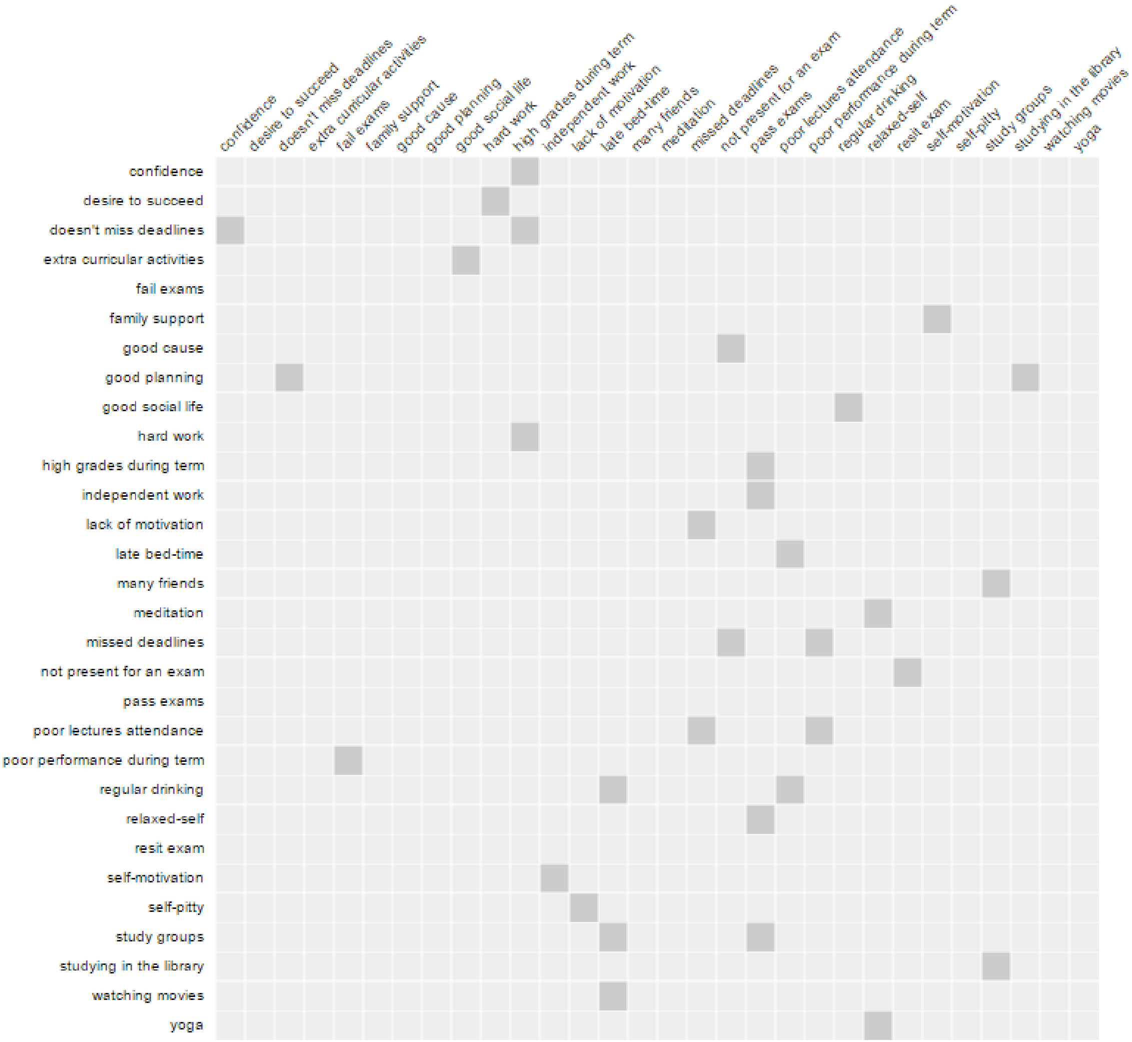
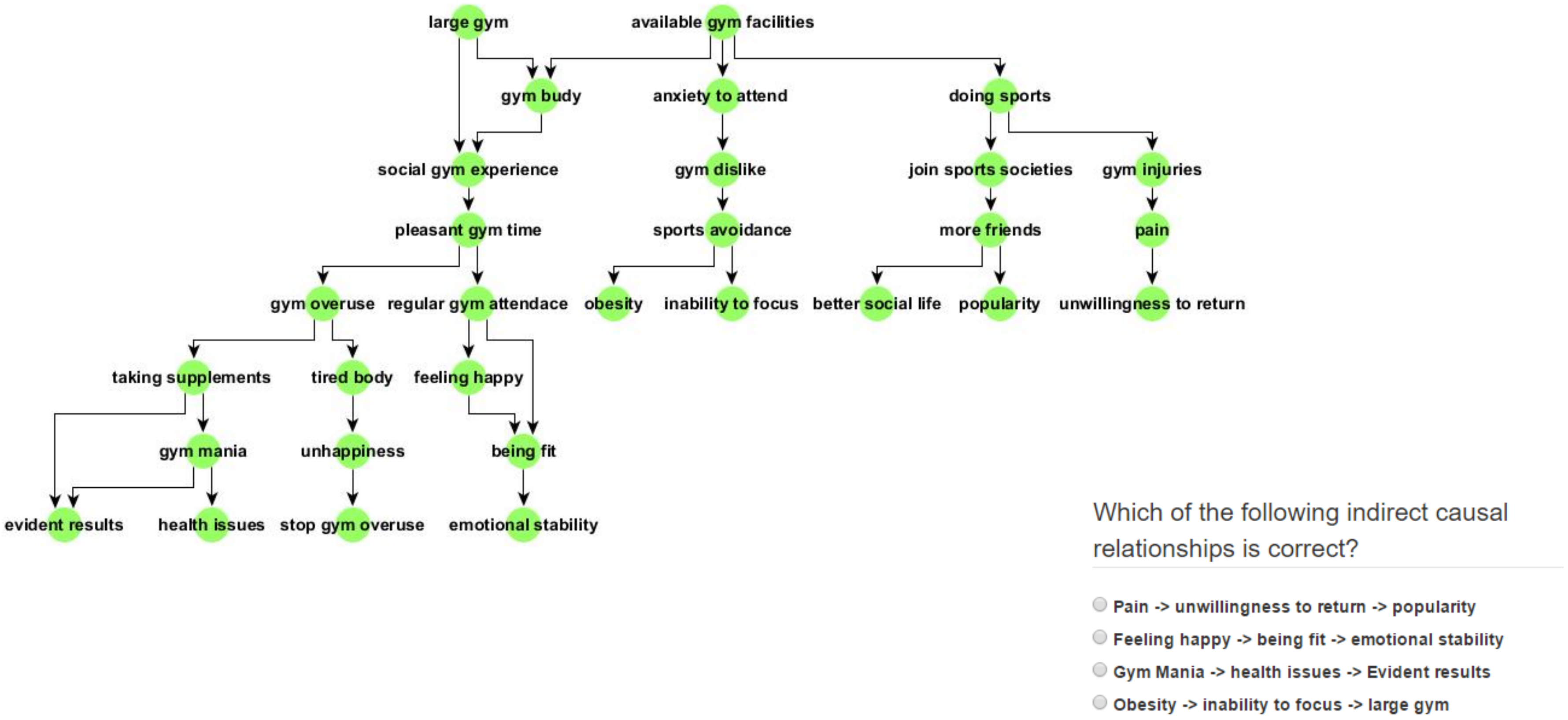


Figure C.2: Large matrix-based graph in alphabetical layout



Which of the following indirect causal relationships is correct?

○ Pain -> unwillingness to return -> popularity

○ Feeling happy -> being fit -> emotional stability

○ Gym Mania -> health issues -> Evident results

○ Obesity -> inability to focus -> large gym

Figure C.3: Example of an experimental question

Bibliography

[1] James Abello, Frank Van Ham, and Neeraj Krishnan. Ask-graphview: A large scale graph visualization system. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):669–676, 2006.

[2] Basak Alper, Benjamin Bach, Nathalie Henry Riche, Tobias Isenberg, and Jean-Daniel Fekete. Weighted graph comparison techniques for brain connectivity analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 483–492. ACM, 2013.

[3] Katy Börner, Adam Maltese, Russell Nelson Balliet, and Joe Heimlich. Investigating aspects of data visualization literacy using 20 information visualizations and 273 science museum visitors. *Information Visualization*, page 1473871615594652, 2015.

[4] Stuart K Card, Jock D Mackinlay, and Ben Shneiderman. *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.

[5] A.J. Cardno, P.S. Ingham, A. Lewin, and A.K. Singh. Data visualization methods, July 10 2014. US Patent App. 14/200,903.

[6] Tuan N Dang, Paul Murray, and Angus G Forbes. Pathwaymatrix: visualizing binary relationships between proteins in biological pathways. In *BMC proceedings*, volume 9, page S3. BioMed Central Ltd, 2015.

[7] Kasper Dinkla, Michel A Westenberg, and Jarke J van Wijk. Compressed adjacency matrices: untangling gene regulatory networks. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2457–2466, 2012.

[8] Tim Dwyer, Nathalie Henry Riche, Kim Marriott, and Christopher Mears. Edge compression techniques for visualization of dense directed graphs. *IEEE transactions on visualization and computer graphics*, 19(12):2596–2605, 2013.

[9] Ronak Etemadpour, Robson Motta, Jose Gustavo de Souza Paiva, Rosane Minghim, Maria Cristina Ferreira de Oliveira, and Lars Linsen. Perception-based evaluation of projection methods for multidimensional data visualization. *IEEE transactions on visualization and computer graphics*, 21(1):81–94, 2015.

[10] Jean-Daniel Fekete, Jarke J Van Wijk, John T Stasko, and Chris North. The value of information visualization. In *Information visualization*, pages 1–18. Springer, 2008.

[11] Emden R Gansner, Yehuda Koren, and Stephen C North. Topological fisheye views for visualizing large graphs. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):457–468, 2005.

[12] Mohammad Ghoniem, J-D Fekete, and Philippe Castagliola. A comparison of the readability of graphs using node-link and matrix-based representations. In *Information Visualization, 2004. INFOVIS 2004. IEEE Symposium on*, pages 17–24. Ieee, 2004.

[13] Alyssa A Goodman. Principles of high-dimensional data visualization in astronomy. *Astronomische Nachrichten*, 333(5-6):505–514, 2012.

[14] Evgeniy Yur'evich Gorodov and Vasiliy Vasil'evich Gubarev. Analytical review of data visualization methods in application to big data. *Journal of Electrical and Computer Engineering*, 2013:22, 2013.

[15] Lars Grammel, Melanie Tory, and Margaret-Anne Storey. How information visualization novices construct visualizations. *IEEE transactions on visualization and computer graphics*, 16(6):943–952, 2010.

[16] Sander Greenland. Quantifying biases in causal models: classical confounding vs collider-stratification bias. *Epidemiology*, 14(3):300–306, 2003.

[17] Sander Greenland, Judea Pearl, and James M Robins. Causal diagrams for epidemiologic research. *Epidemiology*, pages 37–48, 1999.

[18] Taylor Hanken, Sam Young, Karen Smilowitz, George Chiampas, and David Waskowski. Developing a data visualization system for the bank of america chicago marathon (chicago, illinois usa). *Prehospital and Disaster Medicine*, pages 1–6, 2016.

[19] Chun-Cheng Hsu. Evaluation criteria for blog design and analysis of causal relationships using factor analysis and dematel. *Expert Systems with Applications*, 39(1):187–193, 2012.

[20] Imre Janszky, Anders Ahlbom, and Anna C Svensson. The janus face of statistical adjustment: confounders versus colliders. *European journal of epidemiology*, 25(6):361–363, 2010.

[21] René Keller, Claudia M Eckert, and P John Clarkson. Matrices or node-link diagrams: which visual representation is better for visualising connectivity models? *Information Visualization*, 5(1):62–76, 2006.

[22] John King, Kathy Sonderer, and Kevin Lynch. Cognitive benefits of a simple visual metrics architecture. In *International Conference on HCI in Business, Government and Organizations*, pages 319–329. Springer, 2016.

[23] Xiaotong Liu and Han-Wei Shen. The effects of representation and juxtaposition on graphical perception of matrix visualization. In *Proceedings of the 33rd annual ACM conference on Human factors in computing systems*, pages 269–278. ACM, 2015.

[24] William JR Longabaugh. Combing the hairball with biofabric: a new approach for visualization of large networks. *BMC bioinformatics*, 13(1):1, 2012.

[25] Christina Niederer, Wolfgang Aigner, and Alexander Rind. Survey on visualizing dynamic, weighted, and directed graphs in the context of data-driven journalism. *Proceedings of the International Summer School on Visual Computing*, pages 49–58, 2015.

[26] Judea Pearl. An introduction to causal inference. *The international journal of biostatistics*, 6(2), 2010.

[27] Zeqian Sheny and Kwan-Liu Maz. Path visualization for adjacency matrices. In *Proceedings of the 9th Joint Eurographics/IEEE VGTC conference on Visualization*, pages 83–90. Eurographics Association, 2007.

[28] Ben Steichen, Giuseppe Carenini, and Cristina Conati. User-adaptive information visualization: Using eye gaze data to infer visualization tasks and user cognitive abilities. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces*, IUI '13, pages 317–328, New York, NY, USA, 2013. ACM.

[29] Marit M Suttorp, Bob Siegerink, Kitty J Jager, Carmine Zoccali, and Friedo W Dekker. Graphical presentation of confounding in directed acyclic graphs. *Nephrology Dialysis Transplantation*, 30(9):1418–1423, 2015.

[30] Balaji T Thattai, Karthikeyan Shanmugam, Chi Yao, and Yee Lang Lau. Systems and methods for generating data visualization applications, February 24 2016. US Patent App. 15/052,449.

[31] Colin Ware, Eric Neufeld, and Lyn Bartram. Visualizing causal relations. In *Proceedings of IEEE Information Visualization*, volume 99, 1999.

[32] Kuikui Liuös Xiujin Shiás and Yue Liës. Integrated Architecture for Web Application Development Based on Spring Framework and Activiti Engine. *The International Conference on E-Technologies and Business on the Web (EBW2013)*, pages 52–56, 2013.

[33] Hao Zhang, Maoyuan Sun, Danfeng (Daphne) Yao, and Chris North. Visualizing traffic causality for analyzing network anomalies. In *Proceedings of the 2015 ACM International Workshop on International Workshop on Security and Privacy Analytics*, IWSPA '15, pages 37–42, New York, NY, USA, 2015. ACM.