Improving the usability of complex biological networks through interestingness measures and interactive visualization

Hanin Alzahrani

Supervisors:

Dr Sara Fernstad

Prof.Anil Wipat

School of Computing Science

Newcastle University

This dissertation is submitted for the degree of

Doctor of Philosophy

February 2024

Abstract

Complex biological networks are dynamic and intricate systems that reflect the fundamental processes of life. This can range from the molecular interactions in a single cell to the complex communication web between organs and tissues in multicellular organisms, where the network orchestrates different biological functions. Unravelling these complexities is critical to the development of therapeutic interventions. The interdisciplinary nature of studying biological networks involves integrating principles that cut across biochemistry, molecular biology, genetics, and system biology, thereby providing a comprehensive perspective that allows researchers to examine the intricate connections driving the complexities in living organisms.

Biological systems are constantly changing, capturing, and understanding their dynamism can be challenging. Visualization tools can help solve these difficulties by simplifying the complexity and representing the data visually and intuitively, making it easier for the researchers to identify the inherent patterns and relationships within the data.

The aim of this thesis is to examine the features and measures of interest and evaluate the usability of interactive visualization of complex biological networks. To achieve this, five objectives were formulated. First, tasks and patterns of interest regarding the analysis of biological networks were determined through a literature review, interviews, and consultations with biologists. Second, a set of metrics based on tasks and patterns was defined by demonstrating how the concepts of interestingness and visualization can support the analysis of complex biological networks. Third, the study evaluated the usability and limitations of existing network visualization methods used for biological networks to identify how the usability of complex biological networks can be improved. Fourth, a visualization tool was designed and developed that overcomes current limitations and supports human cognition and data exploration, using multiple coordinated views and interactivity guided by interestingness measures. Finally, the network visualization tool was evaluated to identify limitations and areas for improvement. Overall, the evaluation of the developed tool was positive and guided by experts' feedback, which was obtained using survey and interview techniques.

Acknowledgement

‘***To accomplish great things we must first dream, then visualize, then plan... believe... act!***’

-Alfred A. Montapert

I have accomplished and achieved what I had hoped for.

I would like to thank everyone who supported me during this time, when I desperately needed the moral support. First and foremost, I want to express my appreciation and gratitude to my outstanding supervisor for her patience and unending support for me during the PhD journey as well as her encouragement during difficult times, which played a primary role in shaping my research and personal growth. **Dr Sara Fernstad**, thank you for everything. I would also like to thank **Professor Anil Wipat** for his generous help and advice.

Second, I dedicate this research to the one who raised me in his arms and gave me all the strength and support I needed — the one who worked hard to raise me and turn me into a woman he could be proud of for the rest of his life. I hoped you would be here for this. I wish you could share my joy now that I have accomplished what you and I had hoped for. But I hope my voice reaches you, and I am happy to tell you that I have achieved success and fulfilled the promise I made to you. I dedicate this work to you, **my beloved father**, and may God have mercy on your soul.

I would not be who I am without **my loving mother**. You were the light in my darkness. You did so much for me and gave me love and tenderness. You helped me fulfil many of the dreams I had since I was a child. You were with me all the time, whether I was happy or sad. You have always been my safe haven and affectionate embrace. You sacrificed everything to see me achieve my goal.

I am proud of **myself**. Despite the loss of my father, I decided to move forward and be a better woman, one my father would have been proud of. I walked this path and persevered for myself and for those I love.

Finally, I am grateful to the **Kingdom of Saudi Arabia** for granting me the scholarship and providing everything I needed to develop my skills and achieve my dreams. The generous support I have received played an important role in the completion of my academic journey.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

Networks have often been employed to simulate the structures of various biological systems. Many strategies have been utilised to build credible biological networks. The main objective of understanding a biological system is to direct the states of complex systems towards the optimal or desired ones (Li et al., 2018). In addition, the sustainability of ecological processes that sustain life depends on biological variety (Rebolledo et al., 2019). The biological constituents of a natural system do more than engage and impact one another at the local and regional levels. They also generate intricate environmental connections responsive to outside forces (Rebolledo et al., 2019). Consequently, this thesis explores interestingness measures and interactive visualization for the usability of complex biological networks by providing background on their relevant aspects. This section will also provide basic information about the thesis, such as the problem statement, research aim, objectives, and research questions.

1.1. Background

In today's digital era, digital enterprises must manage tremendous volumes of data, commonly known as 'big data'. The exponential increase in the amount of data over recent years has led to the creation of massive and highly detailed datasets. Amidst this data deluge, the paramount challenge involves uncovering valuable insights concealed within this mass of information. Effective and automated methods are needed to identify practical trends and links in the data (Hussein et al., 2015). However, despite the availability of dependable and accurate data mining techniques, the quest for truly captivating patterns remains. Here, the concept of 'interestingness measures' introduced by Garima (2014) comes to the forefront, aiming to assist users in making decisions in extraordinary circumstances. Interestingness measures are important in data mining, as they are meant to choose and rank patterns based on their potential interest to the users (Hamilton, 2007). Good interestingness measures allow for reducing the cost of time and space of the mining process. Measuring the interestingness of identified patterns is one of the active and most significant parts of data mining research. In the context of information visualization, Behrisch et al. (2018) refer to quality metrics as a formal method for evaluating the quality and effectiveness of visualization. The metrics are used in different ways, such as the amount of clutter, the clarity of visual patterns, and how well the visualization supports the user's analytical task. Its goal is to create a standardised way for performance assessment of visualization to enable

comparison of different techniques and to support the design of enhanced visual tools. They are critical because they provide means to quantify subjective visualization elements, like usability and interpretability, more objectively. This helps in guiding the development of visualizations that are not only aesthetically pleasing but also functional and efficient for data analysis. Compared to the current study, the interestingness measure relies more on user perception and engagement, while the quality metric offers a more structured approach to evaluation. Also, interestingness measures are designed to capture how engaging a visualization is from the users' perspective. This is why surveys and interviews were conducted in this study.

In data mining, acquired data properties like consistency, understandability, and interest are essential (Garima, 2014). However, the current study underscores the need to explore and develop novel and domain-specific interestingness metrics. For instance, the associative method proposed by Jalali-Heravi and Zaïane (2010) employs association rule mining to establish relationships between characteristics and class labels. The effective ‘interestingness' metrics of support and confidence are applied to identify relevant association rules within this context. Nevertheless, scholars acknowledge the need for further investigation of interestingness measurements to narrow the pool of mined rules or discover more meaningful patterns in complex systems.

In parallel, advancements in visualization technology have enabled users to recognise critical trends in vast datasets, pinpoint areas requiring more exploration, and draw informed conclusions from the data (Zudilova-Seinstra et al., 2008). Interactive visualization has emerged as a transformative approach, fostering a closer link between analysts, their models, and the studied data. As such, it facilitates improved data understanding and fosters more effective decision-making. Such interactive visualization techniques hold promise in complex biological networks. By allowing researchers to explore and manipulate network representations, interactive visualization enhances data comprehension and empowers the identification of critical patterns and relationships.

Notably, network models have emerged as vital tools for comprehending the biological mechanisms underlying the lifespan and stability of biological systems (Allesina et al., 2015; Aljadeff et al., 2015; Arnoldi et al., 2018; Coyte et al., 2015; Grilli et al., 2017; Novak et al., 2016; Rohr et al., 2014; Su & Guo, 2016). Rich, highly linked biological networks are believed to be more stable and adaptable to environmental changes, making them valuable

sources of insights into complex biological processes (Stone, 2018). However, while the cited authors have contributed significantly to this field, the current study recognises the importance of critically analysing their specific findings and methodologies. It aims to explore potential limitations, biases, and alternatives in network modelling to advance the understanding of the intricate biological interactions driving the development of disease-related phenotypes in biological cells (Benton et al., 2021).

1.2. Problem Statement

The significant problem of this study is identifying relevant methods of measuring and visualising complex biological networks, as they are difficult to investigate experimentally (Paul & Kollmannsberger, 2020). Moreover, effective interestingness measures are needed to adequately measure the knowledge in research databases (Selvarangam & Kumumar Ramesh, 2014). Visual perception has immense potential to identify themes, trends, oddities, and clusters in data (Brodbeck et al., 2009). Visual representation converts a conceptual challenge into a more effective perceptual task (Brodbeck et al., 2009). The goal and knowledge behaviour of visual representation are critical elements of information use.

Moreover, as stated earlier, finding effective and automated methods for identifying practical trends and links in the data is crucial (Hussein et al., 2015). Numerous biologists and bioinformaticians now utilise biological network modelling and analysis frequently because these interactive graphs make it possible to map and define signalling pathways and anticipate the function of unidentified proteins (Millán, 2013). In addition, researchers in biology and medicine are working hard to understand the inherent biological process contexts of clinical illness pathways. Many biological and clinical data, including electronic health records, biomedical images, disease pathways, gene ontology, biomolecular interactions, protein and small molecule structures, DNA microarrays, and genomic sequences, have been generated (Li et al., 2014). Extracting useful information from interaction networks can be challenging, considering the magnitude and intricacy of interactome datasets (Millán, 2013).

Furthermore, data mining challenges must be conquered in order to convert the massive amount of biomedical data into meaningful information for clinical and healthcare applications. These problems involve the processing of computing-intensive tasks (e.g., mining, searching, and large-scale graph indexing) as well as the managing of noisy and incomplete data. These issues present new challenges for data mining researchers in the data-

intensive post-genomic era (Li et al., 2014). Building, characterising, and analysing biological networks can be done using various methods and technologies. Hence, this research seeks to explore the possibility of employing interestingness measures and interactive visualization for the usability of complex biological networks.

1.3. Research Aim, Objectives, and Questions

The key research question of this study is as follows: '*What technique can be used to develop a more suitable visualization tool that can overcome human perceptual and cognitive limitations?*' The overarching aim of this project is to research novel methods for developing more usable visualization tools for biological networks, considering the perceptual and cognitive limitations of humans and utilising the concepts of interestingness measures and interactive visualization. The objectives and the associated research questions are provided in Table 1.

Table 1: The objectives and the associated research questions

Objectives	Sub research questions
Objective 1: To determine the tasks and patterns of interest concerning analysing biological networks through a literature review and interviews/consultations with biologists.	• **Research Question 1.1:** What methods, tasks, and patterns can enhance the usability of complex biological networks? • **Research Question 1.2:** What are the network visualization tools and techniques used for biological data?
Objective 2: To define a set of interestingness metrics based on tasks and patterns.	• **Research Question 2.1:** In what ways can the concepts of interestingness and visualization support complex biological networks? • **Research Question 2.2:** What are the different application patterns of the visualization techniques for complex networks?
Objective 3: To evaluate the usability and limitations of existing network visualization methods used for biological networks.	• **Research Question 3.1:** How can the usability of complex biological networks be improved?

	• **Research Question 3.2:** What interestingness measures can be used to enhance the analysis of biological processes?
Objective 4: To design and develop a visualization tool that overcomes current limitations and supports human cognition and data exploration, using multiple coordinated views and interactivity guided by interestingness metrics.	• **Research Question 4.1:** What are the drawbacks of the selected visualization tool? • **Research Question 4.2:** How can these drawbacks be overcome?
Objective 5: To evaluate the designed network visualization tool and upgrade it based on the testing.	• **Research Question 5.1:** What is the usability of the implemented tool? • **Research Question 5.2:** What is the user acceptance rate of the implemented tool?

1.4. Significance of the Study

The current study aims to develop more usable visualization tools for biological networks, considering humans' perceptual and cognitive limitations and utilizing the concepts of interestingness measures and interactive visualization. This will have several benefits. First, the study will provide useful insights regarding the usability of complex biological networks, the different patterns of visualization used in visualizing data, and the relevant interestingness measures for improving biological data. In addition, it will add to the existing knowledge of interestingness measures, interactive visualization, and complex biological networks. Regarding practice, the information that emerges from this study will help give users more control over the entity being investigated, which will help them make good decisions when investigating biological systems.

1.5. Contribution

The main contribution of this study is providing a holistic approach to enhancing the usability of complex biological networks. The study integrates insights from the literature, experts' opinions, and innovative visualization techniques to enhance the researcher's analytical capabilities in the field of biological networks. The first objective of the study is to determine the tasks and patterns of interest in analysing biological networks through a literature review and interviews/consultations with biologists. In this way, the study can

improve the current understanding of the tasks and patterns of interest when analysing biological networks. This information synthesis, including the literature review findings and the insights obtained from the interviews and survey consultations with biologists, will be critical for redirecting subsequent objectives. Objective two involves defining a set of interestingness metrics based on tasks and patterns. The study contributes to developing a set of interestingness metrics for identifying tasks and patterns in biological network analysis. These metrics will serve as a foundation for evaluating the importance and relevance of information within the networks.

The third study objective is to evaluate the usability and limitations of existing network visualization methods used for biological networks. The study assessed the advantages and weaknesses of the current visualization tools used in biological research. This evaluation offers a critical analysis of the current visualization tools, helping to identify areas where improvements and innovations are needed. Similarly, the fourth objective involves designing and developing a visualization tool that overcomes current limitations and supports human cognition and data exploration, using multiple coordinated views and interactivity guided by interestingness metrics. This advanced visualization tool is specifically designed for biological network analysis. Finally, the fifth objective involves evaluating the new network visualization tool and upgrading it based on testing. The insights gained from the evaluation process will guide further improvements and upgrades to ensure the effectiveness of the tools in supporting human cognition and data exploration in complex biological network analyses.

1.6. Definition of Key Terms

A list of the key terms used in this thesis is presented below.

1.6.1. Interestingness measures

Interestingness refers to attracting and holding one's interest or attention (McIntyre et al., 2021). Measures in data mining are used to select and rank patterns according to their usefulness to the user (Selvarangam & Kumar, 2014; Sharma, 2022;). Interestingness is an important aspect of data mining, which refers to the attractiveness of an association, data collection, filtering, and arrangement in large databases and understanding the knowledge discovery process in rule mining. In the context of the research presented in this thesis, interestingness measures refer to patterns or features of interest when analysing complex biological networks.

1.6.2. Interactive visualization

The concept of interactivity refers to the process through which data are transferred from one segment of a system to another, that is, the interaction that involves sending a search query in the search engine of a system to the production of results as output (Brodbeck et al., 2009). Interactive visualization combines interactivity with visualization to provide the user with the best experiences when using interactive visualization systems. According to Luo (2019), interactive data visualization allows users to directly control the data and the information presented to help them make effective decisions.

1.6.3. Complex biological networks

Biology networks refer to complex sets of dual interactions with different biological systems. These biological functions rely on the architecture of the network, which emerges as the result of a changing feedback process (Salem, 2018). Network biology involves investigating complex biological systems to understand biological functions better through the representation of the binary association among different biological systems. For instance, the human immune system is a network of cells, tissues and organs working in consonance to protect the body from infection. Contrastingly, a non-complex biological network example is the digestive system, which is a network of organs working together to break down food and absorb nutrients (Milenković et al., 2011).

1.7. Thesis Structure

Chapter 1 is the introduction of the thesis, providing the study background and the problem statement. Furthermore, it presents the aim and objectives of the study and discusses its significance. Chapter 2 focuses on visualization tools for complex biological networks, examining and comparing different tools, and reviews related works in the literature. Chapter 3 discusses the factors impacting visualization, which are divided into general evaluation factors and heuristic factors. Chapter 4 deals with the evaluation of the factors. Here, the adapted method, which comprises data collection and data analysis methods, is examined. Interview findings and survey analysis are examined separately, and a comparative analysis between general and heuristic factors is presented. Chapter 5 evaluates visualization tools, including the datasets, layouts, and five specific visualization tools. Chapter 6 introduces an enhanced visualization tool. Background information about visualization techniques is discussed, with a particular focus on the fisheye view, its technology and architecture, design and implementation and updates to the Blurfisheye

visualization tool. Finally, Chapter 7 presents the overall conclusions of the study along with key limitations and suggestions for future works.

1.8. Publications

1- Chapter Three was presented and published as a peer-reviewed poster under the title '***Investigation and identification of essential factors for visualization tools for complex biological networks***' at the ISMB-ISCB (Intelligent Systems for Molecular Biology – International Society for Computational Biology) conference in 2022.

2- Chapters Two, Three, and Four were extended and published as a paper under the title '***An investigation into various visualization tools for complex biological networks***' in the Sage journal *Information Visualization* in 2023.

3- Chapter Six is ready for submission to *Frontiers in Bioinformatics*.

Chapter 2. Visualization Tools for Complex Biological Networks

The advancement of technology and big data generated from various systems has enhanced the study of complex biological systems (He & Wang, 2020). This abundance of biological information has resulted in a complicated network of interactions between genes, proteins, metabolites, and other macromolecules, producing biological networks. Understanding the structure and dynamics of these networks is critical for unravelling the intricacies of life processes, illness aetiology, and therapeutic intervention possibilities. Biological networks are highly complex, and thus, suitable tools are needed to visualise and provide meaning to the data. Therefore, this study section examines the different visualization tools and methods available for biological networks, including Cytoscape, Ondex, Metashark, and ProViz. A detailed discussion of the tools commonly used for visualising biological networks is provided in this chapter.

2.1. Visualization

Previously, visualization was defined as constructing a visual image in the mind (White et al., 1998). Recently, however, it has come to mean something more than a mental visual image; it is more like a graphical representation of data or concepts. Visualization has become an external artefact that supports decision-making. According to Ware (2013), four basic visualization stages are combined in several feedback loops. The first involves the collection and storage of data. This is followed by a pre-processing stage, where the data are changed into something that can be easily manipulated. The third stage deals with mapping the chosen data to a visual representation, which is achieved through computer algorithms that project images to the screen. Finally, the process is completed with the human perceptual and cognitive system stage.

Visualization can reveal unanticipated emergent properties. It helps in understanding large volumes of data and makes problems with data become more obvious, as it shows not only aspects of the data but also how they have been collected. Despite these obvious advantages, there are some inherent challenges, including data complexity. Visualising a large volume of datasets may be challenging, as careful data selection and transformation are needed to ensure there is clarity without overwhelming the viewers (Petropoulos et al., 2022). Another challenge is visual clutter. Overloading a visualization with too many elements or unnecessary details may make it difficult for viewers to extract necessary insights (Maya, 2023). There is also a challenge related to colour and perception. Colour plays an important

role in data visualization, and cultural differences and individual perceptions can affect how colours are interpreted (Grzybowski and Kupidura-Majewski, 2019).

According to Sedlmair et al. (2012), learning, winnowing, casting, discovering, designing, implementing, deploying, reflecting, and writing are nine stages of the design methodological framework for visualization. These are further classified into three top-level categories. The precondition phase, also known as personal validation, comprises learning, winnowing, and casting. Winnowing filters, the ideas and requirements obtained during the 'learn' stage. It encompasses prioritising the most relevant options that align with the project's objective. Essentially, winnowing refines the scope of the visualization project, focusing on the most crucial element (Dush, 2021). Collaborators who can contribute to the winnowing process include stakeholders with deep domain knowledge, data scientists, and project managers. On the other hand, cast is where refined ideas after winnowing are structured and defined to guide the subsequent stages of the design process. It highlights the design specification and technical requirement determinants and sets the stage for the main project phase (Gib and Brodie, 2005). Collaboration should involve technical leads, UX/UI designers, and software developers. The second category is core, which is also known as inward-facing validation, which is comprised of discovering, designing, implementing, and deploying. It focuses on the internal development process to ensure visualization meets the technical, functional, and aesthetic requirements established in the previous stages (Hashemi-Pour et al., 2022). At this stage, collaborating with domain experts is crucial to getting feedback on the visualization's accuracy, relevance, and usability. The third category is analysis, also known as outward-facing validation, which comprises reflection and writing.

Human perception and visualization are the processes through which humans acquire, interpret, and represent information from their surroundings (Ware, 2019). This involves using sensory organs, such as the touch receptors, ears, and eyes, to receive stimuli from the environment, which are then transmitted to the brain (DeSalle, 2018). Then, meaningful patterns and structures are drawn from the complex visual stimuli. Visualization plays a significant role in human cognition by allowing individuals to visually represent and manipulate abstractive concepts and data, enhancing understanding, decision-making, and communication.

2.1.1. The purposes of visualization and interactive network exploration

Visualization and interactive network exploration are vital in complex biological network analysis and understanding. They enable researchers to intuitively understand complex data, identify significant trends and generate hypotheses to be investigated (Nguyen et al., 2024). Part of the purposes of these tools are:

Data comprehension and recognition of trend: Visualization can change abstract information to a more intuitive format, which then assists the researchers in quick understanding of the structure and relationships inherent in the network (Franconeri et al., 2021). A visual representation can show the trends, clusters and weaknesses that may not be clear enough from the raw data.

Hypothesis generation and testing: An interactive tool ensures that researchers can dynamically explore the network. This helps generate novel hypotheses based on pattern observation and interactions (Jing et al., 2022). Furthermore, visualization helps in biological hypothesis testing by explaining how a change in one part of the network could alter the whole system (Biesecker, 2013).

Identification of vital components and interactions: Visualization seamlessly identifies vital nodes with many connections. This is crucial for maintaining the network's integrity and function. Research can, therefore, show the important interactions and pathways that are significant to certain biological processes (Pan et al., 2016).

Functional and structural insights: Visualization identifies functional modules and communities in any network and helps to evaluate the biological processes' organization and modularity more effectively (Alcala-Corona, 2021). Understanding the structural properties of the network, like clustering coefficients and centrality measures, is more manageable and straightforward when represented graphically (Dudzic-Gyurkovich, 2023).

Communication and collaboration: Visualization provides a more precise and effective way to communicate complex data and findings to others from different fields. The interactive tool allows for broader collaborative exploration by enabling researchers to investigate various network parts simultaneously (Isenberg et al., 2011).

2.2. Complex Biological Network

Nature presents humanity with a large mass of structures whose functions are diverse. Structural-functional relationships are highly specialised and are exclusive to one or many specific domains of biological science (Bogdan et al., 2022). These structures can be found in genes, development, neural circuits, and integration. They can also be found in metabolic pathways and trophic interactions. An underlying organisation suggests a universal principle of interactive connectivity across its components in terms of the overall structure and dynamics of the biological domain (Bogdan et al., 2022).

Biological systems can be grouped into components (parts) combined with other elements to become one (Simon 1962). When the parts interact with another part of the system, time, space, information and function are constrained. These are influenced by the external environment. In biological networks, interactions are modelled with graphs and mathematical constructs that join from one point to another, known as lines of vertices (Barabasi and Oltavi, 2004). A graph that describes a combination of structural domains in multidomain proteins will join the vertices and describe structural domains with lines depicting the domain presence in proteins (Aziz and Caetano-Anolles, 2021). When the connection of vertices is not directed, lines will fail to point in any direction, and each connection involves an unordered pair of vertices. Therefore, these lines are called edges. However, when connections are directed, lines will point in a single direction; when each connection has an ordered pair of vertices, the lines are called arcs. Graphs then become networks when value functions such as properties or weights are mapped into vertices of the network nodes, and the lines connecting the vertices link the networks.

Network abstractions in biology are difficult to understand, as is to be expected of complex systems. The network can become structurally complex when its wiring tangles as a result of multiple rules governing network responses to environmental perturbations. In terms of connectivity, the links between nodes differ due to weights, directions, and signs, and they interact in other ways. In terms of diversity, nodes and links can be diverse. For example, the substrates and enzymes in biochemical networks that control cell division differ. In terms of evolution, networks' structure and dynamics can change as they grow, and their wiring diagram expands over time. In terms of dynamics, nodes and links can exhibit nonlinear, long-range memory, or multifractal dynamic behaviours. The state of each node or link could differ in time and be complicated to ensure the achievement of collective goals in a decentralised way (Bogdan et al., 2022).

The complex, diverse, and ever-changing network accurately describes how components in natural systems interact. The correct definition of a biological part is critical to the network modelling process. For example, structural domains are viewed as protein structure units that aid in protein taxonomy classification (Cateno-Anolles et al., 2009). Domains are secondary structure elements that have been folded into well-packed and compacted polypeptide structural units. They are functional models because they fold and function independently, helping to maintain protein stability by establishing various intramolecular interactions and hosting specific molecular functions.

Proper understanding of biological processes is a function of knowledge about biological entities and their interrelationships. Cell differentiation, for instance, depends on which protein is available and which is bound together. A graph, also known as a network, is one natural way of representing the process. This is because graphs model entities and the way they interact. The recent advancements in experimental high-throughput technologies have greatly enhanced the data output from monitors at a reduced cost, leading to a vast number of biological network data (Reuter et al., 2015). The presence of such data makes it easier to address bioinformatics challenges. Such challenges include predicting new drug interactions with biological pathways and the structural anticipation of new protein functions.

Chen et al. (2010) defined complex biological networks as a group of interconnected components within living organisms. These components include cells, molecules, and tissues, and networks are characterised by the dynamic relationships and interactions between them. The key features of complex biological networks discussed in the literature are interconnected components, the hierarchy and organisation of the components, the dynamic interaction between components, and continuous information flow. In a biological network, nodes represent discrete molecular entities such as genes, RNA, transcripts, proteins, metabolites, enzymes and regulatory factors. Meanwhile, edges represent interaction types like activation, inhibition, expression, or catalysis, with a weight that is proportional to the strength or statistical significance of the interactions (Dandekar et al., 2010).

Typical features and the kind of patterns that domain experts can extract from complex biological networks.

Domain experts can extract typical features and patterns from biological networks to get insights into biological systems' dynamics, functions and structure. Some of these key features and patterns are:

Nodes and edges: As stated previously, nodes are biological entities like genes, proteins, metabolites, or species, while edges are interactions or relationships between nodes (Muzio et al., 2021).

Degree distribution: This is the number of node connections. It usually follows a power law showing a few highly connected nodes and many with few connections (Kong et al., 2019).

Network motifs: These are small recurring sub-networks or patterns of interconnections. The common ones in biological networks are feed-forward loops, bi-fans, and feedback loops (Lecca et al., 2016).

Modules and communities: These are groups of tightly interconnected nodes corresponding to functional pathways. Module detection can help domain experts identify gene clusters or proteins that work hand-in-hand in a specific biological process (Sia et al., 2022). On the other hand, communities are larger node groups with dense internal connections and sparse connections to different groups, showing a higher level of organisation in a particular biological network (Mao et al., 2017).

Pathways: These are specific interaction sequences that can lead from one node to another, like signalling pathways, metabolic pathways or gene regulatory pathways (Hue et al., 2016). They help in understanding the information flow in a particular network.

Centrality measures: These are categorised into three: betweenness centrality, closeness centrality, and Eigenvector centrality. Betweenness centrality measures the extent to which a node lies on the shortest paths between other nodes, showing its role as a potential control point in a network (Peng et al., 2018). Closeness centrality shows the level of closeness of a node to all other nodes in the network, which can determine its efficiency at propagating signals (Peng et al., 2018). Then, Eigenvector centrality shows a node's impact based on its neighbours' significance.

Clustering coefficient: This measures the likelihood of nodes' neighbours being connected to each other. A high clustering coefficient means a high degree of local interconnectedness (Smith-Miles and Lopes, 2012).

2.3. Visualization Tools

Several visualization tools have been introduced in the literature, and 13 key tools were selected for this study. This section presents an in-depth review of the following visualization tools: Cytoscape, Osprey, Medusa, ProViz, CN-Plot, Ondex, MAPMAN, Pajek, MetaSHARK, BioLayout Express3D, Arena3D, CellNetVis, and Gephi. These tools were deemed relevant to this study because they facilitate and enhance the discovery of complex data interpretation patterns and can provide useful insights across various scientific and research domains.

2.3.1. Cytoscape

The broad concept of Cytoscape as an open-source software programme for biological network visualization, integration, and manipulation was expanded by Shannon et al. (2019). According to the authors, Cytoscape is a multi-purpose programme that integrates large-scale biomolecular networks, high-throughput expression data, and other forms of molecular states in a common conceptual framework. It works on all major operating systems, with additional features like plugins, and it is freely available for download (Shannon et al., 2003). The major feature of Cytoscape is the basic functionality it provides for the integration of arbitrary data with a visual representation on a graph, including the integrated data, an interface, and filtering tools (Shannon et al., 2019). It also serves as a useful tool for visualising and analysing network graphs. The data integration uses 'attributes', which serve as pairs in mapping nodes or names to specific data values. According to Shannon et al. (2019), the Cytoscape software programme can convert expression data into node labels, colours, border colours, or thicknesses based on the user configuration and visualization schemes. Although Cytoscape can be applied to any system with molecular interactions and components, it is more effective when combined with larger databases, including protein–DNA, protein-protein, and genetic interactions (Shannon et al., 2019). Further, the programme mainly focuses on high-level representations of interactions and components. One of the significant strengths of Cytoscape is its flexible display and series of layouts, which represent different biological relationships (Bell & Lewitter, 2006).

2.3.2. Osprey

Osprey is another important open-source software programme used to visualise and manipulate complex biological networks. According to Breitkreutz et al. (2003), the Osprey network visualization system represents interactions in a flexible and easily expandable graphical format. It offers various options for making practical comparisons among datasets.

They further noted that the tool represents genes as nodes and interactions in the form of edges between nodes. It is a Java-based programme that runs on Linux and different desktop operating systems and can be downloaded for free by academic scientists following registration on the site (Bell & Lewitter, 2006). Osprey can be used in a stand-alone form and as a viewer for online interaction databases. Compared to other visualization tools, it can be fully customised, enabling users to personalise their settings to generate interaction networks (Breitkreutz et al., 2003). This is because any interaction dataset can be loaded into it with one of several standard file formats or through underlying interaction database upload. Furthermore, Osprey uses the General Repository for Interaction Datasets (GRID) for its database, through which a user can easily build interaction networks. Breitkreutz et al. (2003) emphasised that compared to previously existing network visualization systems, it was difficult to search the network for individual genes in the form of large graphs, and there was no functional information in the graphical interface. However, Osprey offers a solution to these problems by providing a one-click link to the different database field nodes with a description of each of their functions and also enables users to conduct text search queries with the use of gene names (Breitkreutz et al., 2003).

2.3.3. Medusa

Medusa is an important Java application that is used to visualise and manipulate interaction networks. The programme was developed to complement and address some of the drawbacks identified with existing network visualization systems. According to Hooper and Bork (2005), Medusa is a network visualization tool that is designed to be simple and straightforward, with a display of up to 10 multiple edges that run simultaneously between nodes. They further noted that, like other applications like Cytoscape and Patek, Medusa enables users to easily add and delete edges and nodes by simply clicking the mouse, and the programme also includes background images that can be used to improve the quality of every design. Moreover, Medusa can be used to describe node properties, such as shape, colour, position, and annotation. It does not require the use of additional packages and can run on different machines if they are Java-enabled (version 1.4.2) (Hooper & Bork, 2005). Additionally, Hooper and Bork (2005) noted that Medusa is designed to be easily accessible and mainly used as a graph visualization tool for constructing figures. It can run as a stand-alone programme and is available as an applet version on web pages or interfaces, and the graphs can be exported as postscripts or in image formats (Hooper & Bork, 2005).

2.3.4. ProViz

ProViz is a powerful visualization tool developed by the Int-Act European project to visualise protein-protein interaction (PPI) networks. According to Iragne et al. (2005), a combination of algorithmic and visualization tools that are well integrated into the software and have the ability to access distant and local data banks is needed to analyse PPI networks. Hence, ProViz was developed and integrated with the Int-Act data model, which enables the interactive visualization of large interaction networks (Iragne et al., 2005). As Iragne et al. (2005) noted, the programme was designed to improve existing network visualization systems by providing a scalable, fast, and open-source tool with many plugins that can integrate developing standards to display relevant knowledge within a biologist-oriented interface. The application was also designed to interactively understand the process through which biologists work, explore large graphs, and recognise proteins and other interactions through a keyword search or analysis of network structures (Iragne et al., 2005). ProViz can manipulate graphs with several elements and can also be used to make comparisons between graphs of various species as well as to extract views and subgraphs for analysis. It also enables the clustering of similar proteins as well as interactions (Iragne et al., 2005).

2.3.5. CN-Plot

According to Batada (2004), CN-Plot is a simple open-source tool for visualising global connectivity in pre-clustered network data or graphs, such as clustering genes or proteins based on expression patterns, biological functionality, or geometrical structures. CN-Plot can be easy to implement and provides well-informed summaries of data and interpretable layouts. A Java implementation of the software programme is freely accessed on the website, and the application can run on any platform with a Java-enabled virtual machine (Batada, 2004). Using Graph-Viz, the standard graph layout can only produce cluttered visualizations that can be difficult to interpret. With CN-Plot, the same data set can be produced clearly and summarised (Batada, 2004). As further noted by Batada (2004), the programme's Graphical User Interface (GUI) enables users to specify several graphical parameters to produce a LaTeX output file that can be easily edited using picture editors like JPicEdit. A combination of LaTeX and JPicEdit streamlines the process of creating high-quality, formatted documents with complex graphical elements. LaTeX handles the text and mathematical content, whereas JPicEdit allows for easy creation and editing of graphical components. This combination is useful for academic and technical writers who need to produce publication-ready documents. Batada (2004) also emphasised that CN-Plot can be

used as a simple open-access tool to analyse large-scale PPI data and genetic interactions. The tool can be used to discover relevant information biologically related to the organisation of the network as well as to generate hypotheses concerning the possible connections and interactions between the different clusters (Batada, 2004).

2.3.6. Ondex

According to Kohler et al. (2006), ONDEX is a database system that can be used to interpret gene expression results. The programme combines two features: semantic integration of databases and text mining with graph-based analysis methods. Kohler et al. (2006) highlighted the effectiveness of ONDEX as a visualization tool for identifying the causal relationships between stress response genes and the metabolic pathways within gene expression data. Moreover, the application can be freely accessed with a General Public License (GPL) and downloaded on the website. The ONDEX database system is important for other visualization tools, as it combines text mining, sequence and graph analysis, and large-scale database integration (Kohler et al., 2006). The combination of these important methods enables the system to acquire relevant knowledge that cannot be derived from using some of the methods alone. Further, Kohler et al. (2006) demonstrated that the ONDEX database system can be used to analyse and interpret experimental results. It is used for storing, querying and visualising biological networks, which makes it a valuable resource for researchers in the life sciences. It can be used to analyse data derived from any organism in a way that is tailored to the unique biological features of such species or organisms, and pathway information derived from other species and model organisms, such as a mouse, can be integrated and further exploited. However, care is needed when making conclusions across different species (Kohler et al., 2006). This is because the ONDEX system allows the integration of different pathway information. As such, there can be significant biological differences between organisms, even closely related organisms.

2.3.7. MAPMAN

MAPMAN is a user-friendly tool that visualises huge genomic datasets as metabolic pathways alongside various biological procedures (Thimm et al., 2004). MAPMAN is made up of hunter modules that gather and categorise the parameters measured into hierarchical functional groups (Thimm et al., 2004). The modules are significant algorithms programmed to hunt for certain information types within the dataset and assign them to the relevant and functional groups. This enables the display of large-scale datasets in pictorial diagrams, which serve as a symbolic representation of the different biological function areas.

According to Thimm et al. (2004), with the use of hierarchical categories as well as diagrams with additional details, several functional areas can be analysed at various levels. The application can be downloaded from the website along with the image annotator module and further relevant instructions (Thimm et al., 2004).

2.3.8. Pajek

Given that many network algorithms are time and space-consuming and thus not suitable for large network analysis, other approaches have been explored for the analysis and visualization of large networks (Batagelj & Mrvar, 1998). Pajek is another open-source programme developed to analyse and visualise such networks. According to Batagelj & Mrvar (1998), the main objectives of the Pajek visualization programme are to provide users with a powerful visualization tool, implement a set of effective algorithms that can be used for analysing large networks, and provide support for the abstraction of large networks using factorisation and dividing them into several smaller networks, which can be further treated with the use of more refined methods. The programme runs on Windows (32-bit) and can be accessed freely for academic use on its website (Batagelj & Mrvar, 1998). As noted by Batagelj and Mrvar (1998), Pajek utilises six data structures in its algorithms, which include vector, cluster, partition, hierarchy, network, and permutation, and besides its input format, the programme also supports several other input and molecular formats.

2.3.9. MetaSHARK

The Metabolic Search and Reconstruction Kit, referred to as MetaSHARK, is an automated software package designed to detect enzyme-encoding genes in genome data that are not annotated in their visualizations within the context of closely related metabolic networks (Pinney et al., 2005). It has a gene detection package known as SHARKhunt, which runs on the Linux system and requires only a few sets of raw DNA sequences as input. According to Pinney et al. (2005), compared to other existing enzyme annotation programmes, which begin by predicting proteins from annotated genomes, using text mining and sequence analysis methods to construct a list of enzymatic functions, SHARKhunt only requires a set of DNA sequences, such as contigs, expressed sequence tags (ESTs), and finished chromosomes and genome survey sequences, to serve as input. This is used to extract new knowledge regarding metabolic capabilities from the initial data derived from genome sequences and unannotated genomes (Pinney et al., 2005). SHARKhunt software and other necessary programmes can be freely accessed on the metaSHARK website.

2.3.10. BioLayout Express3D

BioLayout Express3D is a new open-source visualization tool used to analyse gene expression data and for 3D visualization and analysis of complex biological networks derived from microarray data (Freeman et al., 2007). According to Freeman et al. (2007), these networks comprise nodes in the form of transcripts connected based on the similarity of their expression profiles across various conditions. They further noted that this JAVA visualization tool is fast, versatile, and intuitive and enables the identification of biological relationships that could have been missed with the use of conventional analysis techniques. This application is freely available for download and accessed. The application lets users mine their data and visualise the results in 3D.

2.3.11. Arena3D

Arena3D is a user-friendly visualization tool designed to comprehensively visualise biological and other networks within a 3D space (Pavlopoulos et al., 2008). Pavlopoulos et al. (2008) noted that complexity is a major issue when biological networks are being visualised. This is because the graphical representations become increasingly incomprehensible as the number of entities continues to increase. Hence, there is a need for software to visualise several entities without losing their meanings. Arena3D combines new 3D layers with several layouts, algorithms, data-filtering tools, and other relevant tools. This enables users to use large datasets and divide them into simpler 2D graphs, making the datasets easy to understand, unlike compiling all the data into a 2D graph or a 3D space, which makes the data difficult to comprehend (Pavlopoulos et al., 2008). According to Pavlopoulos et al. (2008), by separating the datasets in 3D, considerable space is created for the vertices within the layers, which helps to avoid possible intersections as well as overlaps that frequently occur in 2D, facilitating the visualization of larger datasets. Moreover, they noted that compared to other visualization tools, Arena3D goes further by integrating various analysis methods alongside visualization to make the exploration and discovery of hidden relationships in more complex networks and large-scale datasets easier. This visualization tool can run on any platform and is free for academic use.

2.3.12. CellNetVis

CellNetVis is another powerful web tool developed for visualising biological networks through force-directed layouts that are limited by cellular components (Heberle et al., 2017). It is also an open-source and freely accessible tool and can be used alongside networks derived from similar databases (Heberle et al., 2017). As Heberle et al. (2017) stated, this

visualization tool is designed to display biological networks in the form of a cell diagram using a constrained layout algorithm. They further noted that even though several visualization tools are used to explore and visualise network models, none of these tools are specifically designed to divide the network models into cell structures like CellNetVis. Furthermore, the tool can be utilised to investigate complex biological networks by generating a reliable representation of the cell diagram on the web (Heberle et al., 2017). They further noted that the web tool is useful for displaying complex network information, edges, and nodes and their relations with portioned cells. CellNetVis is best suited for use with small and medium-sized networks.

2.3.13. Gephi

Gephi is another commonly used open-source software programme for network and graph exploration, manipulation, and analysis (Bastian et al., 2009). The programme employs a 3D render engine to present large networks and graphs in real-time and accelerate the exploration process. It also uses highly configurable and special force-directed layout algorithms, such as the ForceAtlas algorithm, for network visualization and analysis. The ForceAtlas layout algorithm is built with real-life settings that include auto stabilise functions, gravity, speed, and size, among others. According to Bastian et al. (2009), Gephi can visualise large complex networks with over 20,000 nodes and generate relevant visual results, as the programme is built on a multi-task architecture and uses multi-core processors. Further, the programme can personalise node designs into shapes like photos, panels, or textures. On Gephi, the user interface is divided into workplaces. Therefore, a wide range of algorithms can be run simultaneously in different workplaces without hindering the functioning of the user interface. The software also provides wide and easy access to network data and enables network filtering, clustering, spatialising, navigating, and manipulation. Gephi is built to perform extensive functions, including the easy addition of filters to programs, without requiring prior programming experience on the part of the user. Moreover, edges or nodes can be easily obtained and selected manually, and filtering tools are available.

2.4. Comparison of Visualization Tools

Table 2 presents a comparison of different visualization tools that may be used for visualising biological networks.

Table 2: Comparison of visualization tools

Tool	Open Source	File Formats	Layouts	Scalability	Editing	System	URL
Cytoscape	Yes	Supports different input formats, such as GML, SIF, NNF, BioPAX, and PSI-Mi.	A wide variety of simple grids and sophisticated algorithms are available.	Can visualise large networks with nodes and edges. Cannot effectively scale analysis.	It offers predefined visual styles and colour schemes, 17 academic viewers, etc.	Stand-alone	http://www.cytoscape.org/
Osprey	Yes	Supports raw and processed formats from major MRI vendors like GE, Siemens, and Philips.	Circular, concentric, spoke, and dual-ring layouts.	It can use two or more datasets in an additive manner and has filtering options.	Automated identification of input file formats; uses GRID to build interaction networks.	Stand-alone	http://tinyurl.com/osprey1/
Medusa	Freely accessible for academic use.	Data from the STRING database.	Displays 10 multiple edges between nodes using a Bezier curve.	It can be used for graph analysis and construction of figures.	Contains background images that can be used to improve design quality.	Stand-alone	http://coot.embl.de/medusa/
ProViz	Freely accessible with GPL license.	GO and PSI-Mi formats.	GEM, hierarchical and circular layouts.	Provides facilities for navigating in large graphs.	Provides screen updates and content-type helper for interaction database.	Web	http://tinyurl.com/proviz/

Tool	Open Source	File Formats	Layouts	Scalability	Editing	System	URL
MAPMAN	Yes	Map files.	Scavenger modules, image annotator.	Displays genomic datasets.	Display of genes in metabolism.	Stand-alone	
CNplot	Yes	Clustered graphs.	Area minimisation and symmetry, edge crossing, force-directed energy layout.	Can analyse large-scale genetic interactions and protein-protein interaction data.	Free picture editor	Web	https://www.cambiumnetworks.com/cnpilot-free-ap/
Pajek	Freely accessible for academic use.	It only supports networks in Pajek (.net) (Strict file input formats).	Graph layout, neighbourhood detection, clique finding, node merging, etc.	Can visualise a million nodes with over a billion connections.	Manual graph editing	Standal one	http://vlado.fmf.uni-lj.si/pub/networks/pajek/
ONDEX	Freely accessible with a GNU public license.	OBO ontologies.	FastCirculae, Ycircle, JungFR Layouts.	Text mining, graph analysis	Graph filters	Web	http://www.ondex.org/
MetaSHARK	Yes	Raw DNA sequences	SHARKview	Analysis of preliminary genome sequence data.	Colour schemes and coding	Web	http://bioinformatics.leeds.ac.uk/shark/
BioLayout Express3D	Freely accessible with a GNU public license.	GML and Text files containing expression data in tabular formats.	Fruchterman–Rheingold layout algorithm.	Large-scale studies of gene functions.	JfreeChart for customisation of plots.	Web	http://www.biolayout.org/

Tool	Open Source	File Formats	Layouts	Scalability	Editing	System	URL
Arena3D	Free for academic use.	Supports text file formats.	Inter/intra-layer layouts: circle, grid, random, star, Fruchterman–Rheingold layout algorithms, etc.	Can visualise large-scale complex biological networks.	Control buttons for easy 3D navigation: zooming, panning and orbiting.	Web	http://bib.fleming.gr:3838/Arena3D
CellNetVis	Freely accessible with a GPLv.3 license.	XGMML formats	Force-directed layout algorithm.	Can visualise both small and large networks.	Search for nodes by label, manual creation of shapes and position nodes.	Web	http://www.lge.ibi.unicamp.br/cellnetvis
Gephi	Yes	Supports CSV, GEXF, GDF, GML, GraphML, Pajek NET, GraphViz DOT, UCINET DL, Tulip TPL, Netdraw VNA, and spreadsheet formats.	Force-directed layout algorithm (ForceAtlas).	Can visualise large networks (over 20,000 nodes).	Graphical modules, designs of nodes, edges and labels. Options for increasing network clarity and readability.	Web	http://gephi.org/

2.5. Related Works

Studies related to the current research have been published on interactive data visualization tools for biological networks (Ali et al., 2016; Caldarola & Rinaldi, 2017; Faysal & Arifuzzaman, 2018; Pavlopoulos et al., 2008, 2017; Suderman & Hallett, 2007). For example, Suderman and Hallett (2007) investigated the tools used to visually explore biological networks. They looked at the benefits and drawbacks of existing systems to help researchers identify suitable data visualization tools and set achievable objectives for the next generation of visualization tools. The authors employed a systematic review and comparison method. Based on their analysis, as the cost, accuracy, and efficiency of these tools improve, it will be increasingly easier to visualise large datasets, and the varieties of data types will also increase. They also identified different tools that can be used for data visualization, such as Cytoscape, Pathway Studio, PATIKA, VisAnt, and ProViz, and evaluated each of them based on their unique features (Suderman & Hallett, 2007).

The study of Pavlopoulos et al. (2008) is also related to the current study, as it identified visualization tools that can be used in biological network analysis. The authors examined the tools' functionalities, strengths and limitations and how they could be improved for data integration and information sharing. It was found that each visualization tool has specific features that vary in how it deals with the identified challenges. While several visualization tools were analysed, the authors recommended the use of Ondex, Pivot, and Medusa as integrative tools for solving the issue of data heterogeneity, Cytoscape and BioLayout Express 3D to deal with sheer mass issues, tools like Medusa for systems biology data, Pajek for system recognition, and Osprey for the visualization of biological functions with a comparative focus.

Ali et al. (2016) focused on big data visualization, tools, and challenges. The study's main aim was to examine the importance of big data visualization and the challenges associated with the process and methods. The authors employed a qualitative method for the research and a comparison matrix to review the strengths and limitations of different popular visualization tools (Ali et al., 2016). The research found that the visualization of big data is necessary in today's world, which involves the use of digital technologies and the daily use of data. Hence, considering the large volumes of data, traditional visualization methods are no longer sufficient to keep pace, and more advanced tools are needed to cater to the different characteristics of big data that offer better response times and performance and enhance

interactive visualization. They identified five popular and useful visualization tools, including Tableau, Gephi, Microsoft Power BI, Plotly, and Excel 2016, and suggested using them based on requirements. Ali et al. (2016) stated these tools are promising and effective for generating rich and interactive visualizations.

In addition, Caldarola and Rinaldi (2017) surveyed big data visualization tools, the new paradigms, and the methodologies and tools used to visualise large datasets. Their study aimed to analyse the most commonly used visualization tools and techniques for large datasets and identify their functional and non-functional characteristics for the benefit of entrepreneurs and researchers (Caldarola & Rinaldi, 2017). The findings highlighted the flexibility of most of the tools, which are multi-platforms and built with APIs that facilitate easy access. Some of the tools they evaluated for visual analytics and visualization of big data were Gephi, Plotly, Cytoscape, Pajek, Tulip, and SocNetV (Caldarola & Rinaldi, 2017).

In addition, Pavlopoulos et al. (2017) empirically compared visualization tools used for the analysis of large-scale networks. According to the authors, while there are several existing tools for the manipulation, visualization, and interactive exploration of such networks, only a few can scale up in accordance with modern information growth. Hence, they set out to identify some of the available network visualization tools that are suitable for the visualization, analysis and exploration of large-scale networks based on the point of view of a user (Pavlopoulos et al., 2017). They also examined the strengths and weaknesses of the identified tools based on different criteria, such as user-friendliness, scalability, memory efficiency, visual styles, and post-visualization capabilities. Among the wide range of tools examined, they recommended Tulip, Cytoscape, Gephi and Pajek, all of which are stand-alone applications based on the user's level of expertise (Pavlopoulos et al., 2017).

Finally, Faysal and Arifuzzaman (2018) conducted a comparative analysis of large-scale visualization tools, which also relates to the current study. Their study aimed to identify popular visualization tools and factors that should be considered during visualizations and analysis of big data (Faysal & Arifuzzaman, 2018). They also examined the features and operations these systems supported to assess their performance and difficulties in visualising large networks. The tools selected for analysis included Cytoscape, Gephi, Pajek, SocNetV, and Tulip (Caldarola & Rinaldi, 2017). The analysis found that all of the tools can display and analyse large networks, although there are various degrees of applicability and scalability.

Overall, all the studies described above showed that data visualization tools are useful for visualising biological networks. These tools include Cytoscape, Pathway Studio, PATIKA, VisAnt, ProViz, BioLayout Express 3D, tools like Medusa for systems biology data, Pajek, Osprey, Tableau, Gephi, Microsoft Power BI, Plotly, Excel 2016, SocNetV, and Tulip, (Ali et al., 2016; Caldarola & Rinaldi, 2017; Faysal & Arifuzzaman, 2018; Pavlopoulos et al., 2008, 2017; Suderman & Hallett, 2007).

2.6. Summary

This chapter has reviewed 13 visualization tools, including Cytoscape, Osprey, Medusa, ProViz, CN-Plot, Ondex, MAPMAN, Pajek, MetaSHARK, BioLayout Express3D, Arena3D, CellNetVis, and Gephi. The tools cater to different disciplines, including biology, genomics, and complex system analysis. Accordingly, reviewing them strengthens the understanding of visualization applications across different fields. Additionally, each of the reviewed tools has unique features that are specific to their application domains and reviewing them helps the researcher understand their advantages and disadvantages and select the right ones for the current project. Visualization tools help users interpret data, recognise patterns, and gain insights. As this study involves complex data, these tools will be invaluable for uncovering existing but meaningful relationships. For instance, tools like Cytoscape and Gephi are widely used to visualise and analyse biological networks, such as PPI networks. Thus, understanding them is critical to the current project, as they are beneficial for understanding interactions in the network analysis conducted in this study.

Additionally, this chapter has provided a broad review of visualization tools that are relevant in different domains, which can help other researchers with comparative analysis and help them choose the most relevant tools for their projects. Furthermore, it has been demonstrated that visualization tools can help users interpret data, identify patterns and gain insights into complex data. This can strengthen understanding of visualization applications across different fields and promote their use.

The analysis above in sections 2.1 to 2.2 provides the methods, tasks, and patterns that can enhance the usability of complex biological networks, which answers research question 1.1 and objective 1. The analysis above in section 2.3 provides the different network visualization tools and techniques used for biological data. These applications include Cytoscape, Osprey, Medusa, ProViz, CN-Plot, Ondex, MAPMAN, Pajek, and MetaSHARK. Therefore, this answers research question 1.2 and objective 1.

Chapter 3: Interestingness Measures and Factors Impacting Visualization

This chapter starts by explaining the interestingness measures in detail and then discusses the factors considered when evaluating visualization tools using interestingness measures. The measures are grouped into two categories: general evaluation and heuristic factors. For the general evaluation, the factors included filtering tools, plugins, visual styles, advanced search, free or open source, efficient layout algorithms, scalability, different file formats, text mining, user input and customisation, graph analysis, feedback to users, strength, runtime performance, and user-friendliness. The heuristic factors were information coding or encoding, flexibility, orientation and help, minimal actions, prompting, consistency, spatial organisation, recognition rather than recall, removing the extraneous, and dataset reduction. The identified factors and their associated link with the interestingness measures are also provided in this chapter. These general and heuristic factors were evaluated to determine their importance for inclusion in visualization tools for complex biological networks, and the results can be found in Chapter 4.

3.1. Interestingness Measures

The concept of measuring interest is crucial in areas such as data visualization, information retrieval, recommending content, and designing user interfaces (Tan et al., 2002). The measurement of interestingness helps determine which information or content is most likely to attract the user's attention and fulfil their requirements. For instance, comprehending what users consider interesting in content recommendation systems allows the system to offer customised suggestions, thereby improving user satisfaction. When creating data visualizations, measuring "interestingness" helps guarantee that the visual display of data effectively conveys important findings and captivates the audience (Gupta and Chandra, 2020). It significantly contributes to enhancing user experience on different digital channels by customising content to align with user preferences and needs.

3.1.1. Key Attributes of Interestingness Measurement

The key attributes of interestingness measurement are provided below.

1. **Relevance:** This attribute assesses the extent to which the information aligns with the user's interests, inquiries, or needs (Bobi et al., 2023). Content that closely corresponds to the user's present circumstances or search query is more likely to capture and maintain their interest.

2. **Novelty**: The importance of the information lies in its originality or surprise factor. Original content grabs attention by presenting something new or unusual, piquing the user's interest and involvement. People are naturally attracted to material that provides innovative viewpoints or unfamiliar knowledge (Bhatnagar et al., 2008).
3. **Surprise:** This evaluates how much the information differs from what the user anticipates. Unexpected information can be very captivating as it questions assumptions or introduces an unforeseen turn, which makes the content more memorable and captivating (Gkitsakis et al., 2024).
4. **Utility:** The significance of information's practical utility is a key factor in determining its level of interest. Users highly value information that assists them in reaching objectives, resolving issues, or making well-informed decisions (Stansfield et al., 2006). Content that offers tangible advantages or addresses users' needs is more likely to be found interesting.
5. **Aesthetic Appeal**: Considering the visual and structural appeal of how information is presented is crucial. Thoughtfully crafted visuals, well-organized layouts, and attractive formats can elevate the perceived appeal of the data (Van der Geest and Van Dongelen, 2009). Visually pleasing presentations have the potential to improve accessibility and enjoyment of the information, ultimately leading to increased user engagement (Van der Geest and Van Dongelen, 2009).

Considering these important qualities when measuring "interestingness" enables a more thorough assessment of the potential of content to captivate users. In data visualization, it's vital to ensure that the data is essential, practical, and displayed uniquely and visually attractive to increase user engagement and interaction. Likewise, in systems that recommend content, grasping and utilizing these characteristics can improve personalization endeavors, resulting in more impactful and gratifying user interactions. By focusing on these characteristics, designers and developers can produce content and interfaces that more effectively address user requirements and desires, increasing involvement and contentment.

3.1.2. Methods of Interestingness Measurement

The following is the list of methods of interestingness measurement.

1. **User Feedback:** One way to directly measure how engaging something is to gather user input through different feedback channels. Getting input through surveys, ratings, and comments is a valuable way to understand how users perceive content. This method

enables users to express their likes, dislikes, and preferences, providing insight into what aspects of the content engage them or need improvement (Xin et al., 2006). User input can be collected at various points of engagement, offering instant responses and lasting views on the appeal of the content.

2. **Behavioural Analysis:** This approach examines how users behave to determine what they find engaging. Important measures such as click-through rates, dwell time (the duration users spend on specific content), and interaction patterns (such as shares, likes, or comments) can provide insight into user engagement levels. By studying these actions, experts and creators can pinpoint which material engages users the most. For instance, content that keeps users on the page for longer or receives frequent interactions is likely more captivating (Huynh et al., 2005). Analysing behaviour offers a data-oriented method for grasping user preferences without needing direct feedback from users.
3. **Content Analysis:** One alternative method is to evaluate the inherent qualities of the material, such as complexity, uniqueness, and relevance, to determine its potential attractiveness. Content analysis includes assessing how thorough the information is, how unique the insights are, and how relevant the content is to the user's needs and preferences. This method helps identify the natural qualities that make content engaging and can be used to improve and refine future content (Naveed et al., 2011).
4. **Algorithmic Approaches:** Using machine learning models and algorithms can greatly improve the measurement of what is considered interesting. These methods utilise past data and individual user information to foresee and order content appeal. By examining user behaviour patterns and trends, algorithms can produce tailored content suggestions that are anticipated to be engaging for each user (Vaillant et al., 2004). Machine learning models can constantly learn and adjust using new data, enhancing the accuracy and significance of predictions over time.

To effectively measure what makes content engaging and how to customise it to meet user preferences, it is essential to utilise a mix of user feedback, behavioural analysis, content analysis, and algorithmic approaches. Each method provides valuable perspectives that, when combined, offer a thorough understanding of interestingness.

3.1.3. Importance of Interestingness Measurement

Following is the list of identified importance of interestingness measurement for the system.

1. **Enhanced User Engagement:** Analysing interest is fundamental in data visualization, information retrieval, content recommendations, and user interface design. It involves assessing the level of user engagement, significance, and practicality of information or content to determine its appeal and usefulness. Key attributes of interest measurement include relevance, novelty, surprise, utility, and aesthetic appeal. Considering these attributes leads to a better understanding of how content captivates users. Methods for measuring interest include user feedback, behavioural analysis, and content analysis, offering valuable insights into user engagement and preferences (Arapakis et al., 2017).
2. **Improved Efficiency:** The assessment of interest level aids users in swiftly locating the most suitable information, thus enhancing the efficiency of their interaction with the system. By enabling users to readily access highly applicable content to their requirements, they can achieve their objectives more quickly and with minimal effort. Systems must be efficient in information-heavy settings where users might feel swamped by the amount of content. By emphasising the most engaging and pertinent information, systems can make the user experience more efficient and lessen the cognitive burden (McGarry, 2005).
3. **Personalization:** Adapting systems to suit each user's unique preferences is a major benefit of measuring interestingness. Understanding the specific interests of each user allows systems to deliver a more tailored and enjoyable experience (Riegger et al., 2021). Personalised content recommendations can significantly enhance user happiness by presenting information that closely aligns with their interests and needs. Tailoring experiences creates a feeling of closeness between the individual and the system, resulting in a higher chance of sustaining long-lasting involvement and commitment (Riegger et al., 2021).
4. **Better Decision-Making:** Emphasizing crucial and convincing data is vital to help users make informed decisions. It is essential to obtain the most relevant and interesting information, whether for professional purposes like data analysis and business intelligence or for personal use, such as making decisions about entertainment. Measuring the level of interest serves to shield users from an overload of irrelevant data, enabling them to focus on the most crucial information and thus make more informed decisions (Celotto et al., 2019).
5. **User Satisfaction:** Generating and delivering relevant and engaging content is closely associated with an increase in overall user satisfaction and loyalty. People tend to have a better experience on a platform when they consistently find content that matches their

interests and needs (Siro et al., 2023). When people regularly come across content that aligns with their interests and requirements, they are more likely to positively engage with the platform. It is vital for the continued success of any digital platform or service to maintain elevated levels of user satisfaction. This makes the measurement of interest a vital element of user experience strategy.

In summary, assessing interestingness is crucial for boosting user involvement, enhancing effectiveness, enabling customisation, facilitating informed decision-making, and improving user contentment. By paying attention to these elements, platforms can generate more engaging and impactful user interactions, leading to greater achievement and expansion.

3.1.4. Applications of Interestingness Measurement

Following is the list of applications of interestingness measurements.

1. **Search Engines:** Assessing what is considered interesting is crucial for enhancing the precision of search results. Search engines can offer users more valuable and compelling content by giving more importance to showing more engaging and relevant results. It enhances user satisfaction and productivity by allowing individuals to locate the required information quickly without sifting through irrelevant search results. Search engines can improve user satisfaction and effectiveness by giving more importance to captivating and pertinent content (Pon et al., 2011). This simplifies the process for users to find the information they require without going through irrelevant search results.
2. **Content Recommendation Systems:** Assessing the level of engagement with content on platforms such as streaming services, news websites, and social media can significantly improve the precision of recommendations. These systems can improve user engagement and satisfaction by recommending content that matches the interests and preferences of individual users. One way to generate personalised content recommendations is by utilising machine learning models to analyse previous behaviours, preferences, and interaction patterns (De Gemmis et al., 2015). This could improve the attractiveness of the content for users, leading to higher platform involvement and stronger commitment.
3. **Data Visualization:** Considering interestingness measurement in data visualization is essential to create visualizations that effectively emphasise the most compelling and valuable data points. This guarantees that users can easily understand important

observations and patterns without feeling swamped by unnecessary details (Otten, Chen and Drewnowski, 2015). Data visualizations can effectively convey intricate information interestingly and understandably by emphasising relevance, originality, and aesthetic attractiveness (Otten et al., 2015). This can ultimately assist in improving comprehension and decision-making.

4. **Educational Platforms**: Educational platforms must assess the engagement and interest of each learner to tailor learning materials to their needs effectively. Adapting the content to match the preferences and requirements of the learners can enhance their motivation, memory, and overall educational outcomes on learning platforms. Adaptive learning platforms can use measures of interest to suggest materials, tasks, and appropriate and captivating evaluations, thereby assisting individualised learning paths and enhancing educational impact (Valsamidis et al., 2011).

In general, assessing intriguing content improves the efficiency and user satisfaction of different online platforms by guaranteeing that the material and details are captivating, pertinent, and customised to specific requirements and preferences.

3.2. General Evaluation Factors

This section provides complete information regarding the general evaluation factors identified from the literature.

3.2.1. Filtering tools

In network visualization, filtering refers to the removal of nodes and edges that are considered less important or relevant for the analysis. According to Heberle et al. (2017), filtering is one of the options for improving network layouts. Freeman et al. (2007) also used a filtering tool to eliminate redundant probe sets in a genetic locus. Filtering tools also improve compatibility among various file formats in visualization systems, thereby aiding in the smooth integration and handling of different data types in network analysis. Table 3 presents two articles that used filtering tools as an evaluation factor in this study.

Table 3:Articles showing the filtering tools used for evaluation or review

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Baitaluk et al. (2006)	Filtering by the combination of attributes or node types.	Cytoscape and VisANT	While Cytoscape has flexible filters with different nodes and edge attributes,

			VisANT has several available 'select' filters.
Yeung et al. (2008)	The filters served as a more complex and flexible search method than Quick Find.	Cytoscape	The numerical attributes were filtered to determine the minimum and maximum values, and the nodes and edges within this range were identified.

3.2.2. Plugins

Plugins are additional features in visualization tools that can be seamlessly integrated into the network analysis tools. They are important in network visualization and analysis, as they are an important means for advanced users to extend and customise the applications. Several plugins, such as the Biological Networks Gene Ontology (BINGO) plugin, can be used to perform an analysis in a represented network. Table 4 shows two articles included in this study that focused on plugins as a factor for evaluation.

Table 4: Articles on plugins

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Schneider (2013)	This gave the users the means to perform sophisticated analyses and elaborated representation features.	Cytoscape	The tool assisted in accessing PPI repositories and the BINGO plugin for the GO enrichment evaluation of the resulting network.
Cline et al. (2007)	It was used to extend the functionality of the selected visualization tool.	Cytoscape, Osprey, VisANT, GenMAPP, BioLayout Express3D, PATIKA, CellDesigner, PIANA, ProViz	Additional functionality in areas such as download services and data integration.

3.2.3. Visual styles

A graphical representation is needed when visualising complex networks. This important feature of visualization tools enables users to easily modify the visual appearance of a visualised network. Table 5 presents two articles focusing on visual styles, such as graphical representation and colours, to visualise graphical networks.

Table 5: Articles showing visual styles

Reference	What did they use this factor for?	What tools were evaluated using this factor?	What was the outcome?
Kohl et al. (2011)	It was used to change the graphical appearance of the visualised network.	ProViz, VANTED, Cytoscape.	While most tools support a GUI to improve functionality, the functionality offered by most tools is often insufficient for the specified tasks.
Cline et al. (2007)	It was used to create experimental data in a network context.	Cytoscape	The expression data were mapped to node colours.

3.2.4. Advanced search

The advanced search feature is another relevant factor in identifying patterns in research databases. Table 6 shows two articles that identified visualization tools with advanced search, including Cytoscape, VisANT, Ondex Web. and Cytoscape Web.

Table 6:Articles focusing on the advanced search feature

Reference	What did they use this factor for?	What tools were evaluated using this factor?	What was the outcome?
Baitaluk et al. (2006)	It was used to select the analytical search tools to locate direct interactions and covering pathways.	Cytoscape and VisANT	Cytoscape enables node name search on the graph, whereas no search option is available in VisANT.
Taubert et al. (2014)	It was used to enter keywords or find information on regular expressions in the loaded network.	Ondex Web and Cytoscape Web	It helps the user obtain information relevant to the network inputs.

3.2.5. Free/open source

This refers to an application whose source code is made available to the public to allow users to view, modify, and freely distribute the code. This distribution model promotes the availability and affordability of visualization tools without the need for financial considerations. This factor is also necessary for users to quickly make decisions concerning combining multiple tools for effective results. While there are several open-source network

visualization systems, most are issued under a GNU or GPL license and are freely available. For academic use (not commercial exploitation). Table 7 presents two relevant articles which focus on free/open-source visualization tools.

Table 7:Articles that focused on free/open source solutions

Reference	What did they use this factor for?	What tools were evaluated using this factor?	What was the outcome?
Cline et al. (2007)	This allowed the use of the software and feature extension through programming.	Cytoscape, Osprey, VisANT, GenMAPP, BioLayout Express3D, PATIKA, CellDesigner, PIANA, ProViz.	The Cytoscape visualization tool integrated the networks with gene expression and other functional attributes.
Faysal and Arifuzzaman (2018)	It was used for integrating, visualising, and analysing network data.	SocNetV, Cytoscape and Gephi, Tulip, Pajek	They are all free and open-source programmes, except Pajek, which has commercial and non-commercial versions.

3.2.6. Efficient layout algorithms

This factor refers to layout methods, which are necessary elements of network visualization tools. A combination of different layout algorithms makes a network visualization tool an effective option for creating graph layouts. According to Suderman and Hallett (2007), one of the essential features of a network visualization tool is its ability to construct network layouts automatically. Examples of common layout algorithms that are used include force-directed, simple, Fruchterman and Reingold, tree, hierarchical, and multilevel, planar layout algorithms. While some of these layout methods are simple and conventional, such as the circular and grid layouts and the force-directed layout algorithms, other new algorithms are available, such as the multilevel layout algorithm, which offers improved performance with large-scale networks. Moreover, as noted by Pavlopoulos et al. (2017), a fast layout in large-scale network analysis is a form of restriction, as the layout algorithms that are most sophisticated end up becoming memory greedy and, therefore, require a lengthy running time to reach completion. Table 8 presents two articles that identified visualization tools that support efficient layout algorithms, including Cytoscape, Gephi, Cytoscape, Tulip, and Pajek.

Table 8:Articles that focused on efficient layout algorithms

Reference	What did they use this factor for?	What tools were evaluated using this factor	What was the outcome?
Yeung et al. (2008)	To move the positions of the network nodes and edges and reduce overlap.	Cytoscape	Applying the layout to the network produced a more vivid visual representation of data and made the network structure more interpretable.
Pavlopoulos et al. (2017)	To construct graph layouts.	Gephi, Cytoscape, Tulip, Pajek	Most tools have several sophisticated layout algorithms, although Tulip is highly recommended.

3.2.7. Scalability

Scalability in visualization tools is the ability to handle and effectively perform with different sizes and complexities of data. It helps to assess the effectiveness of the tool in accommodating larger datasets and more intricate structures while maintaining an acceptable performance level. In the comparison conducted by Faysal and Arifuzzaman (2018) regarding the scalability of their selected network visualization tools to large networks, they examined the memory consumed by the systems in megabytes and the number of seconds it takes in reading and displaying time by the selected tools. There is, therefore, a need for visualization tools to employ new approaches to provide access to large datasets within end-user limitations (Suderman & Hallett, 2007) to make the visualization function effective. Table 9 identifies two articles focusing on the scalability of the visualization tools.

Table 9: Articles that focused on scalability tools

Reference	What did they use this factor for?	What tools were evaluated using this factor?	What was the outcome?
Pavlopoulos et al. (2017)	It was used for basic visualizations of large networks.	Gephi, Cytoscape, Tulip, Pajek.	While Tulip is preferred for medium-scale networks, it is not as scalable as Gephi. Cytoscape does not scale well for large-scale analysis, whereas Pajek outperforms other tools and offers the highest scalability for network visualizations.

Faysal and Arifuzzaman (2018)	It was used to determine the ability and time it would take the visualization systems to read and display large networks.	SocNetV, Cytoscape Gephi, Pajek, Tulip	They identified Gephi and Cytoscape as effective tools for scaling very large networks because the tools can read and visualise all of the networks presented in the table.

3.2.8. Different file formats

This is a crucial factor to consider, as one of the challenges with most visualization tools is the complexity of input data. Most network graphs specify their input file formats in order to load and store networks. As a result, different tools cannot be easily used with similar datasets since the datasets need to be reformatted each time in line with the specific tool. This makes it difficult to explore the use of the different tools and the benefits of their complementary strengths. Further, while some visualization tools have strict file input formats, some offer various input format options. This factor is also important because it helps determine the file format that is acceptable or compatible with a specific network visualization system. The most commonly used file formats are BioPAX, SBML, PSI-MI, and CML. According to Pavlopoulos et al. (2017), visualizations should be able to load and store data in widely accepted file formats. Table 10 identifies three articles focusing on file formats evaluated with visualization tools.

Table 10:Articles focusing on file formats and visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Cline et al. (2007)	To import and export data into the system.	Cytoscape, Osprey, VisANT, GenMAPP, BioLayout Express3D, PATIKA, CellDesigner, PIANA, ProViz.	Create an image file of the network data using Cytoscape.
Pavlopoulos et al. (2017)	It was used to describe the network structure.	Gephi, Cytoscape, Tulip, Pajek.	Cytoscape is the most suitable, as it accepts several input file formats, compared to Gephi and Tulip, while Pajek is the least suitable because it is not flexible in its input file format.

Faysal and Arifuzzaman (2018)	To support diverse file formats.	SocNetV, Cytoscape Gephi, Pajek, Tulip	Compared to the other tools with diverse file formats, Pajek only supports files in Pajek.net format.

3.2.9. Text mining

Text mining is the process of extracting meaningful and valuable information from unstructured text data. It can also be called data analytics, as different techniques and algorithms are used to analyse, extract, and interpret patterns from a large collection of text-based data (West and Bhattacharya, 2016). It could entail assessing how well the tool can extract meaningful patterns and information from data and visually represent the data. Table 11 describes the text mining evaluation tools identified in one article, including Cytoscape, VANTED, and ProViz.

Table 11: Text mining and visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor**	**What was the outcome?**
Kohl et al. (2011)	This helped to derive high-quality information from the specified texts.	Cytoscape, VANTED, ProViz.	This selected tool provided flexible and advanced text mining capabilities.

3.2.10. User input and customisation

User input and customisation refers to the ability of users to interact with and tailor the visual representation of data based on their specific needs and preferences. This is a functionality that enhances the flexibility and usability of visualization tools, as it permits users to refine the visual output based on the focus of the analysis. Table 12 shows an article identifying the user and input factors used in evaluating the Pathway Studio, Osprey, Cytoscape, ProViz, VisANT, and PATIKA visualization tools.

Table 12:User input and customisation and visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Suderman and Hallett (2007)	It allows users to extend the application and customise the	Pathway Studio, Osprey, Cytoscape, ProViz, VisANT, PATIKA.	While most tools support a GUI to improve functionality, the functionality offered by most

	network input according to their needs.		tools is often insufficient for the specified tasks.

3.2.11. Graph analysis

This factor involves the examination and interpretation of graphs, as they are mathematical structures that represent relationships between entities. Visualization tools with graph analysis capabilities can allow users to explore visually complex networks, helping to reveal patterns, structures, and insights present in a particular dataset. Table 13 presents an article identifying the graph analysis factor used to evaluate visualization tools, such as Cytoscape, MAPMAN, Ondex, PATIKA, and Osprey.

Table 13: Graph Analysis factor and Visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Kohler et al. (2006)	This was used for the exploration and interpretation of biological data.	Cytoscape, MAPMAN, Ondex, PATIKA, Osprey	Unlike other graph-based systems, Ondex supported mapping and automated data linking from various heterogeneous data sources.

3.2.12. Feedback to users

This mechanism is used to report the analysis and visualization of the operation process or outcome to the concerned users. Some visualization tools offer this specific feature with options to print, save, or copy the reports as and when needed. Faysal and Arifuzzaman (2018) considered this a necessary tool for network visualization (see Table 14).

Table 14:Feedback to users and visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Faysal and Arifuzzarman (2018)	It was used as a reporting strategy, with options for users to print, save, or copy the reports.	Gephi, SocNet, Cytoscape, Tulip.	Only Gephi and SocNet were found to have good reporting strategies. Tulip and Cytoscape only offer limited options for users to save the resultant graphs from the operations in specific formats.

3.2.13. Strength

There are positive features that make a visualization tool effective and valuable for a certain purpose. Visualization strength adds to the usability, utility, and overall impact of the tool in terms of assisting users in understanding and interpreting data. Table 15 presents one article that reported the use of a strength factor in evaluating visualization tools.

Table 15:Strength factor and visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Pavlopoulos et al. (2017)	This factor was used to determine the strong points of the selected tools, which are different from other available tools.	Medusa, Cytoscape, BioLayout Express3D, Osprey, Ondex, ProViz, PIVOT, PATIKA, Pajek.	The strength of Pajek is its variety of layout algorithms, while PIVOT, Medusa, and ProViz are best suited for PPI visualization. PATIKA enables efficient visualization of transitions, BioLayout Express3D offers various approaches to microarray data analysis, and the filtering capabilities of Osprey make it a powerful tool for network manipulation. Ondex's strength is combining heterogeneous data types in one network, while Cytoscape's strength is its visualization of molecular networks.

3.2.14. Runtime performance

Runtime performance refers to how efficiently and rapidly the tool can process, render, and show visual representations of data while executing an application. This is related to the speed of the network visualization system in delivering user projects. Table 16 shows an article that reported the use of runtime performance to evaluate various visualization tools, such as Cytoscape, Gephi, Tulip, Pajek, and SocNetV.

Table 16:Runtime performance factor and visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Faysal and Arifuzzaman (2018)	This factor was used to evaluate the tools' community detection ability, layout algorithms, and measures of prominence.	Cytoscape, Gephi, Tulip, Pajek, and SocNetV	This factor was used to exclude other visualization tools that did not possess this feature, such as SocNetV and Cytoscape.

3.2.15. User-friendliness

A user-friendly interface will facilitate easy accessibility, analysis, and quick network visualizations. Table 17 identifies an article that reported user-friendliness factors used to evaluate different visualization tools, such as Cytoscape, Pajek, Gephi, and Tulip.

Table 17: User-friendliness factor and visualization tools

Reference	**What did they use this factor for?**	**What tools were evaluated using this factor?**	**What was the outcome?**
Pavlopoulos et al. (2017)	To implement user-friendly interactive tools.	Cytoscape, Pajek, Gephi, Tulip	Tulip is the strongest in terms of user-friendliness, compared to Gephi and Cytoscape, which are good and medium in user-friendliness, while Pajek is weaker.

3.3. Heuristics Evaluation Factors

Heuristic factors are guidelines that individuals adopt in solving problems, making decisions and simplifying complex tasks (Gigerenzer and Gaissmaier, 2011). They are shortcuts adopted by people when facing uncertainty or incomplete information. They are commonly used in biological network analysis, particularly because they provide practical and computationally efficient approaches to complex problem-solving. Particularly, heuristic factors are used in this study because they offer computational efficiency; since biological networks can be massive and complex, heuristic algorithms provide a faster solution compared to exact algorithms, which then makes them more useful in handling large-scale network data timely. Heuristic factors are critical in the current research because they guide the design and implementation of effective biological network visualizations. For instance, in terms of user-centric design, the factors help to prioritise the needs and preferences of end-users while designing the tool.

Table 18 below identifies three articles that reported heuristic-evaluated factors that were used in visualization tools called visualization heuristics, such as information coding, flexibility, orientation and help, minimal actions, prompting, consistency, spatial organisation, recognition rather than recall, removing the extraneous, and data set reduction.

Table 18:Heuristic evaluation factors and visualization heuristics

Visualization Heuristics	Authors	Description
Information Coding	Forsell and Johannson (2010)	The use of realistic techniques and additional symbols to improve information perception.
	Vaataja et al. (2016)	It is useful for information visualization by mapping data objects to visual elements, such as graphics, symbols, and visual cues.
	Williams et al. (2018)	The mapping of datasets to visual elements.
Flexibility	Forsell and Johannson (2010)	Refers to how the interface can easily adapt to the specific needs of users.
	Vaataja et al. (2016)	Flexibility in network visualizations refers to easy access and available means for users to customise the interface of the visualization tools to understand the processes, working strategies, and task requirements.
	Williams et al. (2018)	The system should provide options for the user to achieve the same goal using different tasks or steps.
Orientation and Help	Williams et al. (2018)	Functions that support users in controlling the levels of detail, action, and representation of additional information.
	Forsell and Johannson (2010)	This involves providing functions to support users in controlling a wide range of details.
	Vaataja et al. (2016).	Using undo/redo options and additional functional information enables users to navigate the system easily to visualise or analyse a complex network.
Minimal actions	Vaataja et al. (2016).	This refers to the extent of workloads based on the number of actions required to complete a task.
Prompting	Forsell and Johannson (2010)	This refers to using a guide or prompts to support users in taking specific actions and providing alternatives within the system. This can be done through data entry or by serving as a guide in performing other tasks.
Consistency	Forsell and Johannson (2010)	Describes how the choice of interface designs, such as naming, formats, codes, and procedures, are maintained in a similar context in a way that differs from that when applied in different contexts.
	Vaataja et al. (2016).	This enables the system to become more predictable and improves learning and generalisations, including errors with the use of the system.
Spatial Organisation	Forsell and Johannson (2010)	This refers to the orientation available to users in the information space, efficiency in space usage, distribution of layout elements, legibility and precision, and alteration of the visual elements.
	Williams et al. (2018)	The overall layout of a visual representation is used to analyse the ease of locating an information element on the display.

Recognition rather than recall	Vaataja et al. (2016)	Used to reduce users' memory load by making actions, instructions, and objects more visible and easier to recognise or retrieve.
	Forsell and Johannson (2010)	To reduce the burden placed on users to recall information while visualising or analysing information or networks.
Remove the extraneous	Forsell and Johannson (2010).	This involves presenting the largest amount of data with a small amount of ink by determining whether additional information will distract or limit the visualization process.
Dataset reduction	Forsell and Johannson (2010)	This will help users to redirect their focus to more areas of interest or relevance and understand the available datasets.
	Vaataja et al. (2016)	Features are included in a system to reduce the dataset and improve their efficiency and ease of use through methods like filtering, clustering, and pruning.

3.4. The link between Interestingness Measures and General and Heuristic Factors

Interestingness measurement is closely linked to both general evaluation factors and heuristic factors in the context of data visualization and user interfaces. The overall assessment criteria include broad elements like accuracy, relevance, and thoroughness, ensuring that the material effectively meets users' informational needs. Meanwhile, heuristic factors involve intuitive guidelines such as simplicity, clarity, and visual attractiveness, guiding the creation of user-friendly interfaces and visual representations that are functional and engaging. Integrating the measurement of interest with these factors allows designers and developers to create data visualizations and user interfaces that are both informative and captivating, enhancing the overall user experience. This comprehensive approach ensures that content is relevant, accurate, engaging, and easy to use.

General Evaluation Factors

1. **Filtering Tools**: Effective filtering tools can help isolate interesting data by removing irrelevant or less attractive information.
2. **Plugins**: Plugins can extend the functionality of systems to capture and measure what users find interesting better.
3. **Visual Styles**: Attractive and clear visual styles can enhance the perceived interestingness of data by making it more accessible and engaging.
4. **Advanced Search**: Advanced search capabilities enable users to find interesting information more quickly and accurately.

5. **Free/Open Source**: Open-source tools often have community-driven improvements that can enhance interestingness through collaborative innovation.
6. **Efficient Layout Algorithms**: Efficient algorithms guarantee that the most captivating information is displayed in a visually attractive and easy-to-comprehend format.
7. **Scalability**: Systems that efficiently manage large datasets can preserve valuable patterns amidst abundant data.
8. **Different File Formats**: Expanding the range of supported file formats enables greater flexibility and thoroughness in data analysis.
9. **Text Mining**: Text mining methods can reveal intriguing patterns and trends within written data.
10. **User Input and Customization**: Enabling users to enter their preferences and personalise the interface guarantees that the most captivating information stands out.
11. **Graph Analysis**: Graph analysis techniques can reveal exciting relationships and structures within data.
12. **Feedback to Users**: Feedback helps users understand what is considered interesting and why, improving their interaction with the system.
13. **Strength**: The robustness of a system contributes to its ability to present exciting data consistently.
14. **Runtime Performance**: High performance ensures that users can quickly access interesting information without delays.
15. **User-Friendliness**: A user-friendly interface enhances the overall experience, making interesting information more accessible.

Visualization Heuristics

1. **Information Coding**: Effective information coding (e.g., colour coding and symbols) makes it easier to identify interesting data points.
2. **Flexibility**: Flexible systems can adapt to user needs, presenting exciting information in various formats and contexts.
3. **Orientation and Help**: Clear orientation and help features guide users to exciting parts of the data.
4. **Minimal Actions**: Reducing the number of actions required to access exciting information keeps users engaged and reduces frustration.

5. **Prompting**: Prompts can draw attention to potentially interesting data that users might otherwise overlook.
6. **Consistency**: Consistent design and interaction patterns help users quickly identify and understand exciting information.
7. **Spatial Organization**: Effective spatial data organisation ensures that exciting information is easily visible and understandable.
8. **Recognition Rather than Recall**: Designing systems that facilitate recognition over recall helps users quickly identify exciting data.
9. **Remove the Extraneous**: Eliminating extraneous details directs users' focus toward what is genuinely captivating.
10. **Dataset Reduction**: Methods for condensing datasets to their most essential components help prevent users from being inundated with excessive amounts of data.

In summary, interestingness measurement is integral to designing engaging, efficient, and user-friendly systems. It combines general evaluation factors and heuristic principles to ensure that the most relevant and captivating information is effectively highlighted and accessible to users.

3.5. Summary

This section discussed the factors considered in evaluating visualization tools, which are classified as general evaluation and heuristic factors. For the former, the study adopted factors including filtering tools, plugins, visual styles, advanced search, free/open source, efficient layout algorithms, scalability, different file formats, text mining, user input and customisation, graph analysis, feedback to users, strength, runtime performance, and user-friendliness. Each of these factors was examined in line with the literature. For instance, Cytoscape and VisAnt were the tools evaluated in terms of advanced search, and the literature showed that Cytoscape enables node name searches on graphs, whereas no search option is available in VisANT. Gephi, Cytoscape, Tulip, and Pajek were used in the literature to evaluate different file formats. Cytoscape was found to be the most suitable, as it accepts more input file formats than Gephi and Tulip, while Pajek accepts the least. Regarding heuristic factors, the study adopted visualization heuristics, including coding, flexibility, orientation and help, minimal action, prompting, consistency, spatial organisation, recognition rather than recall, and removing the extraneous and dataset reduction. These

were described in different studies. For instance, information coding refers to the use of realistic techniques and additional symbols to improve information perception, while flexibility involves easy access and available means for users to customise the visualization tool interface to understand the processes, working strategies, and task requirements better.

Basically, through the application of the two major types of evaluation on widely used visualization tools such as Cytoscape, VisANT, Gephi, Tulip, and Pajek. Cytoscape was found to be the most suitable for most of the factors, while VisANT and Pajek were the least relevant. It was also discovered that the tools had different advantages and disadvantages. However, this depends on the data type and size, including the users' tasks and goals and the level of customisation and flexibility needed. These, therefore, contribute to the field of visualization research by providing a systematic framework for the evaluation of the tools and insights to improve the designs and development of future tools.

Sections 3.1 and 3.2 show how the concept of interestingness and visualization support complex biological networks. The interestingness measures were grouped into the general evaluation and heuristic factors. The analysis shows that the concepts interestingness and visualization support complex biological networks through filtering tools, plugins, visual styles, advanced search, free or open source, efficient layout algorithms, scalability, different file formats, text mining, user input and customisation, graph analysis, feedback to users, strength, runtime performance, and user-friendliness, information coding or encoding, flexibility, orientation and help, minimal actions, prompting, consistency, spatial organisation, recognition rather than recall, removing the extraneous, and dataset reduction. Therefore, this answers research question 2.1 and objective 2. Similarly, the different application patterns of the visualization techniques for complex networks include Cytoscape, VisANT, GenMAPP, BioLayout, VANTED, PIANA, and Gephi, among others. This answers research question 2.2. and does justice to objective 2.

Chapter 4: Evaluation of the Factors

This chapter aims to evaluate the general and heuristic factors identified in Chapter 3 of the literature in order to determine their importance and significance to users based on their perspectives revealed in their survey and interview responses. It also emphasises the importance of identifying the attributes that are most critical for the successful and efficient visualization of complex networks, and it assists developers in incorporating those factors when developing new tools. The evaluation used a mixed methodology, including interviews and surveys. An interview was used to gather qualitative data, and a survey was used to gather quantitative data. The questions are presented in the appendices I and II sections. The interview was conducted with 5 domain experts in biomedical sciences, genetic and molecular biology, immunology, laboratory research and molecular and cell biology, and the survey was conducted with 98 participants with experience in data visualization. After identifying the most important and critical factors, the five visualization tools (Medusa, Cytoscape, Arena 3D, Gephi, and Graphia) were evaluated based on those most important factors to assist users in determining the limitations and strengths, which would guide them in selecting the best tool for specific purposes. This is a critical step in this thesis because it is crucial to understand the current limitations and strengths of existing visualization tools to determine whether it is necessary to develop a new tool or expand the range of existing tools that can be used to visualize complex biological networks. The evaluation outcome of the five tools can be found in Chapter 5.

4.1. Method

According to Saunders et al. (2016), research choice concerns either qualitative or quantitative methods, a simple or complex mix of both, or the use of a single process. Quantitative research involves numbers and mathematical operations, while qualitative research involves collecting vast descriptive data. There are mono, mixed, and multi-methods in research methodology.

The mono method is adopted when the study focuses on qualitative or quantitative data collection. It is applied when the research aims to understand certain concepts, experiences or social phenomena deeply. Therefore, it is best applied when the research question is narrow and well-defined but focuses on a specific part that can be comprehensively studied using a single approach.

Multi-method underscores the use of both qualitative and quantitative methods; while the research is based on one, the other method serves as an auxiliary or supplementary (Melnikovas, 2018). The approach may involve several quantitative and qualitative methods or a combination of both methods, applicable simultaneously or sequentially. Multi-method qualitative research uses different qualitative techniques to collect data. This may combine interviews, focus groups and ethnography. Multi-method quantitative research adopts quantitative techniques such as combining surveys, experiments and secondary data analysis to collect the needed data. Interdisciplinary multi-method research integrates methods from different disciplines to answer research questions across fields. Multi-method research is best applied when the research question is broad and multifaceted and requires a series of methods to capture the full scope of the phenomenon, which often involves interdisciplinary perspectives (Adu et al., 2022).

The mixed method uses qualitative and quantitative methods in one research to achieve different aims and offset the constraints of using a single method (Bahari (2012). It leverages the strengths of both methods to understand the research problem thoroughly. It can be classified into two. First is the concurrent mixed method, which means qualitative and quantitative data are obtained simultaneously and integrated during the analysis phase (Fetters, Curry and Creswell, 2013). The second is a sequential mixed method. This means that data is obtained in phases. Qualitative data could be collected first to explore a phenomenon, while quantitative data is collected afterwards to measure it. Mixed methods are best applied when the research question is complex, and understanding ultimately requires numerical measurements and deep exploration of contexts and experiences (Fetters et al., 2013).

The current study is a mixed-method study that combines quantitative and qualitative methods and involves sequential or concurrent methods based on the study design (exploratory or explanatory) (Terrell, 2012). According to Dawadi et al. (2021), a mixed-methods study uses multiple research methods to provide the answer to the research question. This type of study is beneficial, as it allows researchers to navigate from one research method to another to answer a research question and converge or confirm the research findings (Terrell, 2012). Another benefit of mixed-methods research is that it can solve complex research problems by integrating positivism and interpretivism philosophical frameworks, which combine quantitative and qualitative methods to answer a research question (Dawadi et al., 2021). Additionally, using mixed methods represents a flexible research approach, helping to provide an in-depth understanding of a research issue (Maxwell, 2016). However, conducting a mixed-

methods study involves a good knowledge of research methodology and the ability to interpret findings from different research methods (Terrell, 2012). Nevertheless, the mixed-methods approach is suitable for the current study since both qualitative and quantitative data are required for evaluation in general.

4.1.1. Data collection

Qualitative data are textual data used in qualitative research, which seeks to explore and provide a deeper understanding of real-world issues (Tenny et al., 2022). Qualitative data help the researchers get closer to the phenomenon being studied (Aspers & Corte, 2019). Although the use of qualitative data in research has some limitations, such as smaller sample sizes, high subjectivity, and significant time requirements, it has some important benefits (Rahman, 2016). Qualitative data help researchers understand feelings and experiences and obtain new ideas relevant to a research question in a qualitative study (Ugwu & Eze, 2023). Meanwhile, quantitative data are numerical data used in quantitative research, which provide relevant information regarding the proportions, percentages, and levels of a phenomenon under investigation (Apuke, 2017; Kabir, 2016). In other words, quantitative data reduces a phenomenon into a numerical format for the purpose of statistical analysis (Apuke, 2017). Quantitative data have several benefits compared to qualitative data when conducting quantitative research, including objective results, a larger sample size, and lower time requirements (Rahman, 2016). The current study employed a mixed-methods approach that included qualitative and quantitative data, as both types were beneficial to achieving the aim and objective of the study.

The data collection method used in the qualitative part of the current study is the interview method. The use of interviews allows for knowledge/information generation from the participants, as it helps researchers to understand the existing assumptions of theories and results in the inductive generalisation of new theories (Dunwoodie et al., 2023). The survey tool benefitted the current study in the following ways: It allowed for collecting large amounts of data, led to time and cost savings, and offered a higher chance of achieving accurate results, as statistical tools can be used to analyse such data (Taherdoost, 2021). Hence, the data collection and research methods were helpful in achieving the aim of this study. Closed questions were used for the interview and survey. This is to ensure that respondents' responses follow a consistent pattern, making comparison and avoiding off-topic answers easier. However, unlike a survey, an interview can provide clarification as needed. Thus, conducting

interviews is essential for this objective in order to get comprehensive knowledge from domain experts.

Online interviews using Zoom and Skype were conducted with the participants in academia. This included five bioinformaticians whose years of experience ranged from 4–5, 6–10 and more than 10 years. The conversations were recorded as audio files and later documented in Microsoft Word. The data consists of the demographic details of the participants, their awareness of graph visualization tools, and the general and heuristic factors. For each interview question, the details of which are presented in Appendix B, the participants were allowed to express themselves fully, which is crucial for thematic analysis.

The Newcastle University online survey was used and is presented in Appendix A. The participants first answered demographic questions, followed by questions relating to the project. Each survey question, whose details are found in Appendix A, has five different response categories: strongly agree, agree, neutral, disagree, and strongly disagree. These are also the levels of agreeability. The questions were grouped. For instance, there are five different survey questions related to the visual styles factor of the network visualization tools. There are 138 columns and 98 rows on the questionnaire. Ideally, this implies that each participant answered about 138 survey questions and that the study includes 98 participants with prior experience in the use of visualization tools (e.g. data science and data visualization).

4.1.2. Data analysis method

The responses from five participants were analysed using thematic analysis. Thematic analysis (TA) is a useful research method in qualitative research, which examines systematic identification, organisation, and patterns of meaning across a dataset. Basically, the focus is on the meaning of the dataset, enabling the researcher to observe and deduce collective meanings and experiences. The focus of thematic analysis is not limited to identifying unique and idiosyncratic meanings and experiences. It is also a method for identifying commonalities in the content of a topic of conversation or document and making sense of those commonalities (Braun & Clarke, 2012). Excerpts were extracted from the participant's responses and placed in a column entitled 'Interview extract'. Then, codes were formed from the excerpts, and general themes were formulated. The codes were made to be relevant to the quantitative analysis (survey analysis), and the themes captured both codes and the interview extracts. This is because the participants' responses to most of the interview questions were 'yes' and 'no'.

As a result, ‘strongly agree’,’ ‘agree’, ‘neutral’, ‘disagree’, and ‘strongly disagree’ were used as the codes, while ‘essential’ and ‘not essential’ were used as the themes.

The data derived from the participants’ responses were analysed using IBM SPSS version 25. Content analysis was performed on the responses to each of the survey questions to extract the frequency of each of the levels of agreeability (strongly agree/very important = 1, agree/important = 2, neutral = 3, disagree/less important = 4, and strongly disagree/not important at all = 5). The original data were ‘string’, which is the level of agreeability. The data were transformed into a five-point scale to support the analysis, which can only be done on a ‘numeric’ data type. The frequency of each level of agreeability is expressed as a percentage; however, the frequency tables are attached for a detailed description of the responses. The level of agreeability with the highest percentage was used to identify the views of most of the participants. Additionally, the percentage of the levels of agreeability expressing the same general view, such as ‘agree’ and ‘strongly agree’, which support the survey questionnaire, were added together to identify categories of participants with a certain view and those with different views regarding the survey questions.

4.2. Interview Findings

4.2.1. General factors

Filtering tools

The interview findings are presented in Table 26 in Appendix C. The interview analysis in the table shows that all the interviewees agreed that filtering tools are essential in network visualization tools. According to one of the respondents, ‘filtering tools are really key in network visualization tools, as they help increase the reliability of the visualization tools.’

Plugins

Further, the data indicate that the complexity of the network visualization tool is the major downside to having plugins. The respondents stated that such complexity would negatively affect the plugin of the network visualization tool, and they advocated for the simplicity of the network tool. The findings presented in Table 27 in Appendix C show that the respondents agreed that plugins should be free and available to researchers. Of the five people interviewed, four stated they would be happy if plugins were made available for all researchers for free. The interview analysis revealed that the respondents believed there should be special functionalities in plugins in graph and network visualizations. One of the respondents remarked that ‘having a special type of functionality plugin will make it easier to use network visualization tools.’

Visual style

Table 28 in Appendix C shows that the interviewees agreed that 2D is a better choice than 3D, making it the most preferred by the respondents. Of the five interviewed, three stated they would use a 2D network visualization tool. This implies that 3D, although not captured in the interview, is still associated with potential complexities. This makes it difficult for the respondents to use 3D visualization tools as easily as 2D ones. Further, when asked if gaining insights into the associations present in a graph is a function of the visual style of the network visualization tool, all the respondents answered affirmatively, indicating that this would improve the visual style of the tool.

Advanced search

The participants' responses in Table 29 in Appendix C show that many respondents believe that having a visualization tool capable of searching academic web pages and online databases would be extremely useful and welcome. Also, the respondents agreed that searching out a group is an advanced search in network visualization. This implies that users prefer network visualization tools with the ability to search for a group. On the other hand, a significant number of respondents disagree that there is a need for the network visualization tool to consider regular expression as part of advanced search. Respondents also do not consider the ability of a graph visualization tool to perform an advanced search to be pivotal to their choice of using a visualization tool. According to some of the respondents, that factor is not useful to them.

Open source

As shown in Table 30 in Appendix C, open source is something the respondents look for in a graph visualization tool. However, they prefer free tools to commercial ones. One of the respondents said that being able to use the free tool might convince him to choose the paid one later. The respondents also noted that free graph visualization tools can be used by anyone without restrictions. However, a few respondents said they had not used commercial graph visualization tools often, and thus, they may not have used the tool long enough to understand whether the free version has as many features as the commercial version.

Efficient layout algorithms

As shown in Table 31 in Appendix C, the respondents stated that an efficient layout and algorithm are very important in network visualization tools. They further noted that ease of use is the major factor they consider when judging the efficiency of the layout algorithm in a network visualization tool. The respondents agreed that the efficiency of layout algorithms is dependent on the tool being used. From their perspective, efficient layout algorithms are much

better with a better tool and vice versa. Finally, many of the respondents indicated that they think it is a good idea for graph visualization tools to show the transition of graph visuals into a specific layout.

Scalability

The respondents strongly agreed with all the scalability points in the interviews (Table 32, Appendix C). Regarding whether they expect graph visualization tools to produce visual networks since real-life networks are complex, all the respondents agreed that they should do so. Further, the interview respondents agreed that benchmark complexity should be simpler to ensure users do not have problems while using the network visualization tool. All the respondents felt that the choice of scalability is a key aspect of graph visualization tools.

Different file formats

Table 33 in Appendix C shows the respondent's responses regarding different file formats. They stated that it is important for graphs to work with different file formats. As one respondent argued, this feature makes network visualization tools unique and more appealing to users. Moreover, the respondents agreed that network visualization tools should be able to import networks stored in other file formats. With this ability, it would be easy to share files with different network visualization tools. The respondents also agreed that file formats are important in facilitating the use of graph visualization tools. As one noted, 'accepting different file formats makes it easier to use graph visualization tools when the job at hand is a high priority'.

Text mining

Table 34 in Appendix C presents the participants' responses regarding text mining. When asked if a visualization tool should be able to perform semantic analysis, three of the five interviewees stated that if it could, it would improve the researchers' work. Further, the participants maintained that text mining should be a default feature in text visualization tools, not just a plugin. This, they believe, would make the use of the tool easier and more appealing. While many of the participants agreed that text mining does not affect their choice of graph, they believe that graphs and charts are the main text mining techniques every visualization tool should include. They stated that developers of the tools should consider this important aspect of the software.

User input and customisation

Table 35 in Appendix C presents the respondents' responses regarding the subject matter. User input and customisation appeared to be very important to the participants in the interview, as

they strongly agreed with all the related questions. They felt that zooming should be possible with a touchpad, not just with a mouse. They also agreed that visualization tools provide customisation options for users, that they should support different parameters with charts and graphs, and that user input and customisation are important considerations when selecting a network visualization tool. In fact, two of the participants indicated that user input and customisation should be the first consideration of anyone using a network visualization tool, as it makes the work of the users easier.

Graph analysis

According to the results presented in Table 36 in Appendix C, the participants want graph visualization tools to be used for graph analysis even though they are widely used for network visualization. The participants stated that having an analytical graph feature would make it easier for analysts to combine these features for better results. As one of the participants noted, 'This should be fun to work with'. Again, the participants indicated that graph visualization should be capable of working with Google charts, Tableau, and Chartist. Some stated that it should support principal component and correlation analyses. While the preferred choice varied among the participants, correlation analysis was mentioned by four of the five participants, an indication of its importance as a network visualization tool.

Feedback to users

The results presented in Table 37 in Appendix C show that user feedback on questions helped the participants realise the importance of including a progress bar in a graph visualization tool. As some of the participants stated, this helps them know if progress is occurring while carrying out their analysis. The participants agreed that a bar chart is one of the best formats for graph visualization tools, and it should be possible to export the results. The participants felt that exporting files as Word, PDF, and Excel documents was also good. All the participants agreed that user feedback is a key factor to consider when selecting a visualization tool.

Strength

The feedback from the participants is presented in Table 38 in Appendix C. An important finding is that the strength of a visualization tool is tied to its ease of use by the user. According to the participants, users prefer stronger visualization tools. This consists of having strength in all the areas combined with ease of use and a range of display options (e.g. 2D or 3D). The participants noted that user-friendliness and ease of use are strengths they look for when choosing a network visualization tool.

Runtime performance

The details of the participants' responses regarding runtime performance are presented in Table 39 in Appendix C. They indicated that runtime performance is a crucial aspect of a visualization tool. All the participants agreed that they primarily select a visualization tool based on its speed. They also consider how quickly it can switch from one layout to another. These two runtime performance characteristics are the main criteria determining the participants' choice of a network visualization tool.

User-friendliness

All the participants agreed that there is a need for the screen to be maximised, as it enhances the user experience and improves the tool's user-friendliness (Table 40, Appendix C). They stated that a user-friendly visualization tool should include tool tips, self-explanatory button names, a simple interface, and movable panes. According to the participants, these features will make it easier for users to navigate and use the tool efficiently.

4.2.2. Heuristic factor

Information coding

Information coding ensures that the codes, nodes, and other information input into the network visualization tool are easily coded using different symbols or colours. Based on the interviews (Table 41, Appendix D), the participants agreed that visualization tools should be able to swap source and target nodes, appropriately separating nodes to make it easier for users to identify them. The participants also indicated that different line types should be used to represent edges. They noted that it is important to use colours to identify different nodes and lines, thereby making colour information coding one of the most important types of coding in the development of network visualization tools.

Flexibility

As shown in Table 42 in Appendix D, all the participants identified ease of use as the major criterion they would consider when selecting a visualization tool, concluding that a good tool must be flexible. 'It must not be complex', noted one of the participants, who also stated that it should be compatible with other types of graph visualization tools. The participants also suggested that graph visualization tools should support all possible layouts by default, thereby reducing the need to have these layouts as plugins. The implication of this is that the participants consider plugins to be complex and cumbersome.

Orientation and help

The results presented in Table 43 Appendix D indicate that the participants want orientation and help available offline so that users can access them without being connected to the network. One participant lamented, 'It could be frustrating trying to go online and get the information one is looking for, especially when the network is out of coverage.' The participants agreed that having online communities for support helps with the issues that both experienced and new users may encounter when using network visualization tools. I agree that orientation and help can be in the first stage of the visualization tool. The participants also agreed that textual, audio, and video formats can be used for the orientation and help page.

Minimal actions

As shown in Table 44 in Appendix D, the participants indicated that network visualization tools should have the minimum number of steps needed to produce the required results. When asked about the number of steps that should be taken before performing a task in a network visualization tool, all the participants indicated that it must not exceed two steps at most. They also agreed that there should be minimal steps when importing data. Thus, the participants consider minimal actions to be an important factor when they are choosing a network visualization tool.

Prompting

The interview participants (Table 45, Appendix D) regarded prompting as an important feature, even though it can sometimes slow their work. They also agreed that alert boxes do not distract them when using the visualization tool. Hence, network visualization tool developers can use alert prompt boxes to pass on key messages to users while they are utilising the visualization tool. For example, a prompt message box could alert users when they are about to delete messages or suggest fixes to errors. However, the participants did not like the idea that prompt boxes could prevent the user from using the tool. According to one participant, this represents an 'unwanted restriction'.

Consistency

There was a general agreement among the participants that visualization tools should utilise graph notation based on graph theory (Table 46, Appendix D). They also agreed that lines and icons should be used to represent graph edges and buttons. One participant said, 'It makes their lives and work a lot easier.' Further, the participants indicated that consistency is a key factor in choosing a visualization tool.

Spatial organisation

According to the results presented in Table 47 in Appendix D, the participants indicated that zooming, spate windows, and having an adjustable screen space are good features in a visualization tool. These features enhance the work of users, making it easier for them to remain organised. Thus, the participants felt that spatial organisation is important and should be considered when selecting a visualization tool.

Recognition rather than recall

When answering the questions on recognition rather than recall, the participants agreed on the simplicity of the features. For instance, they agreed that icons, not names, should be used. They also stated that nodes should have different colours for easy identification and that the thickness of edges should be appropriate to improve the readability of graphs (Table 48, Appendix D).

Remove the extraneous

The interview participants, whose responses are presented in Table 48 in Appendix D, indicated that they would like to see many features made available on the screen instead of in the menus. This would improve accessibility and reduce the number of actions to perform. However, they suggested that having the same feature somewhere in the menu is important so that the user can access it from different points. All the participants rejected the idea that pop-ups should be made part of a visualization tool. As one participant said, 'It is so distracting and annoying.' Thus, network visualization tools should have few pop-ups and only those that are extremely very important.

Dataset reduction

The participants reported that they would also like to see a reduction in the amount of data in visualization network tools (Table 49, Appendix D). Moreover, they stated that graph visualization tools should support node reduction. Finally, all the participants insisted that dataset reduction is an important consideration when choosing a network visualization tool.

4.3. Survey Analysis

4.3.1. Reliability test

The reliability test indicates the degree of consistency in the participants' responses across the survey questions. Cronbach's alpha values for the first 50 and second 50 columns in Tables 44 and 45 are 0.943 and 0.961, respectively. As they are greater than 0.7, the responses are considered consistent across the survey questions, and the data are reliable (Adamson & Prion, 2013).

Table 19: First 50 columns for the reliability test.

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardised Items	N of Items
.943	.943	50

Table 20: Second 50 columns for the reliability test.

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardised Items	N of Items
.961	.961	50

4.3.2. Validity test

The Kaiser-Meyer-Olkin (KMO) and Bartlett tests are used to evaluate the validity of data. The KMO values shown in Tables 21 and 22 are 0.700 and 0.795, indicating that the sample adequacy is good for factor analysis. Bartlett's p-value is 0.000, implying that the correlations between variables are significant, which is an indication that the data is suitable for factor analysis. KMO has a threshold value of 0.5, and Bartlett's p-value has a threshold value of 0.05.

Table 21: First 50 columns for KMO and Bartlett's Test.

KMO and Bartlett's Test

Kaiser–Meyer–Olkin Measure of Sampling Adequacy.		.700
Bartlett's Test of Sphericity	Approx. Chi-Square	3247.574
	df	1225
	Sig.	.000

Table 22: Second 50 columns for KMO and Bartlett's Test.

KMO and Bartlett's Test

Kaiser–Meyer–Olkin Measure of Sampling Adequacy.		.795
Bartlett's Test of Sphericity	Approx. Chi-Square	3499.261
	df	1225
	Sig.	.000

4.3.3. Comparative analysis

Comparative analysis involves the study of a group of variables in such a way that a trend can be observed. In this work, various factors are considered when developing a network visualization tool. For each factor, there are several survey questions, so the responses of the participants to the survey questions were grouped by factor, which facilitated the inferential analysis.

4.3.4. General factor

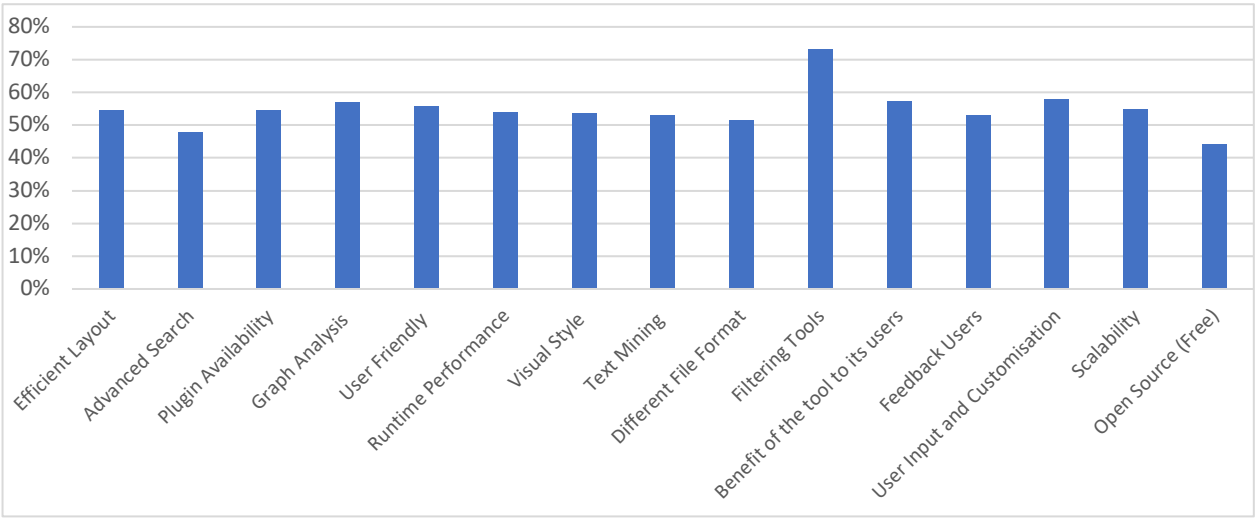


Figure 1. General factor

Figure 1 shows the responses of the survey participants regarding the importance of the general factors of the visualization tool. The highest rating is for the filtering tool (73%), which is thus the most important factor in a network visualization tool, followed by user input and customisation (58%), graph analysis and the benefit of the tools to its users (57% each), and user-friendliness and scalability (56% and 55%, respectively). Efficient layout, plugin availability, and runtime performance all had the same rating (54%), as did visual style, text mining, and user feedback (53%). The lowest participant ratings were for different file formats (51%), advanced search (48%), and open-source features (44%). Overall, 15 factors were rated in their order of importance. Filtering tools were the most significant, while advanced search and open source were the least important. The remaining factors can be considered significant since they were all rated average or a bit above average. This is not the calculated average of the response but just the average importance level of the factors.

Table 23:Standard deviations and mean values of general factors

Factors	Mean	Std. Dev	Factors	Mean	Std. Dev
Filtering Tools	3.88	.52	Text Mining	3.42	.56
Plugins	3.57	.59	User Input & Customisation	3.59	.64
Visual Styles	3.50	.54	Graph Analysis	3.54	.57
Advanced Search	3.44	.53	Feedback to Users	3.52	.64
Free and Open Source	3.41	.58	Strength	3.59	.62
Efficient Layout Algorithm	3.51	.56	Runtime Performance	3.50	.65
Scalability	3.55	.56	User Friendliness	3.57	.62
Different File Formats	3.51	.62			

In Table 23, the lowest mean value is 3.41, while the highest is 3.88. This shows that the ratings of the survey participants are positive on average. For the standard deviation, the highest value is 0.65, while the lowest value is 0.52, showing relatively low variation among the survey respondents. This reflects similarities in the participants' responses.

4.3.5. Heuristics factors

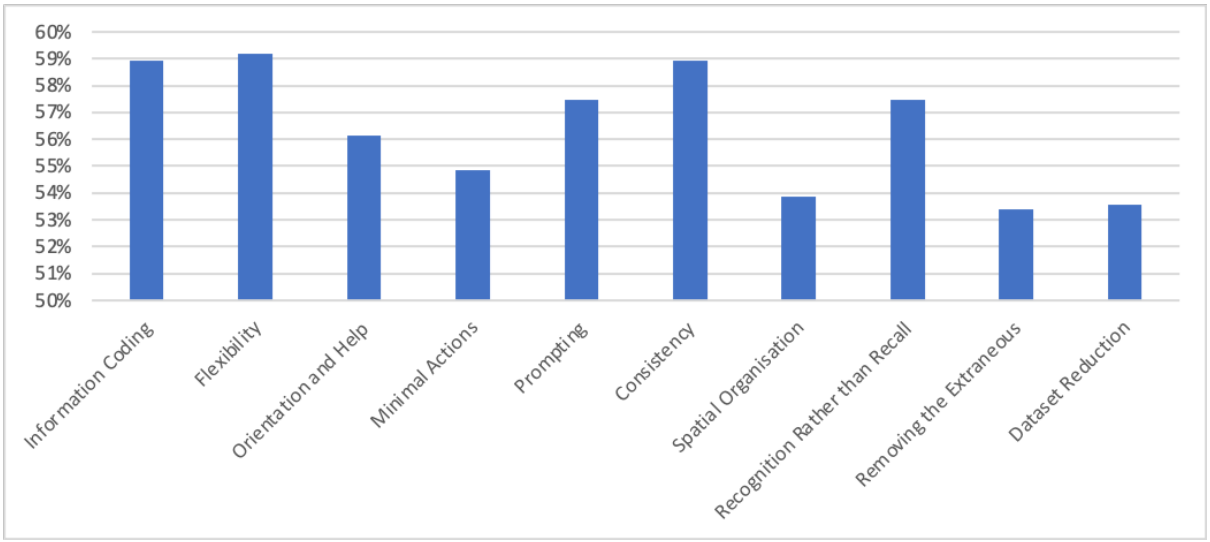


Figure 2: heuristic factors

Figure 2 shows the ratings of the survey participants on the heuristic factors of the visualization tool. Information coding, flexibility, and consistency are rated highest (59%, respectively), indicating that they are the most important factors. This is followed by prompting and recognition rather than recall (57%), orientation and help (56%), minimal action (55%), spatial organisation (54%), and dataset reduction (53%). Every heuristic factor is important since they are above average (50%), but a visualization tool should prioritise information coding, flexibility, and consistency.

Table 24:Standard deviations and mean values of heuristic factors

Factors	Mean	Std. Dev	Factors	Mean	Std. Dev
Information coding	3.61	.55	Consistency	3.57	.59
Flexibility	3.59	.61	Spatial Organisation	3.49	.56
Orientation and Help	3.53	.62	Recognition Rather than Recall	3.52	.67
Minimal Action	3.52	.61	Removing the Extraneous	3.49	.68
Prompting	3.56	.67	Dataset Reduction	3.49	.57

As shown in Table 24, the lowest mean factor value is 3.49, while the highest is 3.61. This indicates that the ratings of the survey participants are positive on average. Regarding standard deviations, the highest value is 0.68, while the lowest is 0.55, showing relatively low variation among the survey respondents. This reflects the similarity in the participants' responses.

4.4. Results and Discussion

Next, a summary of the findings is presented based on a critical analysis of the participants' responses. The participant's responses to the online survey were analysed using various techniques to scrutinise the information embedded in the data. The analysis started by checking the reliability and validity of the data. Cronbach's alpha, the KMO measure of sampling adequacy, and Bartlett's test of sphericity were conducted on the survey data. Based on the results of these tests, the data were considered reliable and valid, as the threshold values for each of the tests were achieved.

The participants' responses were further analysed using content analysis to calculate statistics regarding the participants' opinions on each question. This clearly indicated the number of participants who supported the inclusion of a visualization tool in a complex biological network and those who did not. To better understand the trend of the participant's responses, the percentages of similar levels of agreeability (e.g. 'agree' and 'strongly agree') were added. A specific condition was used to identify essential and non-essential factors that should be considered in a good network visualization application. For each of the survey questions, the factors with the highest percentage of 'agree' answers were deemed 'essential' factors, while those with the highest percentage of 'neutral' answers were categorised as 'non-essential.' None of the factors considered in this work had the highest percentage of 'disagree' or 'strongly disagree' responses. Speculatively, the aim of this research was to investigate existing visualization tools for complex biological networks, seeking to identify the important factors

that could improve the effectiveness of network visualization applications. Based on the results of this work, of the 25 factors considered, 24 were found to be highly essential/important for network visualization applications, whereas only 1 factor (advanced search) was found to be non-essential due to a large number of respondents saying it is not essential. Thus, all of the factors listed above should be considered when developing a network visualization application.

The findings of the quantitative analysis agree with those of the qualitative analysis. The interview participants agreed that filtering tools are an important feature of network visualization tools. This is consistent with the findings of the quantitative analysis, which identified a positive and significant relationship between filtering tools and plugins. The results also indicate that the plugin is important in graph visualization, but the complexity must be reduced to make it easier for users.

In addition, both analyses showed that network visualization users recognise the importance of visual styles in the tool. Visual style was shown to be positively correlated with other variables in the quantitative analysis, while the interview participants stated that it is one of the key factors they consider before choosing a network visualization tool. Furthermore, the advanced search is considered unimportant in the survey and interview. In the quantitative analysis, the result was lower than average (50%), and in the qualitative analysis, users disagreed with the usefulness of some advanced search features and said some of them should be excluded from network visualization tools, such as considering regular expressions in advanced search tools. Both analyses found that advanced search was not an important factor to consider when selecting a network visualization tool. However, it may be improved to include other features, such as searching for a group of nodes and the availability of searching webpages.

In related findings, the interviewees felt that factors such as open source, efficient algorithm layout, and scalability are crucial to ensuring the better use of network visualization tools. Regarding the heuristic features, the respondents and interview participants stated that information coding is an essential foundation for an effective visualization tool. They also indicated that flexibility, orientation help, and minimal actions are features they consider before choosing a visualization tool. This shows that for the visualization tool to perform effectively, many of the general factors and heuristic factors must be intact.

The importance of performance is further emphasized in Chapter 4, where it is highlighted as a critical aspect of network visualization tools. Understanding that performance directly

influences the usability and effectiveness of these tools, the study focuses on this aspect in Chapter 5.

To test the significance of these factors, the next chapter assesses the performance of some selected visualization tools: Medusa, Cytoscape, Arena3D, Graphia, and Gephi. By evaluating these tools, we aim to provide a clearer understanding of how well they meet the outlined criteria and their overall effectiveness in practical applications.

The methodologies applied in this research appeared to have worked well with the data. However, further investigation of the essential factors in developing visualization tools for a complex biological network is needed. The factors could be further broken down into pieces to identify the specific elements of the factors that are essential and should be included in such tools. For instance, plugins have many features. Some participants condemned a plugin initialising automatically, while others supported such a feature.

In future research, it is strongly recommended that the features of the factors considered in this research are investigated instead of the factors themselves. It is highly possible that this would disqualify some of the factors considered essential in this research, as analysing the participant's responses to the features of the factors would reveal more detailed insight into their importance for complex biological network visualization tools.

4.5. Summary

This section evaluated the factors based on the experts' responses. Five participants were interviewed and responded to questions based on 25 factors, including general and heuristic factors. Their responses were analysed using a thematic analytical technique. In addition, a survey was administered online to participants from different categories related to the subject matter. The responses from the interview were compared with those from the survey, and it was found that filtering tools are a crucial feature in network visualization. This is consistent with the findings of the quantitative analysis, which found a positive and significant relationship between filtering tools and plugins. The results also indicate that plugins are important in graph visualization, but their complexity must be reduced to make it easier for users.

The findings of this chapter not only contribute to the field of network visualization research by providing a broad framework to consider when developing and evaluating network visualization tools but also make practical recommendations to improve their usability and functionality.

Chapter 5: Evaluation of the Visualization Tools

This chapter discusses the evaluation of the visualization tools, as they are necessary for investigating biological networks. In Chapter 5, the visualization capabilities of five tools are tested using datasets of different sizes. It also evaluates the layouts available in the visualization tools and whether they are user-friendly when working with large and small datasets. This knowledge is used to capture the layouts used for the tool developed by the researcher. Visualization tools like Medusa, Cytoscape, Arena3D, Graphia, and Gephi are chosen for review simply because they have been widely used and recommended in the literature. Thus, this chapter will describe the datasets, the layouts evaluated, and a summary of the visualization tools.

5.1. Dataset

Each visualization tool was used to visualize the gene-disease association dataset, such that the size of the dataset increased from 2,000 to 10,000 in steps of 2,000 edges. The dataset has 16 columns and 84,038 rows. The columns are defined as follows:

geneId -> Gene Identifier
geneSymbol -> Official Gene Symbol
DSI -> The Disease Specificity Index for the gene
DPI -> The Disease Pleiotropy Index for the gene
diseased -> UMLS concept unique identifier
disease name -> Name of the disease
disease type -> The DisGeNET disease type: disease, phenotype, and group
diseaseClass -> The disease class(es).
diseaseSemanticType -> The semantic type(s) of the disease
score -> DisGENET score for the gene-disease association.
EI -> The Evidence Index for the gene-disease association.
YearInitial -> First time that the gene-disease association was reported.
YearFinal -> Last time that the gene-disease association was reported.
Nomeids -> a Total number of publications reporting the gene-disease association.
NofSnps -> Total number of SNPs associated with the gene-disease association.
Source -> Original source reporting the gene-disease association.

All the columns, apart from geneId, geneSymbol, diseaseId, diseaseName, diseaseType, and

diseaseSemanticType, were used in visualising the gene-disease association. The datasets can be accessed via:

https://www.disgenet.org/dbinfo#:~:text=The%20DisGeNET%20database%20integrates%20information,Mendelian%2C%20complex%20and%20environmental%20diseases.

5.2. Layouts

Layouts are the way in which elements like nodes and edges are arranged in visualizations or on a two-dimensional plane (Gibson et al., 2013). They can also be referred to as the spatial organisation and positioning of nodes and edges to relay information properly. There are different types of layouts, the most common of which are random, circular, and grid layouts. A brief discussion about the layouts is provided below.

5.2.1. Random Layout

This type of layout is used for arranging elements or nodes in a graph or diagram without following any predefined structure or pattern (Hearst and Rosner, 2008). It positions elements at random and with predetermined features. It is commonly used to represent a relationship between elements that are not defined explicitly. The goal of a random layout is to create an organic but visually pleasing arrangement. Thus, it is useful for exploring large and complex datasets in which the underlying structure is not immediately known. Examples of visualization tools using random layouts include Medusa and Gephi.

5.2.2. Circular layout

A circular layout is a typical graph layout that arranges the graph's nodes in a circle (Gansner and Koren, 2006). The nodes are usually placed around a central point, while the edges between them are drawn as straight lines. This layout is used mostly for the visualization of hierarchical data. This type of layout seeks to remove centralised nodes by placing all the nodes in a circle. Examples of visualization tools that use it include Medusa, Cytoscape, and Gephi.

5.2.3. Grid

A grid layout is a fundamental concept in data visualization, providing a structured arrangement of elements within a visualization (Dayama et al., 2020). It involves organising data into a grid-like structure, typically consisting of rows and columns, to present information clearly and in order. This type of layout is commonly used in tables, charts, graphs, and dashboards. Tools that use grid layouts include Medusa and Cytoscape.

5.2.4. Hierarchy

In this type of layout, nodes are represented in the order of their number of connections, such that the nodes with a high number of connections are placed at the top, while those with a low number are placed lower down (Schulz et al., 2010). Examples of tools using hierarchy include Medusa and Cytoscape.

5.2.5. Fruchterman–Reingold

This is a force-directed graph-drawing algorithm that positions nodes in 2D spaces based on their level of connections (Muelder, 2011). It is used to create diagrams of complex networks that are aesthetically pleasing and informative. Its algorithm works through the iterative application of a force-based mode to the nodes in the graph, with each node exerting an attractive force on its connected nodes and repulsive forces on the other sides. Examples of tools using this technique include Medusa and Gephi.

5.2.6. K-means

K-means clustering is an unsupervised learning algorithm that partitions a set of data into k groups known as clusters (Tavallali et al., 2021). A powerful tool for analysing complex biological data when integrated with network visualization, it helps uncover pattern functional relationships. The number of clusters is specified in advance. The algorithm works by iteratively assigning data points to clusters and then updating the cluster centroids. This process is repeated until the cluster centroids can no longer change. Medusa is a tool that uses K-means clustering.

5.2.7. Stack layout

This layout type provides a visual representation of a call stack, which is a data structure that stores information on the active function calls in a particular program (Gralka et al., 2017). It shows the function calls sequentially, starting from the main function and going through the nested function calls. Each function call is represented in a rectangular box, with the function name and other needed information shown inside. It organises data into interconnected layers, thereby providing clarity and focus making it easier to interpret interactions within the network. An example of a tool using stack layouts is Cytoscape.

5.2.8. Attribute layout

An attribute layout plays a significant role in organising and presenting data effectively (Qin et al., 2020). It refers to the arrangement of nodes and edges in a network graph based on some

nodes' characteristics, like gene expression levels and interaction types. It effectively shows the patterns inherent in biological data. Cytoscape uses this type of layout.

5.2.9. Degree-sorted layout

Degree-sorted layouts are used in visualization tools to arrange nodes in a graph based on their degree, that is, the number of connections they have to other nodes (Nishida et al., 2023). This layout algorithm aims to reduce the number of edge crossings and improve the overall readability and clarity of graphs. In a degree-sorted layout, nodes with a higher degree are placed closer to the centre of the graph, while nodes with a lower degree are positioned further away.

5.2.10. ForceAtlas layout

This is a force-directed layout algorithm that uses network visualization (Cheong and Si, 2020). It is designed to show the underlying structure of a network by positioning the nodes and edges in a way that shows their relationships and interactions (Cheong and Si, 2020). It works through the simulation of physical forces that act on the nodes if connected through springs. The algorithm iteratively calculates the forces between nodes and alters their positions until equilibrium is achieved. Gephi uses this tool.

5.2.11. Markov clustering

This unsupervised machine-learning algorithm uses clustering nodes in a graph (Blumenthal et al., 2020). It is based on the idea that nodes are strongly linked to each other and are more likely to be in the same cluster. Iteratively, it works by updating a matrix with similarities between nodes, which is calculated based on the number of paths connecting them. Graphia is a visualization tool that uses this technique.

5.2.12. Edge reduction

This technique is used to simplify or minimise the number of edges shown in a graph or network visualization (Van Den Brand et al., 20203). It is mainly useful for dealing with complex networks, where the volume of connections could result in cluttered visualizations. Graphia uses this type of layout. Other layouts using this technique include Spectral clustering, Affinity propagation, Parallel geometry, Group-attribute layout, Edge weighted-spring, Embedded layout, ForceAtlas 2, and Vifan HU metrics and structures.

5.3. Visualization Tool 1: Medusa

Medusa represents a network of nodes in 2 dimensions (Zhou et al., 2023). It expects network data to be in CSV, TSV, or txt formats such that the nodes definition follows a line with the

string “*nodes” while the connection (edges) definition follows a line with the string “*edges”. Medusa is a standalone application implemented in Java. Chosen because it supports random, circular, grid, Fruchterman, and Hierarchical layout algorithms. It has a simple user interface and supports pre-defined clustering by placing pre-defined clusters in the network in random positions on the screen. It allows the usage of IDs for nodes with optional annotation of nodes.

For our working dataset, nodes are geneId and diseaseId with annotation of geneSymbol and diseaseName, respectively. Since Medusa supports clustering by only one variable, the network was clustered by the “diseaseSemanticType” column because it has a more fine-grained number of clusters (28 semantic types of disease) than “diseaseType”, which has only 3 unique groups.

5.3.1. Visualization results for different datasets

An in-depth discussion of the visualization results for different datasets of different sizes is provided below. Using Medusa for data visualization, From the first 2000 rows, all connections originate from a centre cluster of the genes to disease nodes. However, as the size increased to 4000, Medusa randomly placed the gene cluster on the left-hand side of the screen. Since the data size has increased, more clusters are forming a closed shape. Furthermore, as the dataset increased to 6000, the connections became denser, and the gene cluster was at the top right side of the screen. And as such, more disease clusters, that is, about 13 of them, formed a closed shape. As expected, the network connection becomes denser as the size increases to 8000, but the number of closed-shaped disease clusters remains at about 13. Finally, for the first 10,000 gene-disease associations in the data, the connection is denser; as such, there are now about 15 closed-shape clusters.

The average time taken to load the different data into Medusa.

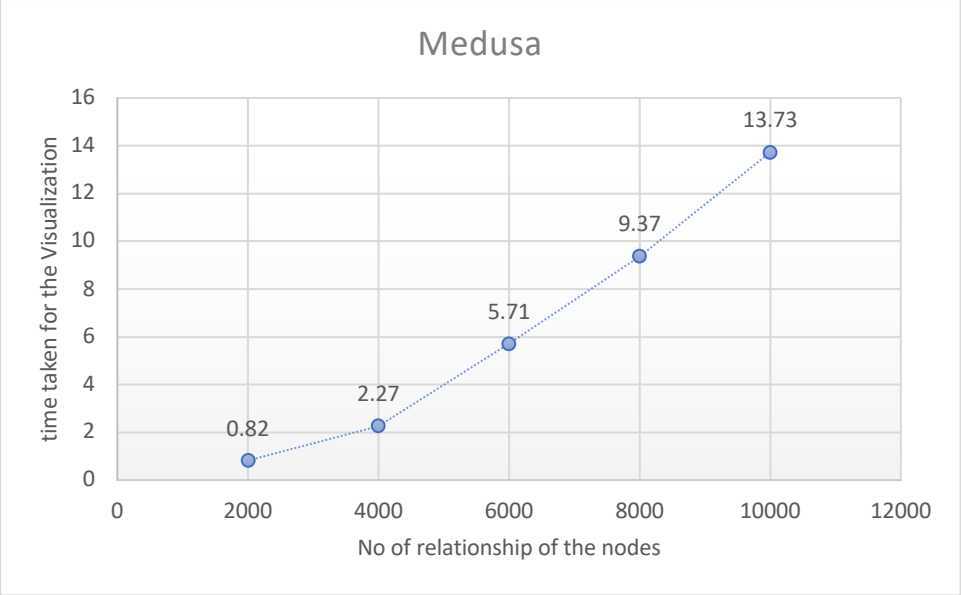


Figure 3: Time taken to load data into Medusa.

In Figure 3, we see an approximate linear relationship between the size of the data and the time taken to load the data into Medusa. In general, the time taken to load the data increases with the size of the data.

5.3.2. Layouts

Medusa supports several layouts, including random, circular, grid, Fruchterman, and hierarchical layouts. The larger size of all these layouts is presented in Appendix H.

Random Layout

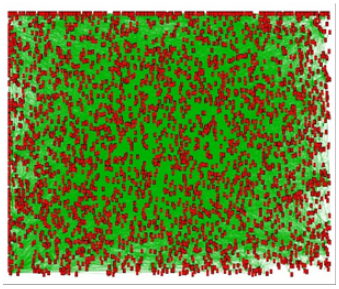


Figure 4. Random layout

Random layout positions the network's nodes and connects them at random based on their relationship with the loaded data. One of its advantages is that it does not take much time to compute. However, it has some drawbacks, including poor readability. Additionally, it is not easy to track a specific connection in the network because of its highly dense edges.

Circular Layout

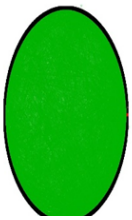
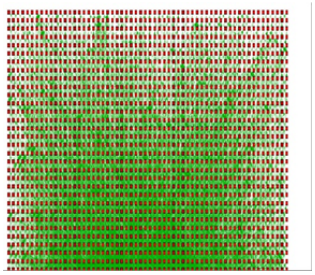


Figure 5: Circular layout

Here, the intention is to remove users' attention from centralised nodes by encircling all the nodes in a circumference. This circular pattern enhances the relationship among visualization. However, a significant limitation is that the edges are highly dense due to many edge interceptions, thus reducing the readability of a particular connection.

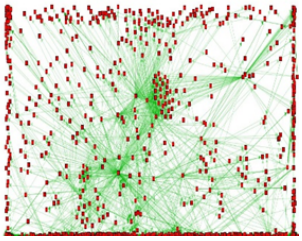
Grid Layout



The nodes in this layout are presented in a grid format. Its layout neutrality encourages the removal of users' attention from the central node, as the nodes on the lowest part of the grid have more connections than those at the top. However, it has difficulty effectively representing clusters in the network.

Figure 6: Grid layout

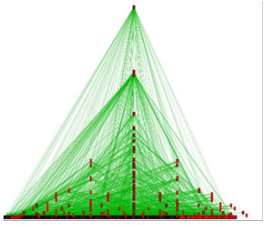
Fruchterman–Reingold



This is a force-directed algorithm used to represent graphs in a more aesthetically appealing way. One of its strengths is that all the edges are equal in length, which minimises edge interception. Hence, highly connected nodes are easily recognised. However, the time complexity is relatively high.

Figure 7:Fruchterman layout

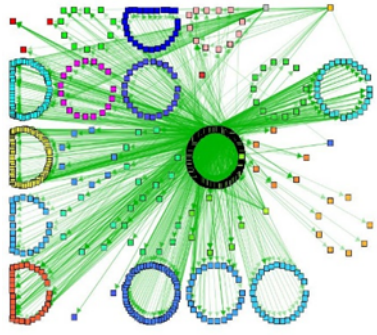
Hierarchical Layout



This layout is used to represent nodes in order of their number of connections. Nodes with a high number of connections are placed at the bottom, while those with a low number are placed at the top. This produces highly readable and intuitive visualizations. However, it is not well suited for all data types.

Figure 8:Hierarchical layout

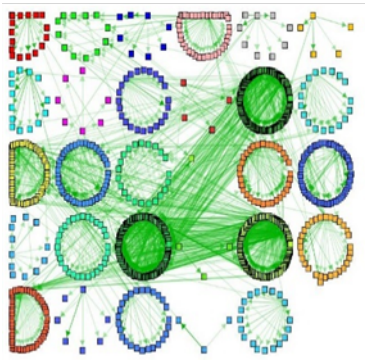
K-means



K-means is an algorithm that is used to cluster data into k number of clusters, with each cluster represented by its centroid. The position of these centroids is calculated so that they are as far away from each other as possible, and for nodes belonging to a given cluster, the centroid of the cluster is the nearest centroid to them. As shown in Figure 9, Medusa's k-means rendering of the data grouped all the gene nodes as a cluster as expected.

Figure 9:K-means

Spectral Clustering



Spectral clustering is a supervised clustering approach similar to k-means in which the user must define the number of clusters. A weighted undirected G graph is partitioned into a collection of discrete groups via spectral clustering. As shown in Figure 10, genes with many associated diseases are formed into a cluster.

Figure 10:Spectral clustering.

5.3.3. Summary

Medusa's rendition of datasets of different sizes has some similarities in the way it displays these datasets. Different types of data can be clustered based on the semantic disease type. However, we observed some differences as the amount of data increases:

1. Increase in the connection density of the network.
2. Increase in the number of clusters.
3. Increase in the amount of time taken to load the data.

A downside of Medusa is that it does not label the names of the clusters in the visualization. In addition, it cannot properly render layers in a graph, and connection lines intercept, making it difficult to trace a particular connection without having to use filtration tools.

5.4. Visualization Tool 2: Cytoscape

This free software application visualises a network of molecular interactions and biological pathways and integrates them alongside other factors, including annotations, gene expression levels, and other state data (Miryala et al., 2018). While it is rooted in biological research, this programme, over time, has developed into a generic platform for visualization and network analysis. One of the major features of Cytoscape is that it is a foundation tool for data integration, analysis, and visualization. Based on this, Cytoscape needs no special data format, although it supports traditional input data types, such as CSV, databases, National Data

Exchange Program (NDex), Tab Separated Values (TSV), and Uniform Resource Locator (URL).

5.4.1. Visualization results for different datasets

Cytoscape's visualization of the first 2,000 rows uses a default (perfuse force-directed layout) layout. The resulting diagrams frequently demonstrate a graph's inherent symmetric and clustered structure and a well-balanced distribution of nodes with only a few edge intersections. A data size of 4,000 results in a dense, lone cluster of gene-disease associations. There are some clusters connected to but extended from the dense cluster of genes and diseases. However, with a data size of 6,000, the network becomes denser, and Cytoscape captures more one-to-one gene-disease associations. With a data size of 8,000, there is a similarity with 6,000-row data, along with a reduction in the number of lone gene-disease associations and an increase in the density of the dense middle cluster of genes and diseases. Finally, the network diagram for a data size of 10,000 contains a greater number of lone clusters of gene-disease associations. Figures 11-15 show the visualization results for different datasets. The visualization indicates that red is the highest DisGENET Score, and blue represents the lowest DisGENET Score.

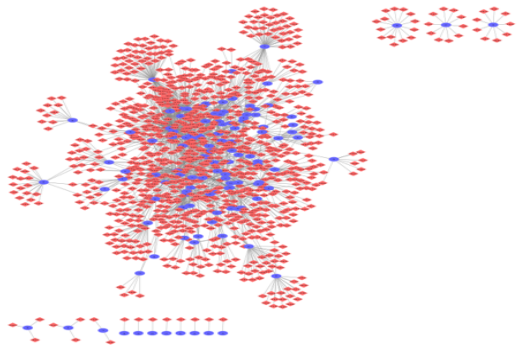


Figure 11: Data size 2000

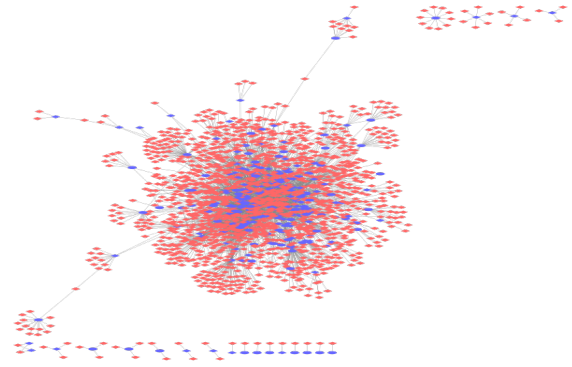


Figure 12: Data size 4000

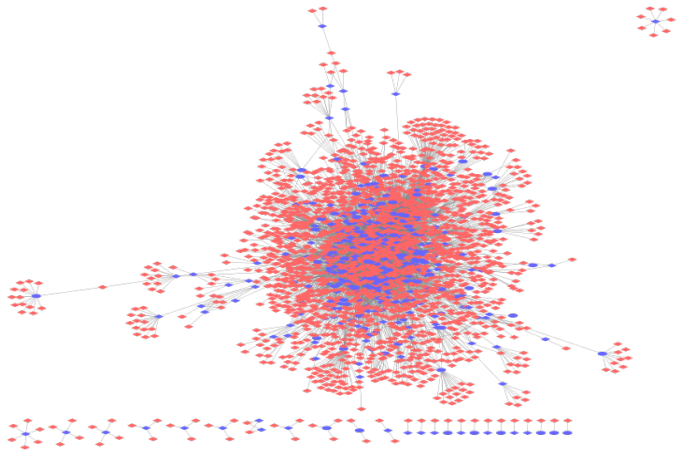


Figure 13: Data size 6000

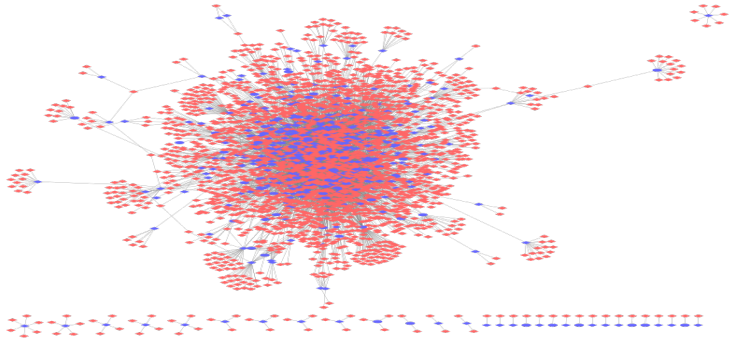


Figure 14:Data Size 8000

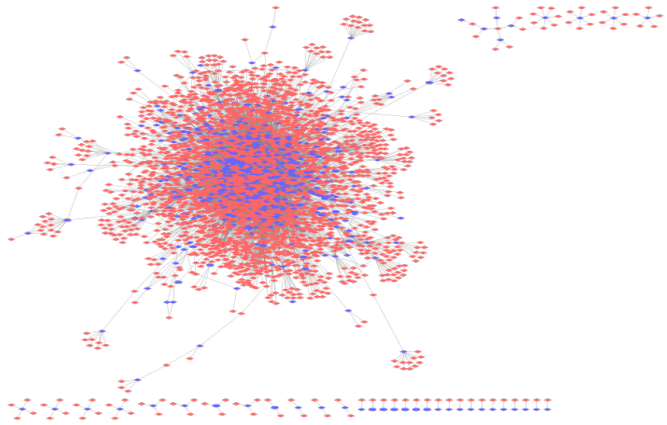


Figure 15:Data Size 10000

Time is taken to apply to perfuse force-directed layout.

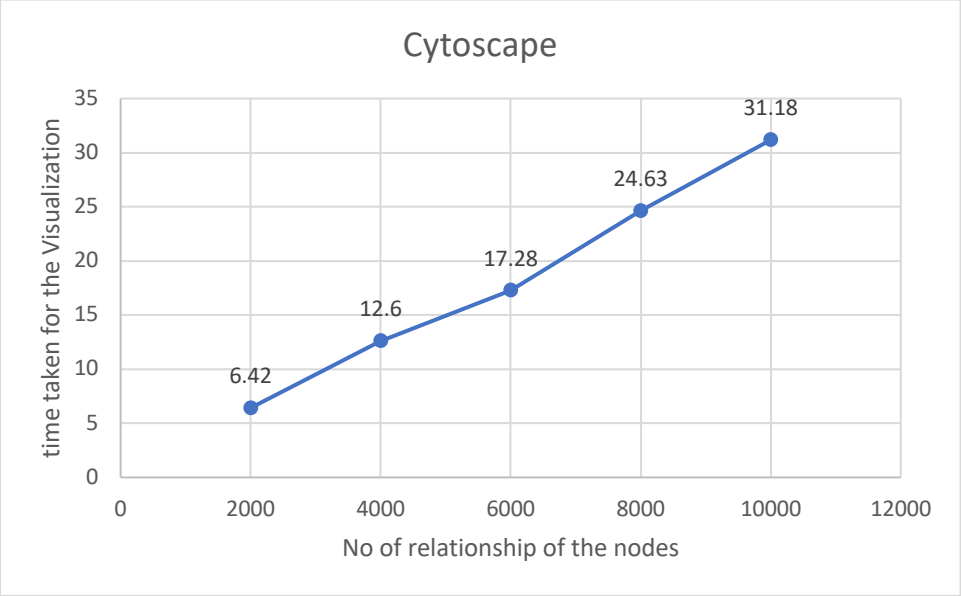


Figure 16:The average time taken to apply to perfuse force-directed layout.

In Figure 16, we see an approximately linear relationship between the size of the data and the millisecond time taken to run the default perfuse force-directed layout algorithm in Cytoscape. The time taken to perform the force-directed layout increases with the data size. Applying a force-directed layout in Cytoscape involves the use of a graph layout algorithm to arrange nodes and edges based on the force between them.

5.4.2. Cytoscape layouts

Grid Layout

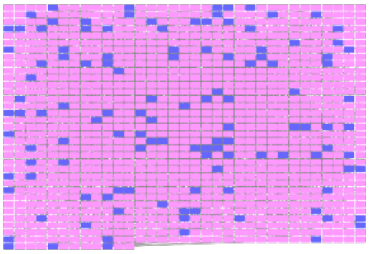


Figure 17:Grid layout

The nodes in this layout are presented in a grid format. Its layout neutrality is beneficial, as it encourages users to remove their attention from the central node, with nodes at the lowest point on the grid having more connections than those at the top. However, this layout has difficulty effectively representing clusters in the network.

Hierarchical Layout

Figure 18:Hierarchical layout.

This layout represents nodes in order of their number of connections. Nodes with a high number of connections are placed at the top, while those with a low number are placed at the bottom. This results in highly readable and intuitive visualizations. However, readability is low for large and complex networks, as shown in Figure 18.

Circular Layout

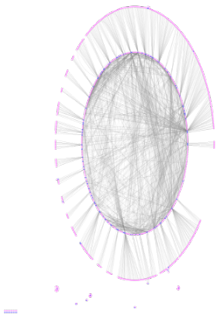


Figure 19:Circular layout

The circular layout generates network visualizations as group and tree structures, dividing the network into segments based on nodes' connections and then organises these segments as distinct segments in a radial tree pattern. However, in Figure 19, we can see a circular network with some faint, smaller lone networks of genes and diseases, reducing its readability. The result of zooming into the big circular network is seen in Figure 19.

Stack Layout

Figure 20:Stack layout

This layout organises the nodes by stacking them vertically and using curves and lines to connect source and target nodes. The disadvantage of this approach is that the layout is not visually appealing for large networks.

Attribute Layout

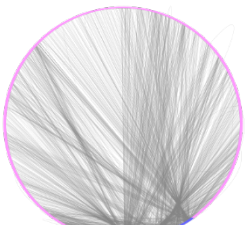


Figure 21:Attribute layout

This is a fast and useful layout that places all the network nodes in a circle, which is beneficial when working with small networks. Consequently, all nodes with similar values in the column are categorised around the circle. Figure 21 presents shows the big circle, represented by

blue nodes. However, not all nodes in the data can fit into the circle, and the excess nodes and their connections are placed outside the circle.

Degree-Sorted Layout

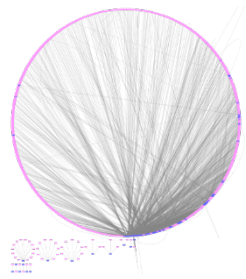


Figure 22:Degree-sorted layout

This layout places nodes on the circumference of a circle such that the nodes are sorted in order of their degrees, that is, the number of connections the node has. Figure 22 shows the results of applying the degree-sorted layout to our data. As seen in the network visualization, the nodes are sorted in order of their degrees, starting from 6 o'clock and proceeding in an anticlockwise direction. It was also observed that genes have the highest degrees, as they are many from the starting position of 6 o'clock. Like the attribute circle layout, excess nodes and their connections are placed outside of the circle.

Group Attribute Layout

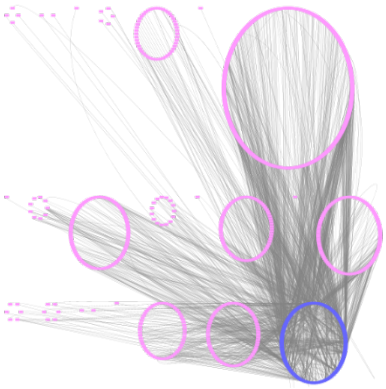


Figure 23:Group attribute layout

This type of layout makes use of a pre-defined cluster to group nodes, as with the attribute circle layout. However, unlike the attribute circle layout, any set of nodes with similar interesting column values is categorised into different circles as opposed to a single circle with all the data nodes. As shown in Figure 23, more than 20 groups of disease nodes were visualised after categorising the disease semantic types.

5.4.3. Summary

Cytoscape is a powerful network visualization tool supporting several layouts, including perfuse force-directed, circular, grid, group attribute, and degree-sorted layouts. It does not require a special data format but supports conventional data formats, including NDex, CSV, TSV, URL, and databases. Its interface allows users to select source, target, and attributes from the columns of loaded data. In addition, the time taken to load and run the default perfuse force-directed layout is proportional to the data size.

5.5. Visualization Tool 3: Arena 3D

This standalone Java application focuses on 3D graph presentations. It utilises a multi-layer concept that analyses big networks in 3D so that the different layers in the representation may be the same as different data types (Zhou and Xia, 2018). When clustering, we adopt the disease type (DisGNet disease type) and semantic disease type, that is, the ULMS disease type. The clustering process is performed for the disease type and semantic disease type layers for various amounts of data. The data sizes range from the initial 2,000 rows to 10,000 rows, with a step size of 2,000.

The Arena 3D graph for datasets of different sizes is similar. The time taken to load the network data for the different sizes of data is shown in Figure 24.

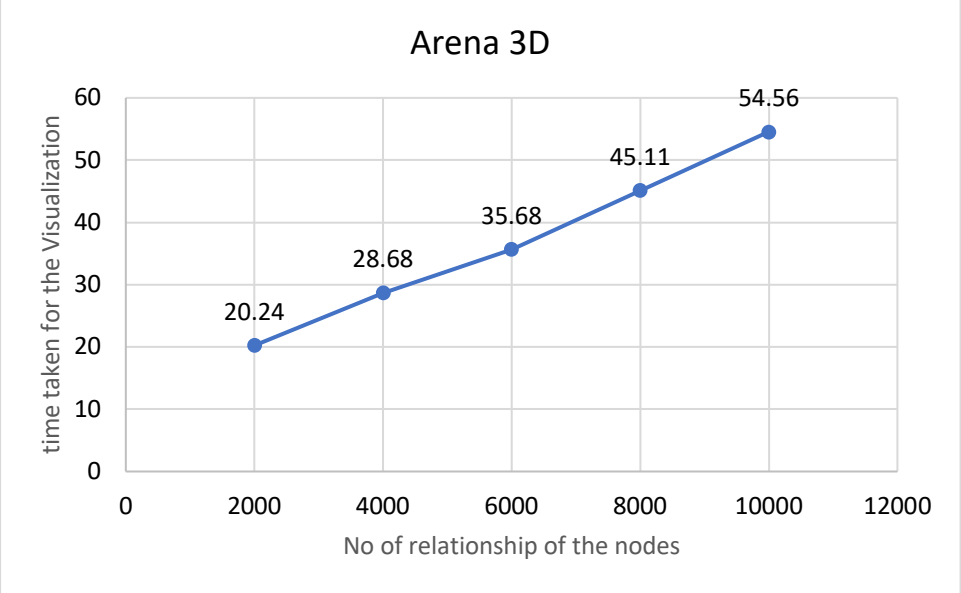


Figure 24:Arena 3D graph dataset sizes

Therefore, we can see that as the data size increases, the time taken to load the data also increases per second.

5.5.1. Layout

Arena 3D employs a multi-layer approach for graph rendering. The multi-layer rendering of the working network data uses gene, disease Type (DisGNet disease type), and disease

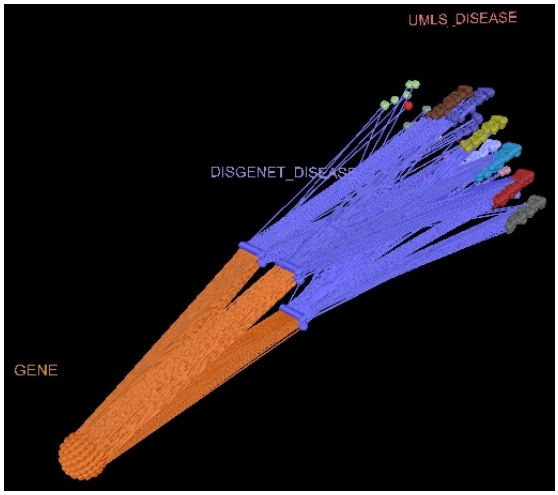


Figure 25:Multi-layer concept

Semantic Type (ULMS disease type) from the layer's working data. In contrast, the clustering of the disease type and semantic disease type layers is shown as follows.

From Figure 25, we see three layers, namely, 'GENE', 'DISGNET_DISEASE', and 'UMLS DISEASE'. These layers are represented by the endpoints of the straight lines (edges) in the figure. These are the three disease types in the data. The orange lines in the graph represent the mapping of certain genes to the disease type. Each disease type is further mapped onto its corresponding semantic disease, which is clustered into 28 groups.

5.5.2. Summary

Arena 3D is a standalone Java application that visualises graphs using a multi-layer layout. It expects data to be in a certain format, as presented above. It can make use of predefined clusters to cluster nodes within a layer, has a simple UI, and is optimised for visualising complex networks.

5.6. Visualization Tool 4: Gephi

Gephi is an open-source graph visualization programme that uses a 3D rendering engine to show graphs in real-time, allowing quick exploration. It accepts different formats, such as CSV, databases, and spreadsheets, as well as edge and node information in separate files in different formats. Edge data are expected to have 'source', 'target', and 'weight' columns for source node and target node identifications and connection strength, respectively. Conversely, the node data should include 'id' and 'label' in its columns for the unique nodes and respective labels. When varying the size of the data from 2,000 to 10,000 connections with a step size of 2,000, data loading is very fast (less than 2 seconds) and virtually the same for the different data sizes.

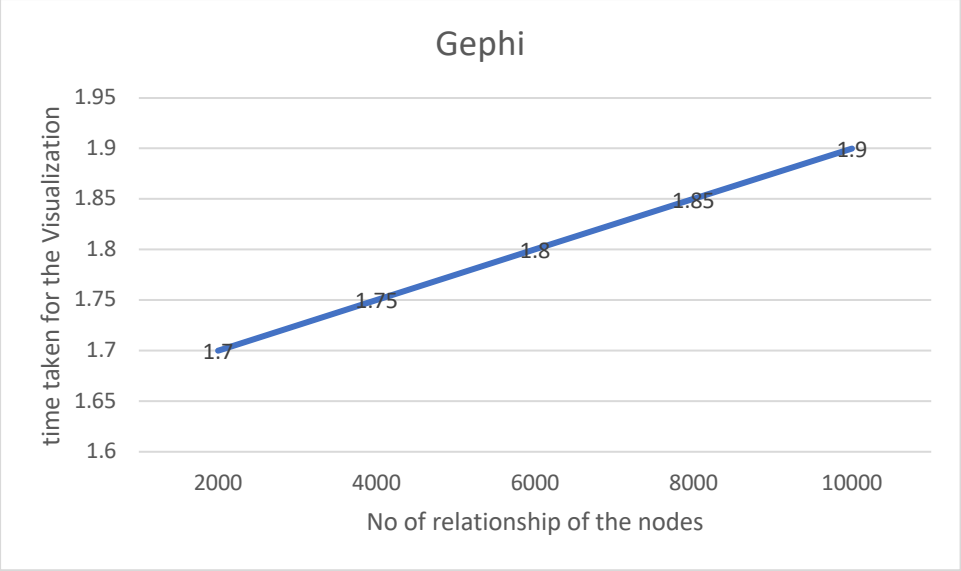
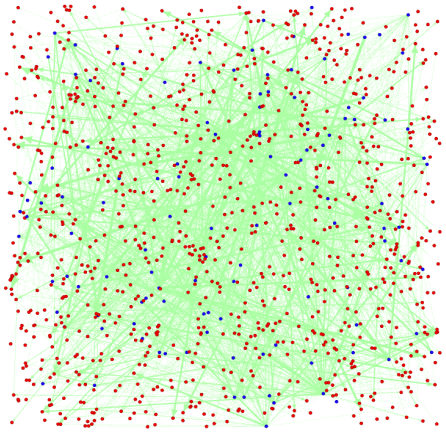


Figure 26:Gephi graph dataset sizes.

5.6.1. Layouts

Layouts refer to the spatial organisation and positioning of nodes and edges to relay information properly.

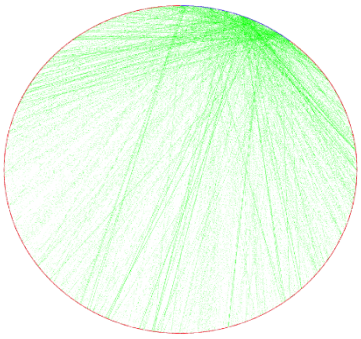
Random Layout



The random layout is the default layout in Gephi. It is a 2D rendering of the network in which the nodes and their corresponding edges are randomly placed in the graph. It is quick to implement, but it does not often show any patterns in the network.

Figure 27:Random layout

Circular Layout



The circular layout is fast and places all the nodes in the network around a circle, which is especially useful when working with small networks. A user-selected node column determines the node order. All nodes with the same column value are grouped around the circle. This circular pattern enhances visualization relationships, but it reduces the readability of individual connections.

Figure 28:Circular layout

ForceAtlas Layout

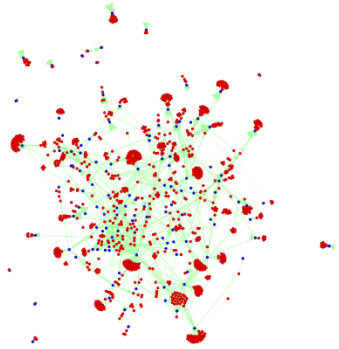


Figure 29:ForceAtlas layout

The ForceAtlas layout, also known as scale-free or small-world layout, is a quality-oriented network layout algorithm. In Figure 29, nodes with low degrees are far away from the centre of the graph, whereas those with high degrees are close to the centre of the graph. ForceAtlas is useful for obtaining quick insights into small networks, as it shows nodes with high degrees.

Fruchterman–Reingold

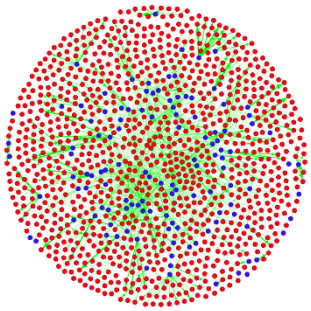


Figure 30:Fruchterman–Reingold layout

This layout represents the graph as a system of mass particles, with nodes and edges being mass and spring particles, respectively. The algorithm's objective function is optimised by minimising the energy of the physical system. This algorithm is often used in graph visualization but has the downside of a slow operating speed.

Yifan HU

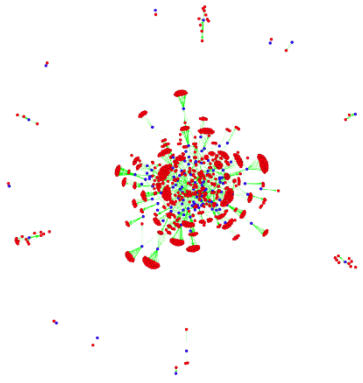


Figure 31:Yifan HU layout

This is a fast algorithm that can be used to visualise large graphs. It reduces complexity through the integration of a force-directed model and a graph-coarsening method. The negative impacts on one node from a cluster of distant nodes are estimated by a Barnes-Hut calculation that treats them as a single super-node.

As shown in Figure 31, the nodes with lower degrees are pulled farther from the centre of the graph than those with higher connections. Disease nodes (Red nodes) are repelled by their gene nodes, thus forming an umbrella and pulling on their gene nodes.

5.6.2. Summary

Gephi is an open-source network graph visualization programme that uses a 3D rendering engine to display graphs in real-time, allowing quick exploration. Its major strength is that it is built on a multi-task architecture that supports multi-core processors. It also supports different file formats (e.g. CSV, spreadsheets, databases). Data loading is very fast, and it supports layout algorithms, including Yifan Hu, Fruchterman–Reingold, and ForceAtlas.

5.7. Visualization Tool 5: Graphia

Graphia is an open-source graph-visualising application that can handle large and complex graphs. It supports adjacency matrix, CSV, GraphML, and TSV files. It may be used for visualising graphs with a large number of nodes and edges in both 2D and 3D (Majeed et al., 2020). The tool provides high-quality rendering, which can be performed on a standalone graphics card. For instance, Figure 32 presents the edges and disease nodes associated with the gene node with geneID 'ABCD1' when zoomed in.

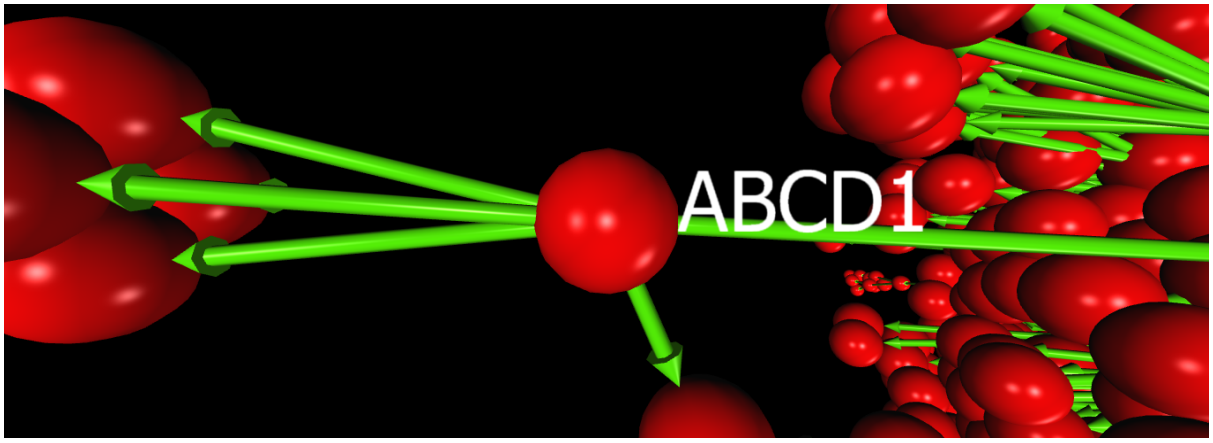


Figure 32:Visualising data in Graphia

Datasets load into Graphia very quickly. We found no significant difference in the time required to load different sizes of datasets into Graphia. In addition, Graphia organises algorithms in the transform tab (or window), which contains various functions that can be applied to graphs.

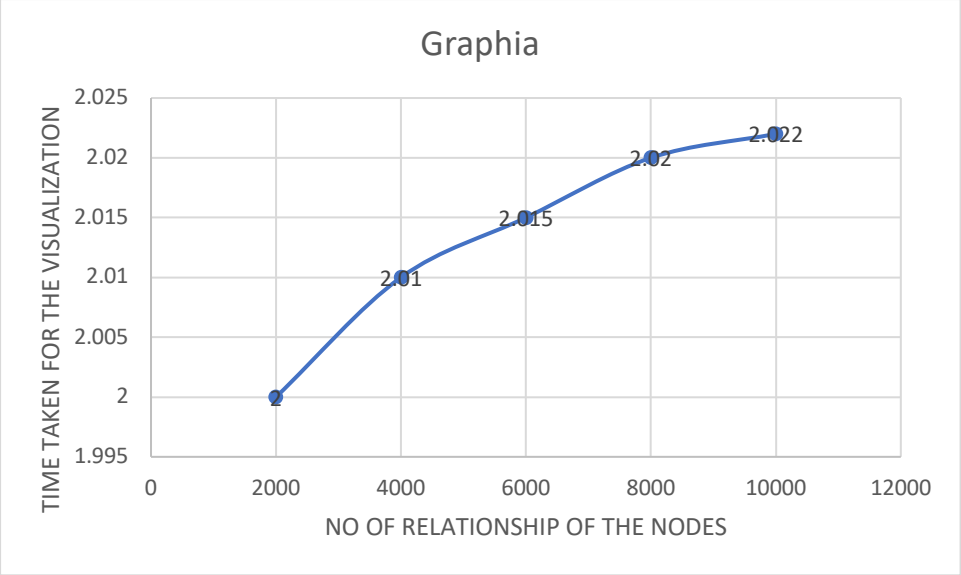


Figure 33:Graphia graph dataset sizes.

5.7.1. Layouts

Edge Reduction

Visualising graphs with a high number of edges is often difficult. Many of these edges may obscure higher-level clusters or patterns and may not contribute to the overall structure of the graph. K-nearest neighbours (k-NN), percentage nearest neighbours (%-NN), and edge reduction are techniques for pruning graphs in Graphia (Figures 34–36).

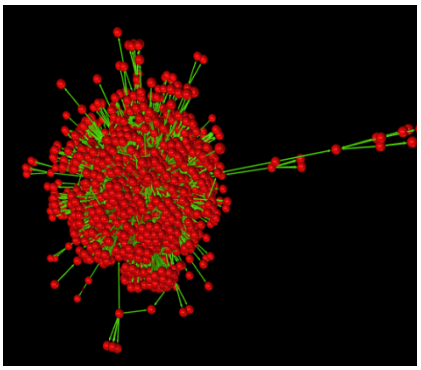


Figure 34:k-NN

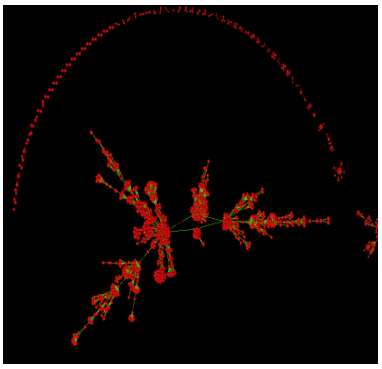


Figure 35:%-NN

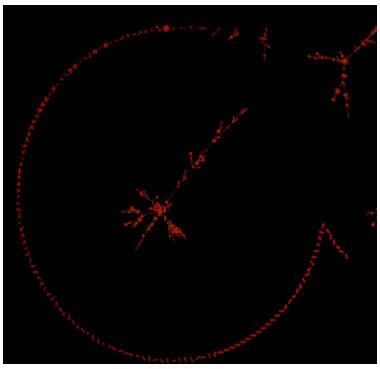


Figure 36:Edge Reduction

Metrics

Graphia uses the following metrics for analysing a graph structure:

Betweenness: This is a centrality measurement that is based on the shortest paths between nodes. It is the number of these shortest paths that pass through the node.

Eccentricity: This is a measure of a node's global position in the graph. Graphia does this by calculating the shortest path between every node and then assigning the longest path length found for each node.

Page Rank: This is used to obtain an estimate of how important a node is by counting the number and quality of links to that node. Its name is because it was originally used to measure the importance of a web page.

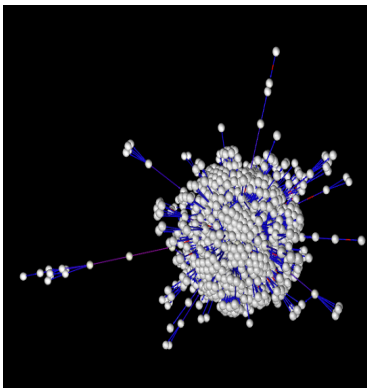


Figure 37:Betweenness

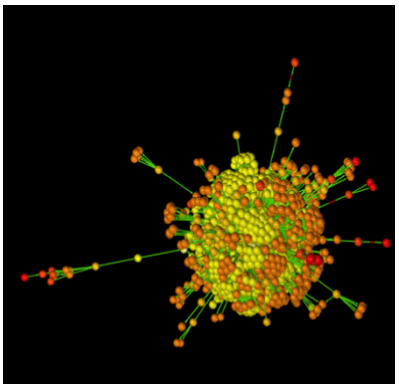


Figure 38:Eccentricity

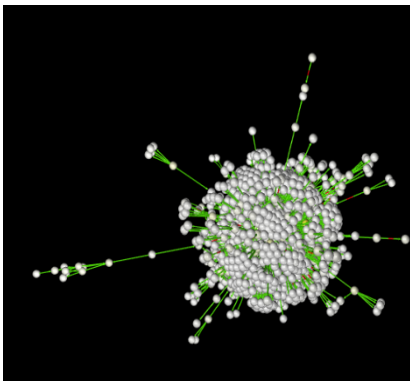


Figure 39:Page Rank

Structures

Graphia can alter the structures of a graph using the following tools:

Remove Leaf: This pruning algorithm is used to remove leaf nodes from a graph to make it easier to see patterns in a graph.

Remove Branches: Similar to removing leaf nodes, the removal of branches utilises a pruning algorithm, but unlike removing leaf nodes, it prunes a graph by removing less important branches of nodes.

Spanning Tree: This is a subgraph (a tree) that includes all of the nodes of the super-graph. In general, a graph can have multiple spanning trees. However, a graph that does not have any connections cannot have a spanning tree.

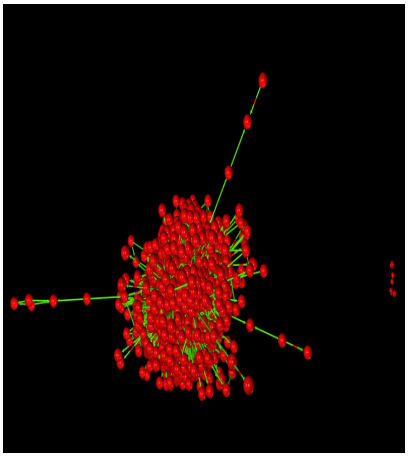


Figure 40: Remove the leaf

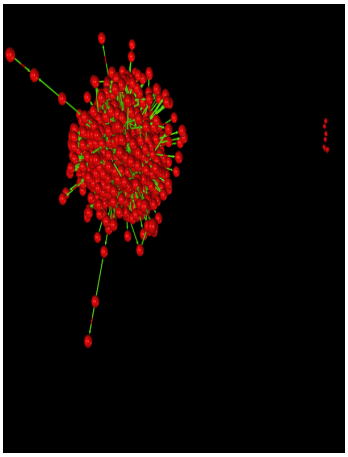


Figure 41:Remove Branches

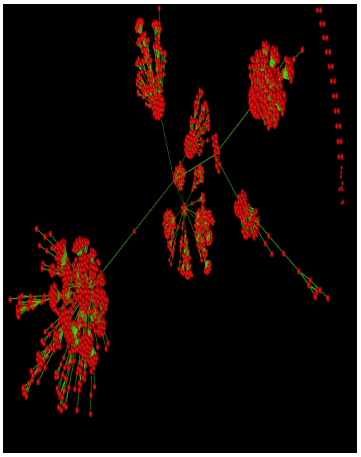


Figure 42:Spanning Trees

5.7.2. Summary

Graphia is a powerful open-source graph visualization application. It supports plugins and can handle graphs with millions of nodes and edges. It can run on a GPU and provides high-quality graph rendition. It supports different file formats, including CSV, TSV, adjacency matrix, and GraphML.

5.8. Summary of the Visualization tools

Medusa, Cytoscape, Arena3D, Gephi, and Graphia are popular tools for visualising and analysing biological networks. As shown above, each has different strengths and weaknesses. For instance, part of Medusa's strength is that it specialises in multilevel network visualization, which helps users explore hierarchical structures in a network. It also allows users to identify and analyse heavily connected substructures. Its weaknesses include fewer options for integrating different data types. Meanwhile, Cytoscape has vast collections of plugins, which broaden its functionality, making it adaptable for different tasks. However, non-experts may not be able to handle it. Furthermore, Arena3D provides interactive features to explore and manipulate 3D structures of biological networks, but it has a less extensive feature set for data analysis. Gephi's strength lies in its user-friendly interface, which makes it easily accessible for users with different levels of expertise. However, it may encounter performance issues when working with large-scale networks. Graphia offers support for the integration of multi-dimensional data, providing a detailed view of biological networks. Its weaknesses include its high license cost, which might be too expensive for some users.

Medusa is a good choice for specialised multilevel network visualization and node clustering tasks, while Cytoscape may be selected because of its comprehensive plugin ecosystem and

community support. Arena3D is used for 3D visualization, while Graphia has strong interactive analysis and data integration capabilities. Finally, Gephi offers a user-friendly interface and supports dynamic network analysis.

In summary, all these tools contribute to the advancement of research by empowering scientists with needed insights to visualise complex biological networks and complex biological data analysis and gain insights into complex relationships within these systems.

Table 25: Overall view of the tools from the researcher's perspective

No	Factors	Medusa	Cytoscape	Graphia	Arena 3D	Gephi
1	Filtering Tools	It has moderate filtering tools. It allows filtration by labels, nodes, and annotation. It uses regular expressions to match searches.	Supports column, degree, and topology filters. It has rich filtration tools that users can use to view information of their choice in the network.	It is capable of filtering by nodes. The graph can be centred on a filtered node. A group of nodes can be selected, viewed, and styled differently.	It has rich filtering tools that make it possible to show or hide lines, labels, grids, layers, etc. It includes numerous filtering tools, which are easy to use, access, and understand.	It is capable of directly selecting an object (node or connection) in the network. Selected nodes are filtered for viewing.
2	Plugins	No plugins – does not support plugins.	It offers great support for plugins, including Bingo, ClueGo, CluePedia, and StringApp.	It has many plugins and is highly extensible through the addition of plugins. It includes a plugin named Generic by default.	Does not support user-defined plugins. It is difficult to customise for personal usage.	Gephi supports plugins. Developers can implement plugins in Java to extend Gephi.
3	Visual Styles	It renders the network in 2D and represents clusters as circles of nodes. Network visualizations do not fit the screen by default. Some networks may not be able to fit the screen by adjusting the zoom level.	It renders 2D network graphs. It is mainly focused on high-level representations of interactions and components.	It is optimised for viewing large and complex graphs. Graphs can be viewed in 3D or 2D. The visualization interface makes full use of human cognitive abilities. Graph data are well rendered, allowing easy understanding and communication.	It renders networks as 3D images with a default dark background that makes the colour of the nodes stand out. The 3D visual result allows for rotation, flipping, and zooming, giving users a	The software accelerates the exploration process by using a 3D rendering engine to display large networks and graphs in real-time. This allows for the styling of nodes

					holistic view of the network.	and edges in the network.
4	Advanced Search	It uses regular expressions to search for nodes by labels or annotations.	It supports searching by edges, nodes, and attributes. Capable of searching the web.	It can search for nodes and centre the graph on the selected node.	It has many tools for searching for direct and indirect nodes, connections, and layers. It can search for keywords and descriptions.	It has limited search functionality. It supports direct selection as a search method.
5	Free/Open-Source	Free for academic use.	Free and open source.	Open source and free.	Free for academic use.	Free and open source.
6	Efficient Layout Algorithms	Supports random, circular, grid, hierarchical, distance geometry, Fruchterman, and Rheingold layouts. In addition, Medusa can use its relaxing feature to perform visual animation.	Provides support for layouts including hierarchical, stacked node, circular, attribute, attribute circle, and perfuse force-directed layouts.	Only a force-directed layout algorithm is available by default. Other algorithms are available as plugins.	Supports multi-layer, circular, random, and grid layouts. Uses efficient layout algorithms like Fruchterman and Rheingold.	It supports many layout algorithms, including ForceAtlas, Fruchterman, and Yifan Hu.
7	Scalability	Scales well with the size of the network. It is capable of visualising large networks.	Moderate capability in reading large datasets. It scales well with the size of the data.	It is highly scalable and can handle graphs with millions of edges.	The time complexity of loading and manipulating the network increases with the network size. It does	It is optimised for large datasets. The data is read faster than Medusa, Arena 3D, and Cytoscape.

					not scale with network size.	
8	Different File Formats	It supports CSV, TXT, or TSV formats.	Can read data from the National Data Exchange program (NDex), CSV, TSB, URL, and databases.	Offers support for a variety of input data formats, including TXT, CSV, and TSV GraphML	A single text file that contains information about layers, nodes, connections, clusters, etc.	Gephi supports various file formats, including CSV, GDF, GML, GEXF, Pajek NET, Graphviz Dot, and spreadsheets.
9	Text Mining	It uses regular expressions to search for nodes whose names match a given pattern. Not many text mining tools.	Cityscape's plugin StringApp supports advanced text mining and analysis.	It is an efficient search tool for keywords and retrieving useful textual information from the network. However, there are not many text-mining tools in the default plugin, although the application can be extended for text-mining purposes.	It has an efficient search tool to search for keywords and retrieve useful textual information from the network. However, it does not have many text-mining tools.	Gephi can be used for semantic network analysis, including stemming and lemmatization.
10	User Input & Customisation	Users cannot extend the application and can hardly customise the inputs.	The application can easily be customised and extended. Users can configure it according to their preferences.	The application can easily be customised and extended. Users can configure it to their taste.	Limited extension of the application. UI can hardly be customised. For example, the left pane of Arena 3D version 2 cannot be resized, and it is only possible to zoom in and	The application can easily be customised and extended. Users can configure it according to their preferences.

					out of the network with the scrollbar of an external mouse.	
11	Graph Analysis	It supports K-means, affinity propagation, and spectral and predefined clustering. It is a moderately effective tool for carrying out graph analysis on networks.	Capable of plotting histograms and generating summary statistics of the network. Plugins like WorldCloud and Network Analyser provide rich graph analysis features.	Capable of various types of graph analysis, including betweenness, eccentricity, page rank, MCL clustering, and Louvain clustering.	Limited graph analysis can be done. One of many possible graph analysis types is principal component analysis (PCA).	Can generate summary statistics.
12	Feedback to Users	Network visualization can be converted to an image file. Networks can be exported to PostScript, HTML, Pajek, Arena 3D, Bio-layout express 3D, Cytoscape, and Graphviz.	Has a show status tool capable of giving updates on processes run by the app. Supports exporting networks to NDex, web pages, and images.	Has tooltips and alert boxes to constantly update users on processes.	Reports and visualizations can be exported to different formats, including jpeg, Pajek, Medusa, and VRML.	Tools in Gephi have tooltips, and graphs can be exported as an image file or PDF file.
13	Strength	It provides simple, useful visualizations of predefined network clusters and can convert from one data type to another.	It is an advanced network visualization application with many plugins. A major strength is the flexible	Optimised for large and complex networks. High-quality graph rendition in 3D. Easily extensible.	Powerful application for 3D visualization of the network. Nice network rendering and representation. Nodes	Software is built on a multi-task architecture that makes use of multi-core processors and can be used to visualise

			display and series of layouts.		can be clustered in a layer.	large complex networks with over 20,000 nodes and generate relevant visual results.
14	Runtime Performance	It is fairly quick to load and run layout algorithms.	Fairly slow in running some layout algorithms.	Very fast in computation.	Slow, especially for large networks. Slow in loading and manipulating data.	Fast in terms of running layouts. Takes less time to represent the layout than tools like Medusa and Arena 3D.
15	User-friendliness	It has a simple and easy-to-understand user interface.	Nice user interface. Users can customise and rearrange panes. Application tools have tooltips.	Very user-friendly. The screen is maximised for viewing graphs. Offline tutorials are available. It has tooltips and is easy to use.	Panes are not moveable. It is not possible to zoom in and out of the network easily. There are no tooltips.	Nice user interface. Users can drag, zoom, and resize visual results. Tools names are self-explanatory.

No	Factors	Medusa	Cytoscape	Graphia	Arena 3D	Gephi
1	Information Coding	It can display up to 10 multiple edges that run simultaneously between nodes through the use of	It can be used to convert expression data into node labels, colour, border colour, or	Information is well coded. It provides top-notch representations of graphs	Effectively converts network data into visual information in 3D. It represents the network	It makes clustering, spatialising, navigating,

		Bezier curves. Supports predefined clustering.	thickness based on the user's configuration and visualization schemes.	in 3D and has an overview mode.	in multiple layers, which makes it more readable.	and manipulation of networks easy.
2	Flexibility	The interface cannot easily adapt to the specific needs of users. Users cannot readily customise the interface.	The interface can easily adapt to the specific needs of users. It allows the use of different data types and ways of creating network graphs. Supports command line tools.	Highly flexible. The interface can easily adapt to the specific needs of users. Users can have multiple windows, each with different plugins.	The interface cannot easily adapt to the specific needs of users. Panes are immovable, not resizable, and can have limited zoom capabilities.	Panes can be resized and minimised. Users can customise the interface of the visualization tools.
3	Orientation and Help	Medusa has a tutorial page on its website. Tools have tooltips and icons that explicitly and implicitly explain their functions, respectively.	It has a rich user manual and tutorial page. All tools have tooltips.	It has offline and online tutorials. The website is informative.	It has no help tab or tooltips. The website explains its usage, but it has not been updated for a long time, while most dependencies have undergone an update.	Has online documentation and tutorials. Users have access to information regarding the app's usage.
4	Minimal Actions	Extremely few actions are required to load and visualise the network. Changing network layouts requires a few steps.	It requires many steps to load data and style the network and nodes.	Loading CSV and TSV files requires several steps.	Few actions are needed to get results or to visualise the network. The network is visualised after loading the data.	Many steps are required to read data and run layout algorithms compared to tools like Medusa.

5	Prompting	Tooltips and icons serve as guides for users before taking any action.	Provides tooltips and alert boxes to users.	Users are well prompted for necessary actions. Has tooltips and prompt boxes.	No prompting is available. There are no tooltips. Users may take risky actions they did not intend to take.	Has prompt windows and buttons for users to run certain tasks. Tools in the application also have tooltips.
6	Consistency	Naming conventions and file-supported format formats are almost the same as those of convention tools.	The notations are consistent with conventional ones.	Notations are consistent with those generally used in the study of graphs and networks. Objects are named properly.	The data input format is different from those of conventional tools. It cannot accept traditional CSV files, TSV files, spreadsheets, or databases.	Notations in the application are consistent with the generally used ones (e.g. edges and nodes).
7	Spatial Organisation	It can effectively organise clusters of network data. However, the default size of the network is small. Zooming is often needed to view the entire network. Has a zoom-to-fit feature.	The network graph is well organized and is detachable to a new window.	Highly organised and intuitive rendition of networks. Nodes are arranged in ways that ease visualization, communication, and understanding.	The network is well represented in 3D with different layers. Predefined clusters can be visualised easily.	Nodes are nicely arranged in space. Nodes can be resized based on certain attributes.
8	Recognition Rather than Recall	The functions of tools can be easily understood. Tools and tabs are expected, and users do not need to recall most steps.	Tools can easily be recognised even without the tooltips. Tool icons are self-explanatory.	Buttons and tools are easily recognised. Conventional icons are used.	The usage of tools can be easily understood. Tools and tabs are given expected names. Users	Users can easily recognise tools and tabs based on their names and icons.

					do not need to recall most steps.	
9	Remove the Extraneous	The layout is simple, with no extraneous information. Space for visualization network is maximised.	There are extraneous panes, but they can be minimised.	The user interface is simple, with no extraneous and distracting panes.	The UI is simple, with no extraneous panes. Network visualization is not distracted by other tools or information in the UI.	There are extraneous panes, but they can be minimised.
10	Dataset reduction	Individual nodes can be selected, viewed, and deleted. The network can easily be pruned.	Data can be reduced. Tables can be loaded into an existing network.	The dataset can be reduced; %-NN, Edge reduction, and k-NN are some of the available data-reduction tools.	Data can be easily reduced to visualise information about a particular node, connection, or layer.	Data can be reduced by selecting nodes or connections of interest.

5.9. The selected tool - Cytoscape

Five network visualization tools were reviewed in the previous sections. Based on the literature, evaluation, and several meetings with the domain experts, Cytoscape was chosen for this study, particularly because it is an essential tool for network visualization and is useful for visualising complex biological systems. Some of its benefits include its ability to deal with complex visualization networks. Moreover, its interactive and customisable interface allows researchers to explore and analyse complex relationships within the data. Furthermore, Cytoscape has a vibrant community that is active in developing and maintaining a wide range of plugins, which extend the programme's functionality, allowing users to integrate additional algorithms, data formats, and analysis tools into their workflows. Cytoscape supports the integration of diverse data types, enabling researchers to overlay a series of information into the network visualization, thereby enhancing the exploration of comprehensive data (Piñero et al., 2021).

Cytoscape can be compared to other visualization tools like Gephi, which is capable of dynamic and interactive network visualization, is suitable for exploring large-scale networks, and supports different layout algorithms. However, Cytoscape is preferred in this study because of its broader user base and more comprehensive support in the scientific community. Graphia also emphasises visual analytics, and it is designed for the interactive exploration of complex networks (Mousavian et al., 2021). Although Graphia provides advanced visual analysis features, the extensive plugin ecosystem of Cytoscape offers a broader array of functionalities. Medusa is specifically built for multidimensional biomedical data visualization, and it effectively represents complex relationships within biological systems. However, Cytoscape has a more general-purpose approach and can be applied more broadly to diverse areas (Ragueneau et al., 2021). Arena 3D is usually adopted because of its 3D network visualization capabilities. Indeed, it has a lot of experience exploring networks in 3D space. Meanwhile, Cytoscape is more commonly used in 2D network visualization, although it supports some 3D visualization plugins like Arena 3D.

Additionally, Cytoscape supports integration with diverse data types and offers a platform for multi-omics data visualization. Doncheva et al. (2022) demonstrated that Cytoscape effectively depicted complex biological systems within biological networks. Cytoscape is used to understand and explore network and genomic sequences by loading, visualising, searching, filtering, viewing, and traversing network nodes and clusters.

5.10.Interactivity of Cytosacpe

The term ‘interactivity’ describes the interaction between users of digital materials or devices and the capacity of a computer, software, or another piece of material to react to a person’s actions (Genially, 2022). Cytoscape offers a highly interactive interface, allowing this interaction to be managed and altered. As part of the interactivity process, networks can be imported into Cytoscape from Excel spreadsheets. Interactivity helps a user better understand a technology’s features and use it to execute tasks more effectively. Further study is needed on interaction despite its significance in the Information Communication Technology (ICT) world. Currently, interactivity is one of the hottest conversation keywords in modern ICT. Communicating complex data concepts using plain text is difficult for many people to understand (Case Guard, 2022). Cytoscape layouts created or integrated by developers are referred to as layouts. A layout is the process of manipulating the network visualization and determining the node or edge location in a network, given certain restrictions. This feature is packaged as a reusable component in Cytoscape. Cytoscape has several layouts, including grid, hierarchical, circular, stack, attribute grid, prefuse force-directed, degree-sorted circle, and prefuse force-directed opencl. The choice of layout affects the tool's user experience. For instance, a network layout helps developers map out nodes or clusters (called edges), which helps them analyse the network and its behaviour. It can also be complemented by adding a style to the network, enhancing it visually and making it easy to understand (Cytoscape, 2022), while node layout tools help align, distribute, rotate, scale, and stack nodes. This feature is available via the menu command View >> Show Tool Panel via layout >> Node Layout Tools. The layout in Cytoscape affects the way networks are viewed. While other layouts could have been useful for visualization, the Cytoscape desktop application only has a static layout. However, adding animation to the layout will improve the visualization of the network and make it more engaging and fun to work with than a discrete layout. After evaluating various visualization tools and having several meetings with the domain experts, it can be concluded that Cytoscape needs to add more engaging forms of interactivity. Specifically, adding blur and fisheye view as layout-based properties in Cytoscape would be helpful.

5.11.Summary

This chapter focused on the evaluation of visualization tools. Different layouts can be used in different visualization tools, including random layout, circular, grid, hierarchy, Frutchterman–Reingold, K-means, stack layout, attribute layout, degree-sorted, ForceAtlas, and Markov clustering layouts. The evaluated visualization tools were Medusa, Cytoscape, Arena3D,

Gephi, and Graphia. Each of these tools has unique layouts as well as ones that are common to others. Medusa is suitable for specialised multilevel network visualization and node-clustering tasks, while Cytoscape offers a comprehensive plugin ecosystem and community support. Arena3D is used for 3D visualization, Graphia has excellent interactive analysis and data integration capacities, and Gephi has a particularly user-friendly interface and allows dynamic network analysis. Overall, each of these tools has strengths and weaknesses.

Five existing network visualization methods used for biological networks were evaluated based on their usability and limitations. They are Medusa, Cytoscape, Graphia, Arena3D and Gephi. Sections 5.3 to 5.7 comprehensively describe how the usability of those complex biological networks can be improved. Thereby answering research question 3.1. Furthermore, interestingness measures such as random layouts, circular layout, grid layout, edge reduction, and ForceAtlas layout, among others, in section 5.2, subsection 5.2.1 to 5.2.12, answer research question 3.2 and thereby do justice to objective 3.

Chapter 6: Introduction to Enhanced Visualization Tools

This chapter provides detailed information on the enhanced visualization tool developed for this study, as well as background information on the visualization techniques discussed in the literature. It also includes detailed information about the fisheye view and blur techniques used in the enhanced visualization tool. The researcher focused on interactivity, so the fifteen general factors were considered during the development stage. Therefore, the tool has only been evaluated using the ten heuristic factors and the ICET.

6.1. Background Information About Visualization Techniques

A range of visualization techniques exists that are designed to help users focus on certain parts and details in the data while maintaining awareness of the complete data structure. Particularly because biological networks like protein-protein interaction (PPI) networks, gene regulatory networks, and metabolic pathways, among others, are inherently complex. They have large numbers of components known as nodes and intricate relationships known as edges. The visualization technique then allows simplification and representation of this complexity, making it easier for researchers to understand, analyse and extract detailed information from the complex networks. While the research presented in this thesis has focused on fisheye views, several other alternatives have been considered. A summary of these alternatives is provided below, followed by an in-depth description of the fisheye view. Fisheye view, a visualization technique, balances detail and large complex dataset contexts (Janecek and Pu, 2002). This technique allows users to focus on certain parts of the data while maintaining an awareness of the overall data structure. Some similar techniques include the following:

Zooming and panning: Zooming is the ability to alter the scale of a visualization, allowing users to focus on certain areas of interest within the network. Meanwhile, panning entails moving the view of visualization horizontally or vertically, allowing users to navigate across the network (Franconeri et al., 2021). Both are critical to the interactive exploration and analysis of complex biological networks and help researchers investigate network structures interactively, leading to the identification of important components and a deeper comprehension of the relationships and interactions within the network (Junker et al., 2021). Specifically, zooming enables users to focus on certain nodes of interest and supports comprehensive data exploration (de Paula, 2019), while panning allows users to move around the whole biological network to obtain a comprehensive view of it. However, despite these obvious advantages, there are inherent disadvantages of this technique, which is why it was not

chosen for this study. One major issue is the tendency to make navigation complex, especially if users lose track of their position within the network (Jusufi, 2013). Panning over large networks requires a great deal of navigation time, which may make it overly cumbersome for users to move across extensive biological networks (Praneenararat et al., 2011).

Overview + Detail: This technique combines an overview visualization and detailed views. Users can interact with an overview to select and zoom into a certain area, while the detailed view furnishes the users with additional information (Cockburn et al., 2008). One of the main strengths of this technique is that it allows users to zoom in on a specific node and gather comprehensive information, which facilitates the identification of important components, clusters, or pathways within the network and helps in analysing specific biological processes (Cockburn et al., 2008). It also enhances pattern recognition and anomalies, which is important for biological network analysis, especially where subtle patterns could indicate specific biological phenomena (Kim & Mukhiddinov, 2023). However, some key limitations prevented the programmer from adopting this method. It can be challenging to implement this technique effectively, especially with large and complex biological networks, and balancing the representations of both global and comprehensive information can overwhelm users (Kim & Mukhiddinov, 2023).

Treemaps: This technique shows hierarchical data using nested rectangles, where the size and colour of each rectangle are a representation of different parts of the data. In complex biological networks, treemaps are used to represent hierarchical structures, such as the relationships between entities like genes and functional categories (Gillespie et al., 2022). Some of the inherent advantages of treemaps include effective representation of the data structure hierarchy, which is useful for illustrating the organisation of and relationships in complex biological networks (Gillespie et al., 2022). In addition, it is easy to identify individual nodes and clusters within a treemap, which facilitates the analysis of specific pathways in biological networks (Balzer & Deussen, 2005). However, despite the obvious advantages, some inherent limitations prevented the researcher from adopting this technique. Treemaps can be difficult to interpret due to cluttering, especially when the biological network is densely interconnected. Moreover, irregularly shaped cells in the treemaps could distort the representation and readability of the visualization, hindering accurate interpretation (Balzer & Deussen, 2005).

Heatmaps: Heatmaps represent data values in a grid-like form based on colour intensity. With this technique, users can zoom in to view data inputs closely and specifically and adjust the

granularity level (Rohlig et al., 2019). Heatmaps provide a visual representation of the relationships between many components in a biological network, which is useful for understanding co-expression patterns, interactions, or correlations among genes or proteins. At the same time, graphs represent interactions and relationships between entities (Rohlig et al., 2019). However, the researcher chose another technique because of some of the limitations of heatmaps. They are primarily visual tools, and their interpretation may be subjective due to a heavy reliance on users' ability to identify patterns. They may inadvertently place too much emphasis on certain features while others may be overlooked (Wong, 2012).

Focus + Context Cartograms: This technique uses distortion to emphasise certain regions while maintaining a contextual view. It is mostly used with geographical (Nusrat & Kobourov, 2016). An advantage of this technique is that it distorts less significant nodes, which helps reduce visual clutter, making it easy to focus on the most relevant details (Marcílio-Jr et al., 2021). However, it is subject to distortion, which could lead to the loss of more comprehensive details, which might be significant when attempting to obtain a detailed understanding of a biological network (Marcílio-Jr et al., 2021). Consequently, the researcher did not choose this technique due to a preference for detail and precision.

Semantic Zooming: This refers to changing the detail or abstraction level based on the actions of the users. For instance, semantic zooming may be used to show enhanced information about data elements when zoomed in, while less detail is presented when the user zooms out (Bhowmick et al., 2023). Part of the strength of this technique is that it enhances the user experience by allowing intuitive exploration and navigation of complex biological data, which makes it easier for researchers and other users to identify patterns and relationships (Bhowmick et al., 2023). However, zooming in too much on highly detailed data could overwhelm the users with a large volume of unnecessary information, making it challenging to obtain relevant insights. Hence, it was not chosen for this study.

Fisheye View: This technique allows distortion of data representation to give focus and context to a visualization instance. Users can zoom in on a certain part of the visualization while maintaining an understanding of the encircled data that is already compressed and distorted (Turetken & Schuff, 2002). The fisheye view is a valuable approach for visualising complex biological networks, particularly in terms of data preservation and reducing clutter. Hence, the researcher chose it for use in this study.

All these techniques are designed to assist users in moving through the data effectively and understanding the complexity of the data by creating a balance between local and global contexts. However, which technique should be adopted depends on the specific needs of users and the available data.

6.2. Fisheye View

The fisheye view is a visualization technique for distorting data representation and providing a focus-and-context view of a larger dataset. Blurred nodes outside the radius effectively direct the users' attention to the selected node and its immediate surroundings. This is beneficial because reducing the visual clutter allows the users to focus on more relevant information with less or no distraction. It is useful for showing complex information, as specific areas of interest are highlighted while displaying the surrounding context as well. Common applications are described in the following.

6.2.1. How it works

The fisheye view consists of a focus region, context region, and transition. A focus region is a certain region, usually the centre of the visualization, where the data details are shown accurately. Meanwhile, the context region is distorted to produce a compressed view. The distortion degree increases as the user moves further from the focus region. The transition between the region focuses, and contexts is stepwise, allowing the viewer to transition effortlessly between detailed and compressed views. A simplified overview is presented in Figure 43.

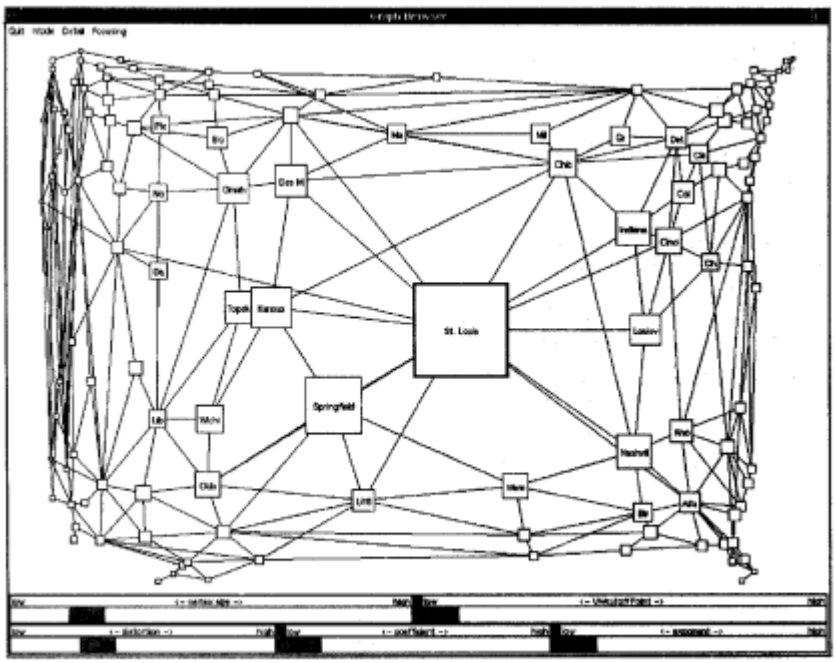


Figure 43:A simplified fisheye view of the graph (Sarkar & Brown, 1992)

6.2.2. ***Application***

The fisheye view can be applied in different areas, including graph visualization, information retrieval, and document browsing. For instance, it is commonly used for graph visualization and to visualise complex networks or graphs, such as computers and biological and social networks (Huang & Lai, 2006). It lets users or viewers focus on certain nodes while offering a global overview of the whole network (Huang & Lai, 2006).

Another area in which the fisheye view is applied is information retrieval. It may be used in the search interface to display results, while the central area shows comprehensive information about the chosen items and the output is compressed (Janecek et al., 1970). Fisheye views are also useful in document and web browsing to provide a detailed view of a chosen document or webpage. At the same time, they present a condensed view of close content, making it simple and easy to move through collections of large documents (Janecek & Pu, 2002). Furthermore, fisheye views are applicable to geographic information systems (GIS), as they provide comprehensive views of certain geographic areas while showing the entire map ((Huang & Lai, 2006).

In software development, the fisheye view may be used in sourcing for code visualization, which enables programmers to focus on certain sections of the code while maintaining the code's structure (Storey, Best & Michand, 2001). It is also valuable for network traffic analysis

and monitoring, allowing the network administrator to zoom in on certain areas of the network while tracking the overall traffic patterns. In task management, the fisheye view helps manage to-do lists by emphasising critical tasks while displaying the less important ones in a compact form.

Overall, fisheye views effectively balance the need for details and context, can deal with large, complex datasets, and help users maintain situational awareness while allowing them to zoom in on other interesting areas for a more detailed analysis.

6.2.3. Benefits and drawbacks of using the fisheye view for visualization.

There are many benefits of using the fisheye view for visualizations. Some of these include balancing detail and context, improved focus, and contextual understanding. For example, in balancing details and context, the fisheye view can provide the right level of detail and context for large or complex datasets. The fisheye view allows users to zoom in on areas while keeping an overview of the overall data structure (Luciani et al., 2018). Furthermore, it provides a clearer focus on the central area, where comprehensive details are accurately presented, making it easier to concentration on certain data areas or points (Kumar, 2022).

Fisheye views help to provide contextual understanding, as users can understand the surrounding context, which is visible even though it has been compressed visually. It is important to understand how a certain data point interacts with the entire region (Kumar, 2022). Additionally, users can effortlessly change between focused and compressed views, making it easy to navigate the data. Furthermore, the fisheye view supports interactive exploration, allowing users to zoom in, panning, and interact with data in real-time, thereby improving their engagement and understanding. It is particularly useful here for data structure hierarchies like graphs and trees, allowing users to explore various detail levels in the hierarchy.

There are some known drawbacks to using a fisheye view for visualizations. For instance, it distorts the data outside of the centre of the field of view. The more the user zooms out from the centre of view, the more distorted the data becomes, making it difficult to read correctly (Storey et al., 2001). Some users may not find it easy to use the fisheye view at first because of the distortion, requiring time to attain proficiency. It is particularly effective with circular or radial layouts (Brauer-Burchardt et al., 2010). Moreover, creating and executing visualizations with fisheye views can be difficult due to the complex algorithms and user interfaces needed to ensure smooth transitions and interactions (Brauer-Burchardt et al., 2010). Importantly, this

technique is only beneficial when there is a need to focus on certain elements in a wider context, and it relies on user interaction, as a need to zoom in and out of the visualization.

6.3. Fisheye View of Complex Biological Networks

Exploring and analysing complex biological networks present some unique challenges, including distinguishing complex relationships within large volumes of data. Applying innovative visualization techniques is crucial to fully understand these networks. One approach that is gaining prominence is the fisheye view for complex biological networks, which can be used in the following ways.

Hierarchical structure:

Hierarchical or modular networks are used in many complex biological networks, including Protein-to-protein interactive protein-protein interaction (PPI) networks and gene regulatory networks. Fisheye views display hierarchical data well, allowing users to zoom in and out of the view to see different levels in the hierarchy (Junker et al., 2006).

Focus on key components:

Biological networks often contain essential nodes or routes that researchers wish to examine in greater detail. Fisheye views enable users to focus on specific nodes or areas, thereby facilitating the study of critical components (Cockburn et al., 2008). However, it should be noted that fisheye views may not be as effective for visualizing long paths within the network.

Maintaining context:

Biological networks with many connected parts can be incredibly large and complex. Using fisheye views, it is possible to keep track of the details of the network while also being able to zoom in on specific areas (Schaffer et al., 1996). This allows users to comprehend how their own interactions are connected to the larger network.

Interactive exploration

Most fisheye views offer interactive exploration, which means it is possible to zoom in and out and interact with the data. This is especially helpful when working with intricate biological networks, as it is possible to explore different parts of the data and paths in real-time (Schaffer et al., 1996).

Spatial organisation

Fisheye views can be used with either a circular or radial layout, which is useful for many biological networks. A circular layout can reveal the connections between elements like genes or proteins in an easy-to-read way (Tominski et al., 2006). However, it is not useful for force-based layouts and others because they are dynamic iterative and adjust continuously to attain equilibrium where all the forces are balanced, making it challenging to use.

Reducing visual clutter

Nodes and connections in biological networks can be very dense, and a fisheye view can help reduce the visual clutter by reducing the amount of irrelevant information in the background, allowing the user to concentrate on the centre of the network (Holten, 2009).

Navigational aid

It is challenging to navigate complex biological networks. Fisheye views provide navigational help, enabling users to effortlessly transition between a focused view of certain genes, pathways, or proteins and an overview of the whole network (Boyle et al., 2012).

Identifying pathways and clusters

Fisheye views allow users to identify pathways, networks, or subnets within a larger biological system. Scientists can focus on specific regions to study the functional relationships among genes, proteins, or other biological systems (Holten, 2009).

Fisheye view techniques are particularly beneficial for large biological network visualization. It is important to consider certain dataset features, and the goal of this study was to determine whether this technique is the most suitable choice. It is also crucial to consider users' familiarity with when implementing this technique for biological networks to ensure effective data exploration and interpretation.

6.4. Technology and Architecture

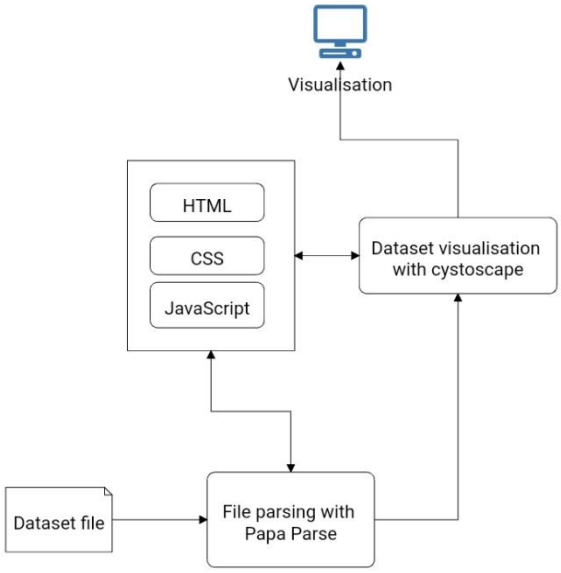


Figure 44:System architecture of the tool

The system's architecture is presented in Figure 44, showing the flow of data and the interrelationship between the system's components. From the figure, we can see that the file containing the dataset to be visualised is parsed by Papa Parse, which is reputed to be the fastest plain-text file parser. The file parser uses JavaScript technology to parse the file and extract the dataset as an array of objects. The array of objects is then used as an input to the dataset visualization module, which Cytoscape powers. The module uses HTML, CSS, and JavaScript to visualise the dataset. The final visualization of the dataset is rendered on the monitor's display.

Some of the technologies used in developing this tool are summarised below.

1. Cytoscape.js: This is a JavaScript library for visualising data. This library powers the actual visualization of data in this tool.
2. Papa Parse: This library is used to parse plaintext files. It has the reputation of being one of the fastest plaintext parsers in the world.
3. HTML: This was used for the layout of the front end, including the canvas for visualization.
4. CSS: This was used to apply HTML styling.
5. Vanilla JavaScript: This made the tool dynamic.

6.4.1. Algorithm for Fisheye view and blur technique

1. Get the position of the node that was clicked.

2. For each visualised node, do the following:

   a) If this node is the clicked node, skip it and go to the next iteration. Else, proceed.

   b) Get the X coordinate of the position of this node.

   c) Get the Y coordinate of the position of this node.

   d) Get the difference between the x coordinate of this node and the X coordinate of the clicked node. Save the value as distanceX

   e) Get the difference between the y coordinate of this node and the Y coordinate of the clicked node. Save the value as distanceY

   f) Compute the distance between this node and the clicked node using the equation:

$$\sqrt{\text{distanceX}^2 + \text{distanceY}^2}$$

   g) Compute the new height of this node using the equation:

$$\frac{\text{baseHeight}}{1 + (\text{distanceBtwNodes}/\text{radius})^2}$$

   h) If the distance between this node and the clicked node is less than or equal to the specified radius, then this node is within the radius. Then do the following:

      i. Apply the inRadiusStyle (defined in Appendix G) to this node.

      ii. A node's height must not be less than the minimum height; otherwise, the node will become too small for visibility. Hence, if the computed height is less than the minimum height, then set the new height to the minimum height.

      iii. Apply the new height and width to this node. The width is the same value as the height.

      iv. Make all the edges between the clicked node and the nodes within the radius to be more pronounced.

   i) Otherwise, if the distance between the current node and the clicked node is greater than the specified radius, this node will be outside the radius. Then do the following:

      i. Apply the blurredStyle (defined in Appendix G) to this node.

ii. Apply the new height and width to this node. The width is the same value as the height.

iii. Make the edges between the clicked node and this node to be blurred by decreasing the line width and the opacity.

3. If the graph is a complex graph, then apply the selectedNodeStyle (defined in Appendix G) to the nodes that were initially selected from the dropdown menu. This is necessary because the style could have been changed from the above operations.

4. Apply the clickedStyle (defined in Appendix G) to the node that was clicked.

6.5. Incorporating Interestingness into the Design of the Enhanced Visualization Tool

In general, integrating measures of attractiveness into a visualization tool requires combining different methods and techniques to guarantee that the displayed data is captivating, pertinent, and beneficial to the user. Here's a thorough description of how it was incorporated into the design and implementation of the enhanced visualization tool.

1. Data Pre-processing and Analysis

Before visualization, data must be processed and analysed to identify the most exciting elements. This involves several steps which were used in the pre-processing of the data.:

- **Data Cleaning**: Eliminating unimportant or disruptive data that could divert the user's attention. This process guarantees that the data used for visualization is precise and dependable, thereby preserving the integrity of the insights being conveyed (Himeur et al., 2023).
- **Feature Selection entails identifying** the primary characteristics or aspects of the data that are expected to be noteworthy (Khaire & Dhanalakshmi, 2022). This entails choosing the variables that significantly influence the result or offer the most understanding of the data being examined (Khaire & Dhanalakshmi, 2022).
- **Statistical Analysis**: Analysing data with statistical techniques to identify meaningful patterns, trends, or irregularities. Methods such as regression analysis, correlation, and hypothesis testing can reveal connections within the data that may not be readily obvious (Guetterman, 2019).
- **Text Mining:** Using techniques from natural language processing (NLP) to derive valuable information from textual data. Text analysis can uncover patterns, emotions, and other

valuable details from extensive text data, which can be presented visually for a better understanding (Gutierrez et al., 2021).

2. User Profiling and Personalization

Understanding the user's preferences and behaviour is essential for personalizing visualization. Therefore, the following steps were conducted.

- **User Profiles**: Develop comprehensive user profiles by analyzing their previous engagements, preferences, and demographic data. These profiles help tailor the visualizations to match different user segments' specific interests and needs (Dwivedi et al., 2022).
- **Feedback Loops:** Create a system for users to give input on the visual representations, which can be utilized to improve upcoming content. This may involve systems for rating, commenting, or asking direct survey questions to enable users to share their satisfaction and provide suggestions for enhancements (Sutton et al., 2020).

3. **Visualization Design Principles**

Creating visualizations that emphasize engaging content requires the application of numerous fundamental principles. The following are some of the concepts that were incorporated into the system.

- **Relevance and Context**: Ensuring the visual representations are appropriate for the user's specific situation and inquiries (Lynch, 2001). For example, a financial overview should emphasise necessary measures like profit margins and revenue patterns, whereas an educational resource might concentrate on student achievement statistics (Hastings et al., 2013).
- **Clarity and Simplicity:** Use simple and straightforward designs to make the data easy to understand. Reducing clutter and focusing on essential data points can improve engagement, as this will help users grasp the information quickly (Franconeri et al., 2021).
- **Aesthetics:** Utilizing attractive visual elements such as suitable colour schemes, fonts, and designs to captivate and maintain users' interest. The visual components should enhance the information instead of taking attention away from it, ultimately improving the user's experience (West et al., 2020).

4. Interactive Elements

Incorporating interactive features can significantly increase the appeal of visual representations. Following are some of the features that were incorporated into the enhanced visualization tool to improve user experience.

- **Filtering and Drilling Down:** Enabling users to refine data and delve deeper into specific perspectives assists them in investigating elements that capture their interest. Users can personalise their experience and concentrate on the data that matters most to them (Sjodin et al., 2021).
- **Dynamic Updates:** Keeping the visualizations up to date in real-time or near real-time helps maintain their relevance and keep the audience interested. This is especially crucial for dashboards that track continuous processes or real-time events, where timely information is crucial (Nadj, Maedche and Schueder, 2020).
- **Tooltips and Annotations**: Adding tooltips and annotations that provide additional context or insights when users hover over data points. These components can provide further explanations, relevant data, or references to more comprehensive information, enhancing the user's comprehension (Jueneman, 2023).

5. Advanced Analytical Techniques

Using advanced analytical methods can assist in discovering and emphasising noteworthy data points. The following aspects were discussed in the literature, but it is proposed that it could be used in the future to enhance the tool.

- **Machine Learning Models**: Machine learning techniques analyse past data to anticipate which data or patterns will capture the most attention from users. These designs can utilise previous user engagements and preferences to emphasise the most pertinent information in upcoming visual representations (Lisboa et al., 2023).
- **Anomaly Detection**: Using methods for anomaly detection to identify unusual patterns or outliers that could be especially noteworthy. The presence of outliers can offer valuable insights into underlying issues or possibilities that may be missed when focusing on more typical data points (Habib ur Rehman et al., 2017).
- **Trend Analysis** involves monitoring and representing patterns over time to understand how specific measurements are developing. Analysing trends can uncover extended patterns and changes, providing insight into the past and enabling predictions of future events (Samariya and Thakkar, 2023).

6. User Feedback and Customization

Ensuring the visualizations stay meaningful and enjoyable involves incorporating user feedback methods and customisation choices.

- **Customisation Choices:** This option allows users to personalise the visual representations based on their preferences, like picking specific metrics to show or selecting various chart styles. Personalising the experience through customisation increases user involvement by enabling individuals to adapt it according to their preferences (Sarker, 2021).
- **Feedback Mechanisms:** Creating functionality for users to rate or comment on the visualizations to gather feedback for enhancing future versions. Continuous feedback helps to keep the visualizations aligned with user needs and ensures ongoing relevance and interest (Dwivedi et al., 2021).

Summarising and assessing interestingness in data visualization and user interface design is a complex process that includes thorough data preparation and analysis, a firm grasp of user preferences, adherence to practical design principles, interactive features integration, advanced analytical methods, and ongoing user feedback and personalisation. By concentrating on these aspects, designers and developers can produce compelling and visually attractive data visualizations that improve user experience and satisfaction.

6.6. Design and Implementation of the Blurfisheye visualization Tool

The design and implementation of a fisheye view is an important approach to visualization. This innovative method utilises fisheye distortion to focus specifically on certain elements within an intricate network, providing researchers with a nuanced and enlarged perspective of the complex system. It has been shown clearly that this technique can increase network analysis efficiency, revealing hidden patterns and fostering a deeper understanding of the underlying biological relationships. A description of how the functional requirements of the visualization tool were implemented is provided below. The Blurfisheye visualization tool can be accessed at https://blurfisheye-visualization.com/.

6.6.1 Visualization of a dataset

Users can use the tool to visualise data, as shown in Figure 45. They can select a file from the drop-down menu labelled ‘Select a file to visualise’ or upload a file containing the dataset they wish to visualise.

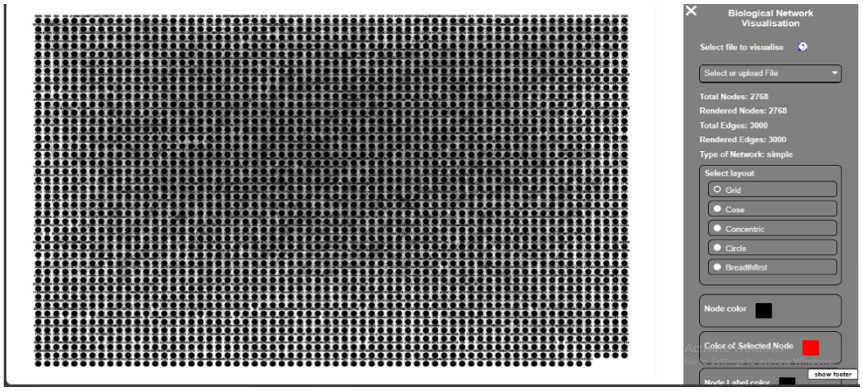


Figure 45:Data visualization

6.6.2 Visualization of all the nodes in a simple dataset

If a graph has 7,000 or fewer connections (edges), and all the nodes and edges are rendered when the dataset loads.

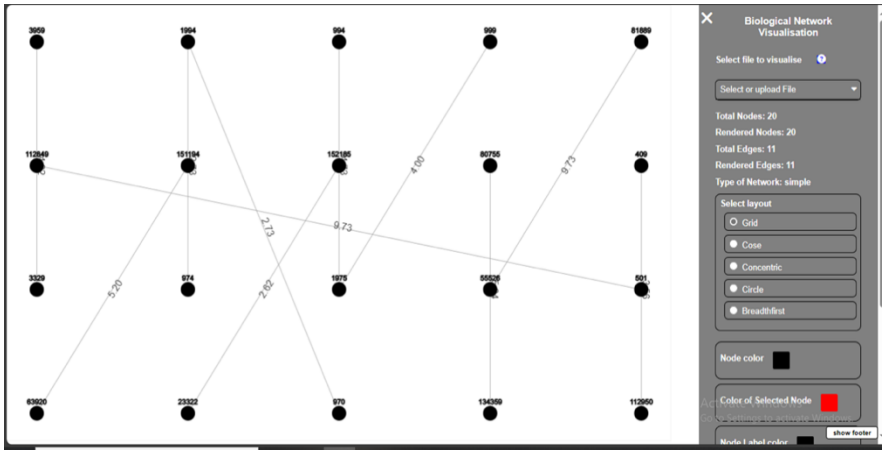


Figure 46:Visualization of a simple graph

6.6.3. Allowing users to select the nodes to be visualised in a complex dataset

A complex dataset is one that has more than 7,000 connections (edges). The tool does not render this type of dataset when the data are loaded. Instead, the user has to select the nodes that they want to render using search and filter, attribute-based selection, and manual selection. This is because a very large dataset requires a great deal of resources to be fully rendered. Figure 47 shows the state of the graph when the data are fully loaded. Figure 48 shows the dropdown menu before selecting a node from the menu, while Figure 49 shows the graph after rendering a selected node.

Figure 47:Initial rendering of a complex dataset

Figure 48:Selection of the node to render

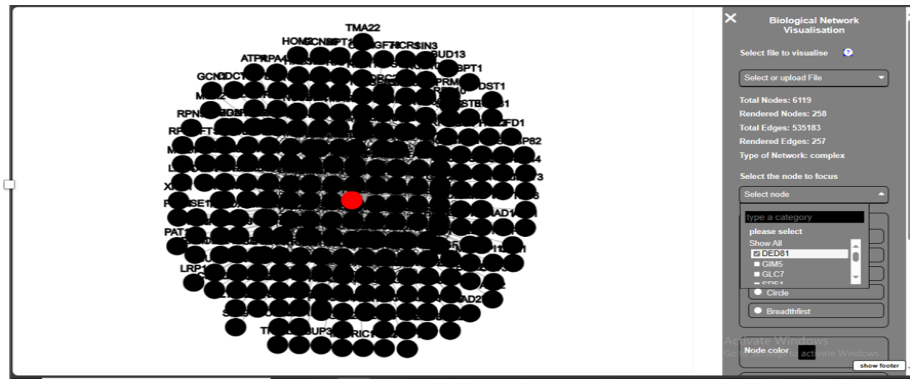


Figure 49:Rendering of the selected node

1. The fisheye effect when the user clicks a node in the graph

When the user clicks on any of the nodes in the graph, all the nodes outside the radius of the fisheye effect are blurred. Additionally, it is blurred so that users can smoothly explore different parts of the graph. As the mouse is moved or interacts with the graph, there is a dynamic change, which shows the details while keeping the big picture intact. Further, the size of the nodes is recalculated based on their proximity to the clicked node (Figure 50).

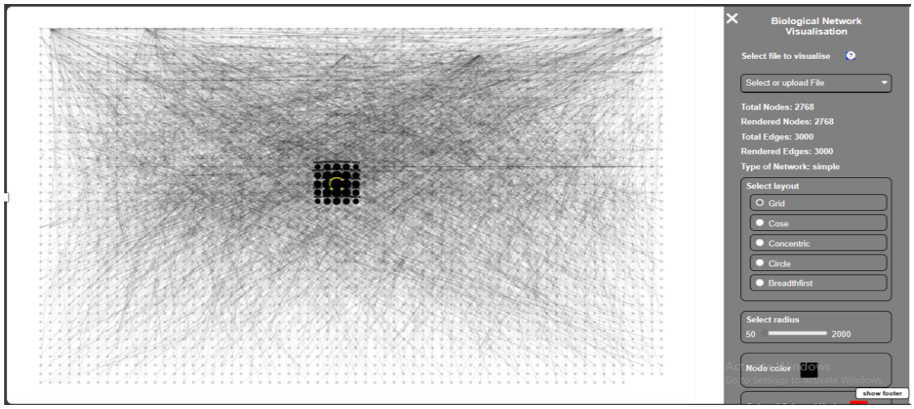


Figure 50:Fisheye effect

2. Changing layout algorithms from the toolbar

The tool has five layout algorithms from which the user can choose. These include the grid, CoSE, concentric, circle, and Breadthfirst layouts. The user can change the layout from the toolbar section labelled 'Select layout'. The rendering of each of the layouts is shown in Figures 51 to 55.

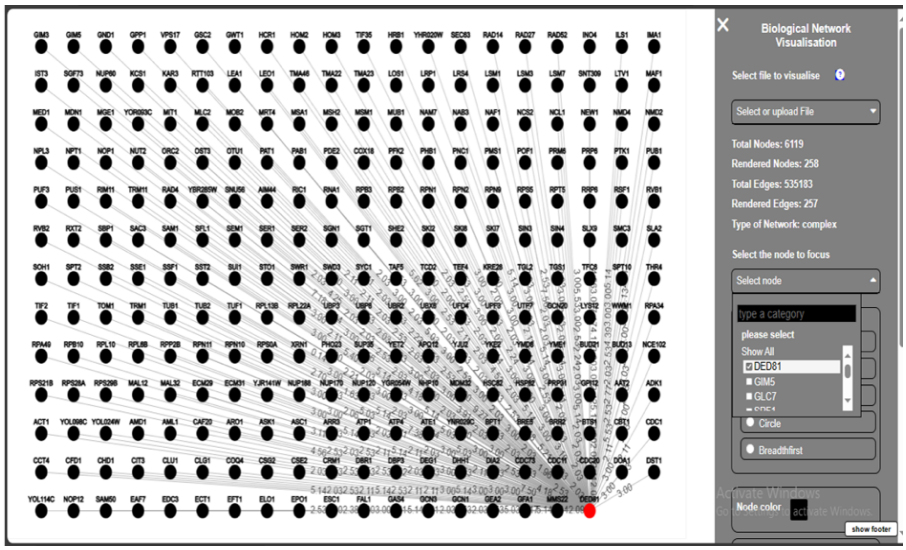


Figure 51:Grid layout

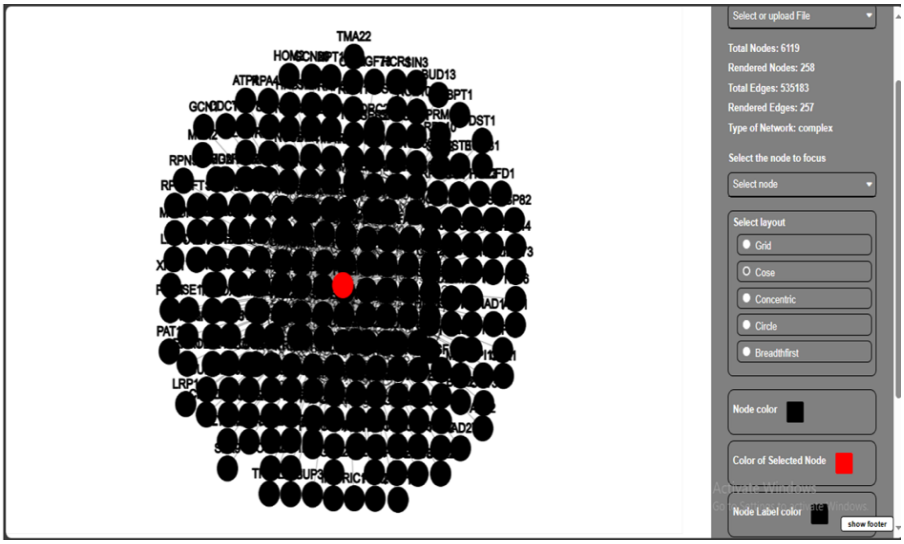


Figure 52:CoSE Layout

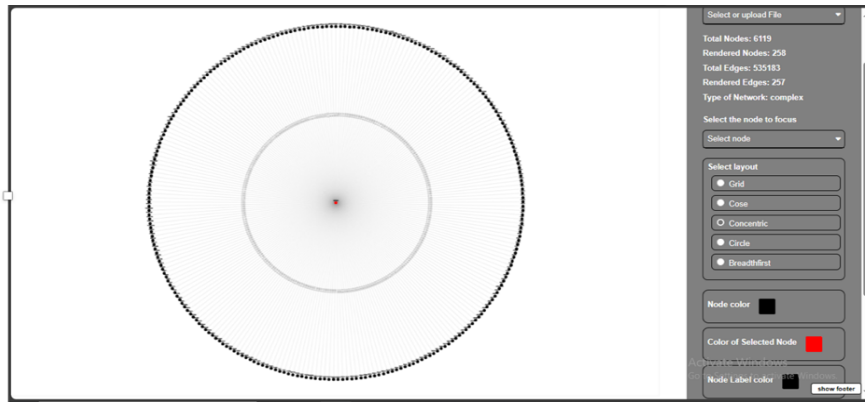


Figure 53:Concentric layout

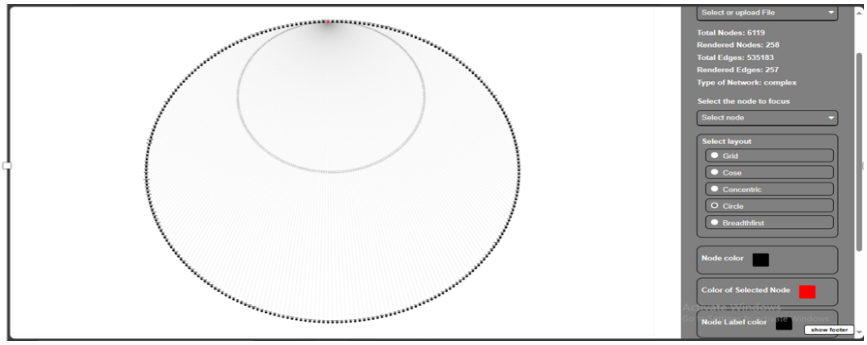


Figure 54:Circle layout

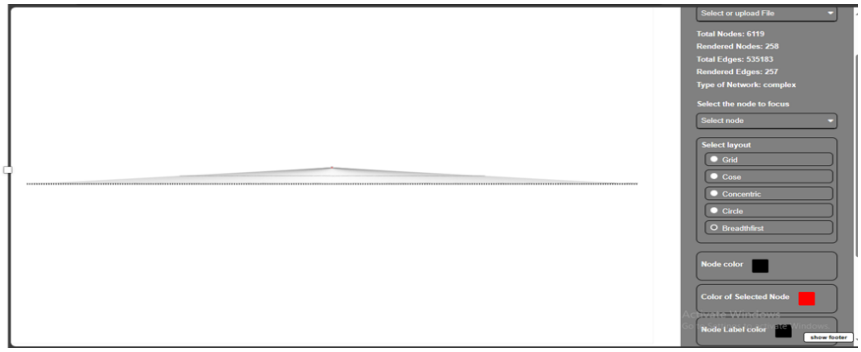


Figure 55:Breadthfirst layout

3. Allowing users to customise the tool

Users can easily change the tool's appearance by changing colours and hiding labels using the toolbar. The attributes that can be changed are the node colour, the colour of the selected node, the colour of the node label, the colour of the edges, and the background colour of the canvas. Figure 56 shows a fully customised graph.

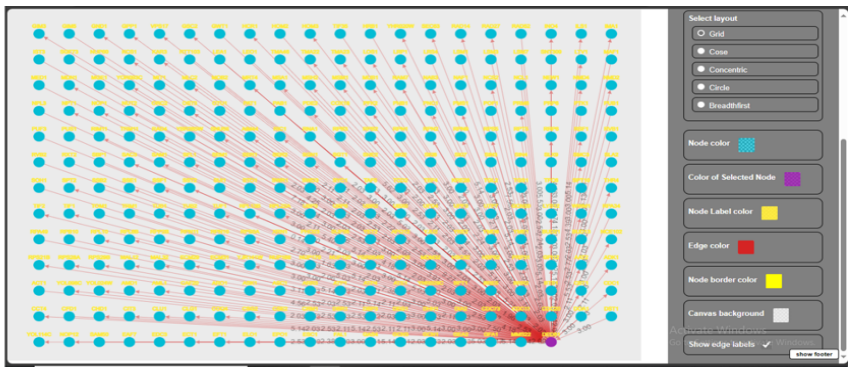


Figure 56:Customised graph

4. Tutorials and user guide

Tutorials were added to make it easy for users to find their way around the tool. The tutorials cover uploading files, changing data visualization algorithms, creating a fisheye effect, changing the fisheye radius, changing the colour of nodes, changing the colour of edges, disabling edge labels, and changing the background colour. Figure 57 shows the tutorial view. It can be opened by clicking the tutorial button in the toolbar.

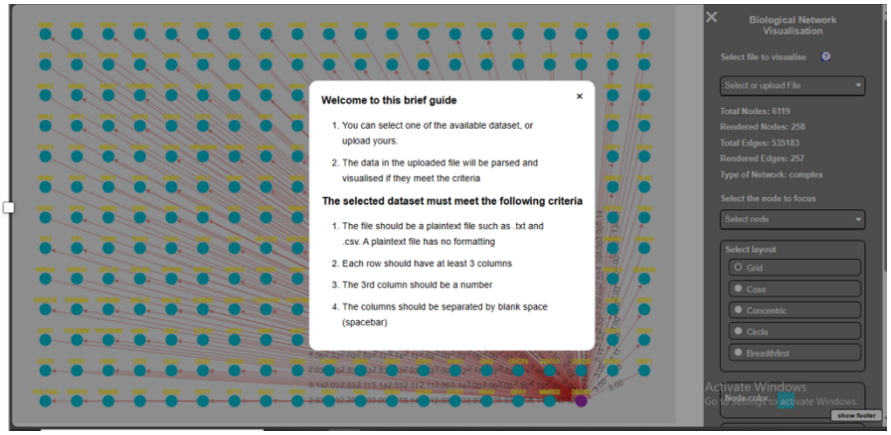


Figure 57:Tutorial

5. Owner's details in the footer

The details of the owner of the project were added to the tool's footer. The view can be toggled by clicking the show/hide footer button. The footer can be seen in Figure 57.

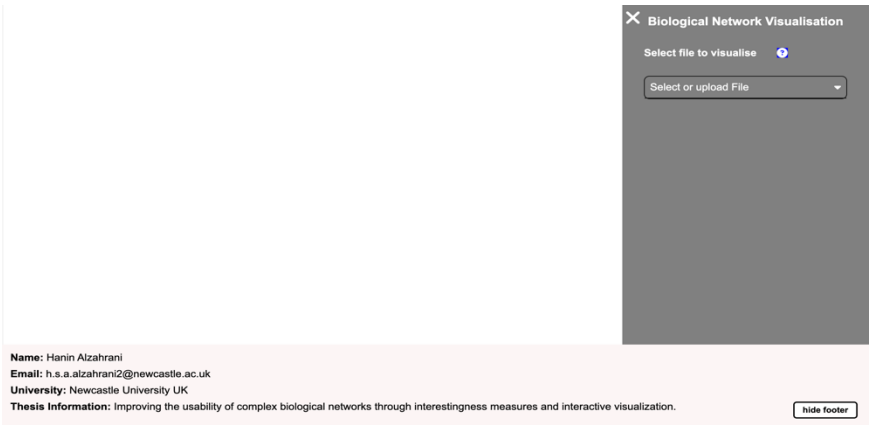


Figure 58: Owner's information

6.7. Alternative methods

6.7.1. Sedlmair's Design Study Methodology (DSM)

This nine-stage framework, Sedlmair's DSM, has been proven to guide the visualization designs for complex, real-world problems like biological networks. The stages are Learn, Winnow, Cast, Discover, Design, Implement, Deploy, Reflect, and Write (Sedlmair et al., 2012). They are further divided into three categories: the precondition stage, the core stage, and the follow-up stage.

i. The precondition stage, also known as personal validation, comprises learning, winnow and cast.

ii. The core stage is also known as inward-facing validation and comprises discover, design, implement and deploy.

iii. The follow-up stage consists of reflecting and writing.

Emphasize Sedlmair's DSM's iterative nature, which focuses on development and validation at different stages. This approach effectively ensures that visualizations meet user needs and solve technical constraints.

6.7.2. Munzner's Nested Model

Munzner's nested model, an abstract and hierarchical framework for visualization design, is known for its adaptability. It features four levels of decision-making: domain characterization, data abstraction, visual encoding, and algorithm design (Munzner, 2009).

i. Domain characterization: this is to understand the specific domain where visualization will be used.
ii. Data abstraction: This is where domain-specific needs are transformed into abstract data types.
iii. Visual encoding: This is where the visual representation of the data is designed and how users interact with it.
iv. Algorithm design: This ensures that the underlying algorithms efficiently support the visual encoding and interactions.

6.7.3. McCurdy's Action Design Research

This framework, McCurdy's Action Design Research, integrates action research principles with design sciences. It focuses on iterative development and close collaboration with stakeholders throughout the design stages, ensuring that everyone's input is considered. It is useful where the problem definition and solution evolve together through design cycles and evaluation (McCurdy et al., 2016). The framework encourages solutions that are not technically driven but practicable in real-world applications.

6.7.4. Comparison

Table 26: Comparison between alternative methods

	Sedlmair	Munzner	McCurdy
Level of abstraction	Pragmatic and more grounded in real-world application, with emphasis on	It is highly abstract but focuses on systematic validation at each level. As	Flexible, adaptive and ideal for cases where the problem is well-defined and

	iterative development and stakeholder engagement (Sedlmair et al., 2012)	such, it is more suitable for projects with meticulously structured design processes (Muzner, 2009).	needs continuous iteration and stakeholder engagement (McCurdy et al., 2016)
Iterative development	It is iterative all through its stages, especially at the core stage (Sedlmair et al., 2012)	Iterative within each design level (Muzner, 2009)	Iterative all through the whole process, with continuous development and reflection cycle (McCurdy et al., 2016)
Stakeholder engagement	Holistic stakeholders' integration throughout the whole process, particularly the pre-condition and follow-up stages (Sedlmair et al., 2012)	Stakeholders are involved right from the initial domain characterization stage (Muzner, 2009)	Strong stakeholder collaboration throughout the process shows its roots in action research (McCurdy et al., 2016)

6.7.5. Justification for Blurfisheye's view approach against the other three models

i. Blurfisheye view framework for visualization design integrates focus and context, which allows users to zoom in on details while maintaining the entire dataset overview. This is particularly useful when dealing with large, complex datasets where users explore global patterns and local information at the same time (Bjrk and Holmquist, 2002). However, this is a challenge to traditional approaches; while the other models are robust for design, they do not explicitly address the effective maintenance of balance between global context and local information in their final visualization stage (Munzner 2009).

ii. The blur fisheye view approach is user-centric, enabling users to interact with visualization more intuitively through smooth transitions between overview and details (Turetken and Schuff, 2002). However, this feature is less pronounced in the other models due to their structure, process-oriented approach in Sedlmair and Munzner, and action-oriented scope in McCurdy (Turetken and Schuff, 2002).

iii. Blurfisheye view is well adapted for addressing scalability problems, especially in complex domains such as biological networks where micro and macro analysis are critical. It is also dynamic for large dataset visualization without overwhelming the users. These are more challenging to achieve with the other models (Ali et al., 2016).

iv. McCurdy's adaptive nature and iterative process of the Sedlmair and Munzner models can be integrated into the Blurfisheye view to enhance visualization adaptability (McCurdy et al., 2016; Sedlmar et al., 2012). This is because it allows real-time changes based on user interaction, alignment with other methodologies, and stakeholders' inputs.

6.7.6. Corresponding Sedlmair DSM steps with Blurfisheye view approach.

The blur fisheye view design corresponds to the design and implementation stages in Sedlmair DSM. For example, in the design stage, the focus is to create visual encodings and interaction systems that best represent the data and meet the users' needs (Bertini et al., 2011). The Blurfisheye view design is a visual encoding and interaction technique addressing how the users can explore data. At this stage, the design team explores different ways to integrate the Blurfisheye view into the visualization to ensure it effectively balances the focus and contexts for the users (Wu and Chang, 2024). Furthermore, the design concept is turned into a working prototype at the implementation stage. The Blurfisheye view is technically implemented within the visualization tool at this same stage. It involves algorithm development and user interface components to enable dynamic zooming and detail-in-context features defining the blur fisheye view (Wu and Chang, 2024).

6.8. Evaluation of the Blurfisheye Visualization Tool

The evaluation was conducted using the data collection methods that are vital for researchers to obtain data from different sources to answer research questions, test hypotheses, and gain

insights into the concept under study. Other types of data collection techniques can be used in research.

Surveys: These are structured questionnaires used to collect quantitative or qualitative data from a pool of respondents. They can be administered in different formats, such as paper-based, online, telephone, and face-to-face surveys (Sajjad Kabir, 2016). One significant advantage of a study is that a large audience can be reached, standardised questions ensure consistency in the survey ensure consistency, and it is cost-effective for large samples (Sajjad Kabir, 2016). However, it has a limited depth of responses, the potential for low response rates, and if the questions are poorly designed, the response may be biased.

Interviews: This involves direct, face-to-face, telephone or online interaction where a researcher asks questions to the respondents, and they can be structured, unstructured or semi-structured. (DeJonckheere and Vaughn, 2019) Structured interviews follow a predetermined set of questions with few variations. Its significant advantages lie in the fact that it is easier to compare responses and efficient in collecting specific data. However, it has limited flexibility and may be unable to capture deep responses (DeJonckheere and Vaughn, 2019). Semi-structured uses a guide with essential questions but gives room for flexible responses. Its advantages are that it balances structure and flexibility and can explore subject matter deeply (Whiting, 2008). However, it consumes time. Unstructured interviews, on the other hand, have no predetermined questions, thus allowing a natural free flow of conversation. It captures rich and comprehensive information due to its high flexibility. However, it is difficult to compare responses, and this consumes time (DeJonckheere and Vaughn, 2019).

Focus Groups: These involve guided discussions with a small group of respondents led by a moderator to explore their perceptions, opinions, perspectives, and attitudes on some topics. Their advantage is that they generate rich data from group interactions, participants can build on each other's ideas, and they are effective (Nyumba et al., 2018). However, the dominant participant may skew the results, resulting in a biased outcome.

Observations: This system records behaviours, events, or conditions in natural settings. Observations can be participatory and non-participatory. The former is about the researcher actively engaging or getting involved in the setting, while the latter is about observing without interaction (Amerstorfer, 2021). Observation offers real-world context and can capture non-verbal behaviour. Hence, it is useful for behavioural and interactional studies. However, it can consume time, be prone to observer bias and may not deeply capture the intrinsic state of things

(Dewaele and MacIntyre, 2016). The data required for this evaluation were collected using survey and interview techniques to gain information on users' usability evaluation and testing of the tool, using different parameters, which are described below.

6.8.3. Evaluation Process

Collaboration with experts

Early in the development stage, the researcher collaborated with two bioinformaticians from Newcastle University with extensive experience in using visualization tools for biological networks to identify the right technique and design that would meet users' needs. This involved comprehensive discussions about existing visualization tools for biological networks, and the researcher also presented a preliminary version of the prototype to them, introducing every feature that was included and sharing ideas for beneficial aspects that had not been considered. Furthermore, the discussion touched on the user-friendliness of the biological network, particularly the ease of arranging nodes to form desired patterns and the ease of deducing new patterns from existing patterns, among other things. The researcher noted the outcomes of the productive discussion and made the required changes or amendments to the prototype design. One of the key suggestions to improve the prototype was the addition of further functionalities, such as providing detailed information about the uploaded networks, total nodes and edges, rendered nodes and edges, and network type in the toolbar. However, this was more of an informal discussion with no timeline or method of evaluation. Mainly, it was an opportunity to extract information that could be used as a guide to ensure the technology meets the experts 'needs.

Usability evaluation (survey)

To enhance the usability of the Blurfisheye tool, a heuristic usability evaluation, as well as Insight, Confidence, Essence, and Time (ICET) heuristics, were conducted (Forsell & Johansson, 2010). This was preferred to the classical heuristic in this study because this new set was developed by Forsell and Johansson (2010) specifically to assess serious issues with the usability of the InfoVis technique (Nielsen, 1994). Furthermore, Forsell and Johansson (2010) discussed 10 out of 63 heuristic factors as the best practices; that is, the 10 heuristics are derived from the 63 previously published heuristic sets, whose numbering has letters that refer to the original heuristic in combination with their original number. Six evaluators were contacted to seek their consent to participate in the study. Since three to five evaluators are considered a benchmark for heuristic evaluation, the researcher worked with the six who gave

their consent, indicating a comprehensive and thorough study. The survey questions were carved in line with previous work from the literature. There were three sections; the first section, also known as Section A, collected demographic information of the six evaluators, while Section B attempted to seek the knowledge and experience of the evaluators on visualization tools. The questions were five in number, all culled from the literature. Then, section C, which was about requirements of network visualization tools and divided into two parts, general factors and heuristic factors, was designed using the 5-Likerts scale of Strongly Agree, Agree, Neutral, Disagree, and Strongly Disagree responses. Both general and heuristic factors had three to five questions per variable. The study was conducted face-to-face with six evaluators, more than the number that is considered sufficient for a heuristic evaluation. According to Holzinger (2005), three to five evaluators are generally considered sufficient for heuristic evaluation, which the current study surpassed, indicating a comprehensive and thorough study. In this study, the evaluators (three males, two females, and one other/preferred not to) had computer science degrees and extensive practical and theoretical knowledge and experience related to data visualization. They included a research associate, lecturer, graph practitioner, scientist, PhD students, and a postdoc. Their expertise in visualization tools ranged from one to three, four to five, and six to ten years, with some having more than ten years of experience. The age of the experts ranged from 25 to 55 years of age.

The testing session was done separately to ensure that there was no evaluator bias. It is important to note that the assessors were permitted to inquire about the subject matter to get future clarity. Each of the evaluators tested the tool two times. The first time was to achieve familiarity with the general scope of the tool, while the second time was to focus on the visual and interactive interface elements concerning the available heuristic lists. Two additional pilot studies were conducted before the main study to refine the study. Both test participants were PhD students researching visualization in the School of Computing at Newcastle University.

User testing (interview)

Usability evaluation is a more general term for the process of assessing the usability of a tool, and it includes methods like heuristic evaluation. However, user testing is a more specific type of usability evaluation that involves the observation of users as they interact with the tool. The aim of user testing in this study was to derive valuable insights about the usability of the tool and identify possible areas where improvement might be needed.

Various tools and software programmes are commonly utilised to analyse and visualise biological data in biological networks. Many of these tools have made essential contributions

to visualising biological data in biological networks. However, because biological networks are intricate and involve different types of data like protein interactions and gene expression, existing tools may not comprehensively and effectively cover all the parts of the diverse data types. Furthermore, as biological datasets increase in complexity and size, the existing tools may not be able to efficiently manage large-scale networks. Therefore, researchers require tools that can seamlessly scale this hurdle and accommodate big data. One such tool is Blurfisheye, which is designed to help researchers understand complex interactions in a biological system, such as gene regulatory network PPIs. This study included a biomedical scientist, a genetic and molecular biology and immunology specialist, a laboratory researcher, and a lecturer in molecular biology and cell biology, with experience ranging from 6 to 10 years to more than 10 years. The participants took control of the tool using the Microsoft Teams feature. Microsoft Teams was selected for this purpose because it has a sharing feature that allows other session participants to use their cursor and interact with the host machine. The participants were asked to test the tool using their datasets, and they worked on this for three to seven days. After a week, when the participants had interacted with the tool, the researcher conducted semi-structured interviews with the participants, which involved a mixture of open and closed-ended questions. To ensure that the interview details were fully captured, the interview was recorded and later transcribed into text. Furthermore, the participants reviewed the transcript to ensure their agreement with the interpretation of what was said prior to the analysis. There were also discussions after the participants had a chance to test the tool and provide feedback. Each interview lasted between 60 and 75 minutes.

6.8.4. The result of the usability evaluation (survey)

A usability evaluation was conducted by analysing the participants' responses based on different variables, including information coding, flexibility, orientation and help, and minimal actions. Tables 51-62 of Appendix E present the results of the data visualization specialists' analysis (evaluation) of the usability of the developed Blurfisheye visualization tool.

Tables 51-64 in Appendix E show four components and the constituent heuristics for each component. Tables 51-64 also show the summary (average) ratings for visualising each heuristic and the standard deviation of each rating. Starting with the tool's information coding, the mean rating is 4.4, as seen in Table 51, indicating that the tool's layer is easy to use to differentiate nodes. Also, it is easy to use different line types and colours to represent edges, and at the same time, it is easy to use different colours to represent node clusters. Regarding the tool's flexibility, the mean rating is 5 (Table 52, Appendix E). This suggests that the tool

is highly flexible, making it easy to arrange the node, enlarge a node to get more details, and import datasets into the systems. Regarding orientation and help, the overall mean value is 4.33 (Table 53, Appendix E). This implies that it is easy to understand how to change the toolset using the panel on the right. The overall mean value for minimal actions is 4.7 (Table 54, Appendix E), indicating that it is very easy to fix inconsistencies in the features of the tools, such as colouring. For prompting, the overall mean value is 4.3 (Table 55, Appendix E), which means that it is easy to understand when certain actions are being taken by the tool. In terms of tool consistency, the overall mean value is 4.2 (Table 56 Appendix E), indicating the ease of fixing and finding inconsistencies and predicting the time it will take to render a dataset in the tool. The overall mean value of 4.8 for spatial organisation implies that it is easy to identify the differences in the nodes using the fisheye view (Table 57, Appendix E). Furthermore, as shown in Table 58 in Appendix E, the overall mean value for recognition rather than recall, another heuristic factor, is 4.6, which shows that the tool is more accessible to recognition than to recall. In other words, it is easy to identify the similarities and differences between datasets when there are changes. In terms of removing the extraneous, the overall mean value is 4 (Table 59, Appendix E), indicating a high level of ease in removing extraneous information, making it easy to find unwanted nodes and remove parts of the tools that are not relevant. Similarly, as shown in Table 60 in Appendix E, the overall mean value for dataset reduction is 4.3, indicating that it is easy to render parts of a large dataset, find irrelevant elements in the dataset, and partition a dataset into several actions. The overall mean value for time is 4.2 (Table 61, Appendix E). This indicates that the time taken to render a large data set is impressive, and it takes a short time to find patterns in a dataset, so a short time is needed to remove unwanted features using the tool. The overall mean value for insight is 4.2 (Table 62, Appendix E). This indicates that it is easy to judge a dataset's accuracy, find faults, and find inspirations when visualised using the blur fisheye tool. The overall mean value for essence is 4.2 (Table 63, Appendix E). This indicates that it is easy to make sense of random values, easy to reach a conclusion from visualised data and easy to get the message in a dataset. The overall mean confidence value is 4.3 (Table 64, Appendix E), which implies a significant confidence level in the tool.

From the analysis above, the lowest overall mean value is 4 (Table 59, Appendix E), while the highest is 5 (Table 52, Appendix E), which indicates that the tool is highly usable, considering the different heuristic factors.

6.8.5. User testing (interview) results

The user testing evaluation was carried out by analysing the participants' responses using heuristic and ICET factors. Tables 65-78 of Appendix F present the results of user testing of the developed Blurfisheye visualization tool based on discussions with a biomedical scientist, a genetic, molecular, and immunology specialist, a laboratory researcher, and a lecturer in molecular biology and cell biology.

Regarding the information coding, the first question posed was how relevant the visualization outcome was to the users' needs. It was found that the application provided users with good visualizations due to its flexibility and easy tunability. Some of the challenges faced in visualising a dataset include web lagging, the clunkiness of the interface, a lack of understanding of what the node represented, and node dragging, which slows down the web. Thus, regardless of the relevance of the tool, some inherent challenges can impede its functionality. All four layout algorithms, grid, Breadthfirst, goose, and circle, conveyed the information about a dataset (see Table 65, Appendix F).

In terms of flexibility, the extent to which zooming and panning were utilised shows that those two features were a nice addition to the visualization tool. However, the smaller amount of white space makes panning a bit more difficult. Additionally, the number of nodes visualised in the dataset depended on the selected radius. However, numbers like 1,986, 3,000 or 19,623 could be derived. No adjustments were made to the visualised network after rendering because of the web loading speed and the lag problem (see Table 66, Appendix F).

Regarding orientation and tool help, it was straightforward to understand the information displayed when the help button was clicked and that the dataset had to be a TXT file before it could be uploaded. An information prompt and layout explanation should be included in the help instructions (see Table 67, Appendix F).

In terms of minimal actions, it was not challenging to achieve the fisheye view in the way the participants preferred, but it was hard to change the look of the visualised network. Hence, there is a need for an undo button to make minimal actions easy and add more sensitivity to the range radius. Further, when the user tried to change the look, there was a slow-down, but a minor adjustment would correct this issue. (see Table 68, Appendix F).

Regarding prompting, the most relevant prompts in the system were the cluster of relevant genes and nodes and edge highlights. These are relevant because they relate to other clusters more clearly. However, concentric circles are not necessary in the system. It does not appear

there is a need to add any additional prompts. Should such a need arise, the ability to add notes and groups would likely suffice (see Table 69, Appendix F).

No inconsistencies were identified in the tool. Although there are differences in the rendering time of datasets, the datasets can be enhanced to minimise the differences. Further, the visualization of the datasets was consistent with what was expected (see Table 70, Appendix F).

In terms of spatial organisation, the layout algorithms where the fisheye view is most useful include grid, goose, and Breadthfirst. CoSE is the layout algorithm that produces the most visually appealing renderings and can arrange the nodes in a way that suits users' purposes (see Table 71, Appendix F).

Regarding recognition rather than recall factor, it is possible for node clustering and the analysis of the visualization's concentric and data parts to have gone unnoticed if not for the visual cue and spatial arrangement. Additionally, information recognition is more efficient than information recall. The tool helped in this area by providing useful recognition and remembering data. (see Table 72, Appendix F).

In terms of removing the extraneous, there was no part of the tool that the participants wanted to remove but could not; there were no unnecessary features that served no real purpose. Moreover, it was easy to find unnecessary elements in the dataset (see Table 73, Appendix F).

Regarding dataset reduction, it is unnecessary to render datasets in small chunks instead of all at once. Further, the tool was used to partition a dataset into sections. Some features that could make it easier to reduce datasets include feature snipping and easy deletion of uninteresting nodes (see Table 74, Appendix F). While the tool is not overly time-consuming, it took more than 30 seconds to render the largest dataset, which is not good enough. Loading the dataset, changing the layout, and rendering are the functions that took the most time (see Table 75, Appendix F).

In terms of insights, the cluster of related genes and PPI for heart disease treatment were identified using the visualised network. A cluster of essential nodes was the new pattern that emerged from all of the visualised networks. It was possible to explore the relationship between different genes when selected from the drop-down list using the Blurfisheye visualization tool, as shown in Table 76 in Appendix F.

Regarding essence, no special meaning existed in any of the visualised datasets. However, adding interaction capabilities, a way to analyse specific edges, and side-by-side layouts could make it easier for users to find meaning in datasets. (see Table 77, Appendix F).

Finally, in terms of confidence, it is rational to make confident decisions based on the visualization of a dataset because of the identity clustering and clarity of the dataset. Indeed, binding decisions can be made based on the visualization of the dataset because all interactions are clear. Improving the programme based on users' needs, providing easy data accessibility on edges, and allowing users to redownload specific views of interest could increase users' confidence in the visualizations produced by this tool (see Table 78, Appendix F).

6.8.6. Summary of the evaluation

The study evaluated Blurfisheye, a tool supporting the exploratory analysis of large and complex biological networks. The tool was evaluated using a set of evaluation heuristics and participant feedback. Blurfisheye received mostly positive feedback, and experts achieved good results using the tool, such as identifying targets for drug discovery and clusters/areas that might be of interest for future research. In addition, two potential proteins for heart disease treatment were discovered. From an educational perspective, it could be used to teach students about biological networks. Most importantly, the tool's primary goal of identifying nodes of interest and then analysing them using the various available views was met. The feedback indicated that only minor changes were needed to the tool and that the technique itself was sound. Most of the changes were minor, such as adding new toolbar functions like snipping and notes.

6.9. Updates to the Blurfisheye Visualization Tool

Following the evaluation and user testing, a number of updates were made to the Blurfisheye visualization tool, including highlighting nodes with a specific number of interactions, adding notes, and UI blockers for intense operation, among others. The details of the update are therefore presented as follows:

1. Highlighting nodes with a specific number of interactions

Users can filter nodes with a certain number of interactions. The user can set the number of interactions using the range selector or the text field in the 'number of interactions' section. The number of interactions represents the minimum number of edges that should be connected to the selected nodes. In Figure 54, the four nodes have a minimum of 50 interactions.

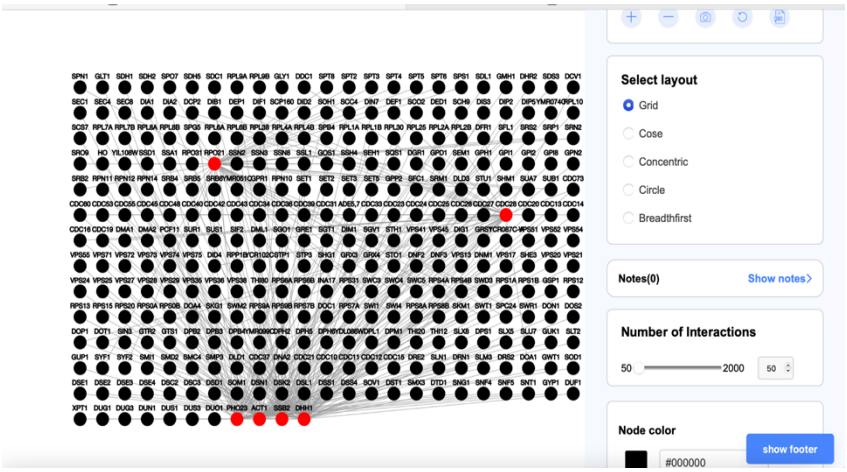


Figure 59:Nodes with specific instructions

2. Adding notes

Users can take notes as they use the tool. The notes are saved in the local storage of the user's browser so that the user can access them anytime they open the tool. The view for adding notes is shown in Figure 59.

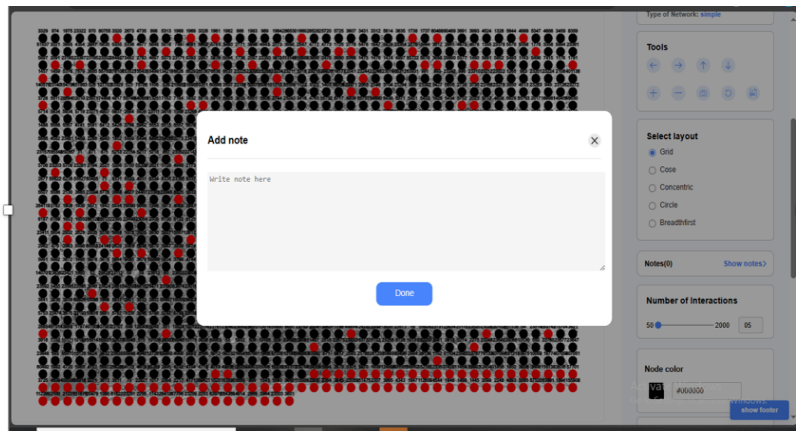


Figure 60: Adding a note

A user can edit an existing note by clicking the edit button in the button list beside the name of the note, as shown in Figure 60. This will open the edit text area, where the user can change the existing text, as shown in Figures 61-62.

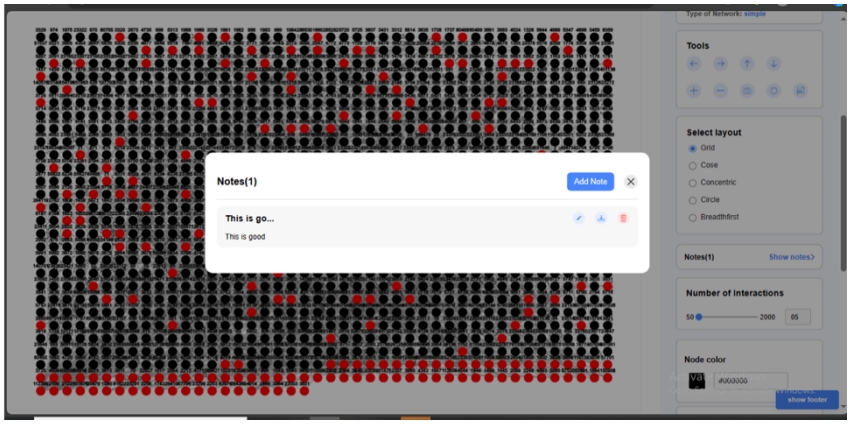


Figure 61: Edit note 1

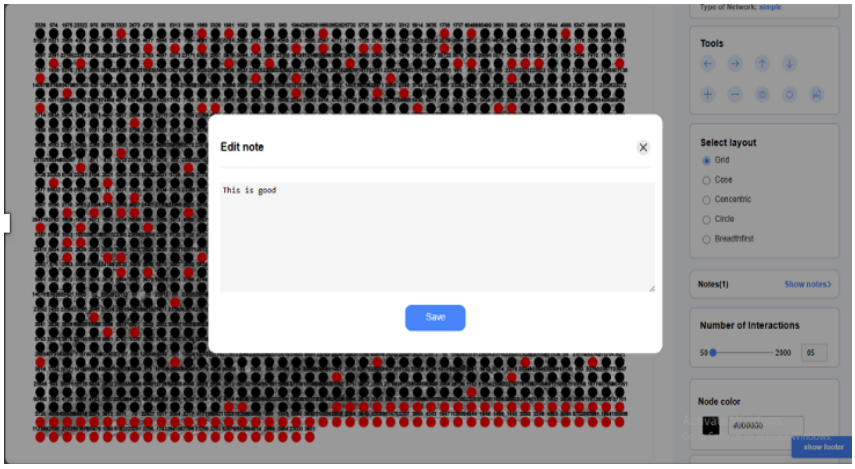


Figure 62:Edit note 2

Users can delete an existing note by clicking the trash button beside the note's name. The user will be prompted to confirm that they want to delete the note, as shown in Figure 63.

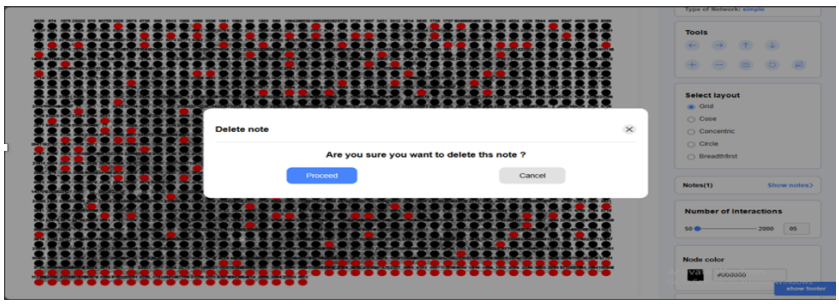


Figure 63:Deleting the note

A user can download an existing note by clicking the download icon beside the note's name. This will automatically trigger the download of the note in TXT format, as shown in Figure 64.

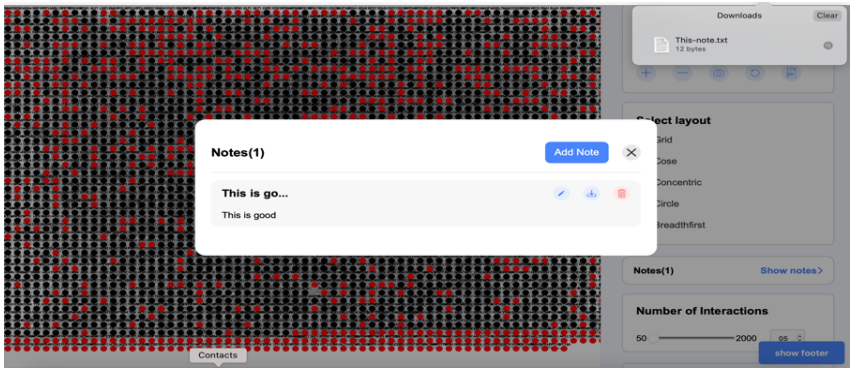


Figure 64:Downloading note

3. UI blockers for intense operations

During intense operations, such as file parsing and node rendering, it is possible for the UI to freeze momentarily, and further interactions by the user could exacerbate the problem and further slowdown the operation. Hence, a UI blocker was introduced to improve performance, as shown in Figure 65.

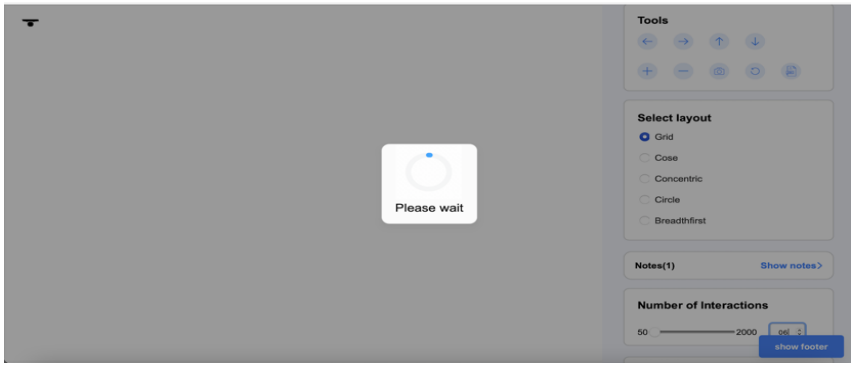


Figure 65:UI Blocker

4. Taking and downloading screenshots

A user can take a screenshot of the graph by clicking the camera icon in the 'Tools' section of the toolbar. This will trigger the download of the screenshot of the full graph, as shown in Figure 66 below.

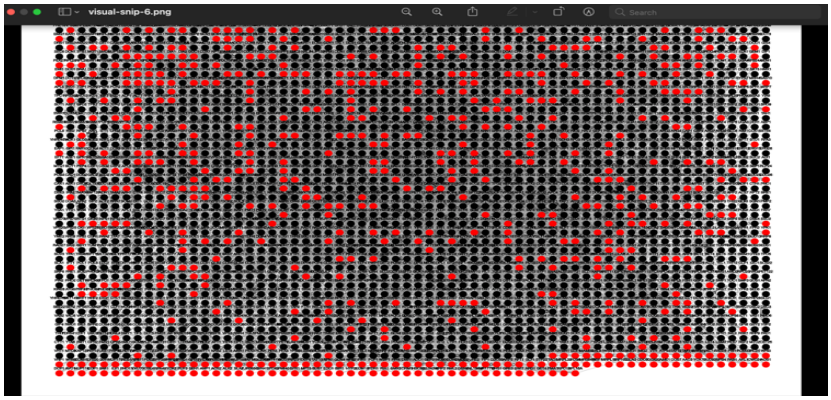


Figure 66:Screenshot

5. Undo changes

Users can undo some changes that they have made to the visualization by clicking the undo button in the 'Tools' section of the toolbar (Figure 66). Some actions that can undone are colour changes, zoom changes, and pan changes.

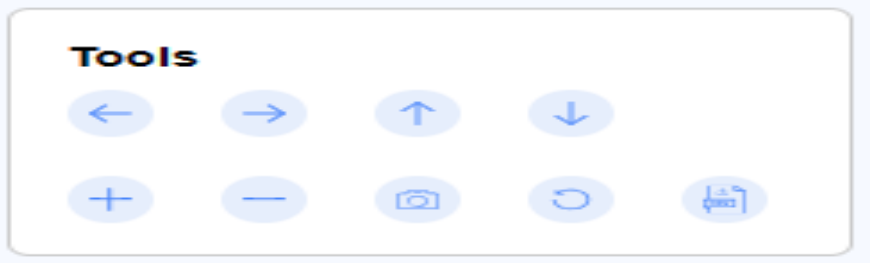


Figure 67:Undo feature

6. Zoom feature

Users can zoom in and out of the visualization area to better view the nodes, as shown in Figure 68. This can be done by clicking the '+' or '-' icon in the 'Tools' section of the toolbar. Clicking the '+' zooms in while clicking the '-' zooms out.

Figure 68:Zoomed in

7. Panning feature

The tool gives users the ability to pan the visualization without touching it. There are instances where touching the visualization could alter it, resulting in the need for panning buttons. These buttons (arrows) can be found in the 'Tools' section of the toolbar. Figure 69 shows a visualization that has been panned to the right and downward.

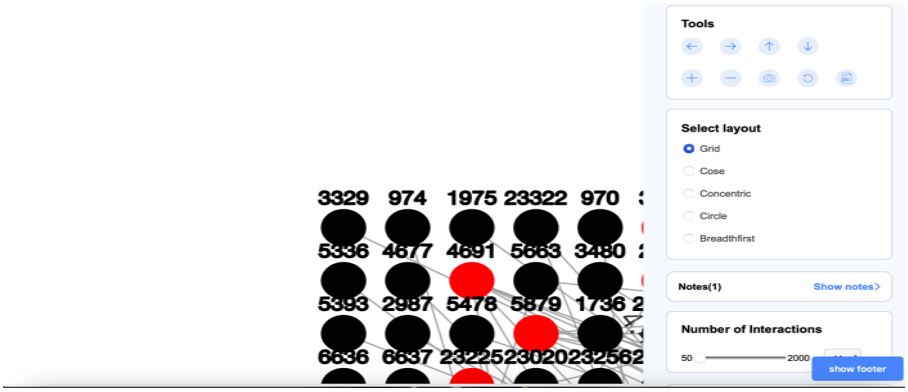


Figure 69:Panned view

8. Tutorials and user guide

The tutorial and user guide were updated in version 2 to provide users with information about the tool's main purpose as well as detailed instructions on how to use each feature (Figure 70).

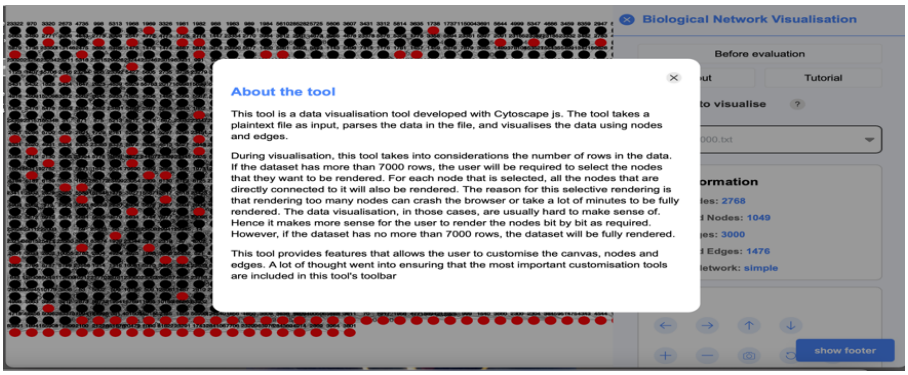


Figure 70:Tutorials and user guide view

6.10. Comparison of the enhanced visualization tool with the existing tools.

The latest tool, utilizing fisheye view and blur methods in the Cytoscape framework, is a major improvement compared to other network visualization tools like Medusa, Cytoscape, Graphia, Arena 3D, and Gephi. Although traditional tools excel in filtering, plugin support, visualization styles, and scalability, the new tool improves user interaction and data clarity with innovative visual methods. The ability to zoom in with a fisheye view lets users concentrate on individual nodes and edges while still keeping track of the entire network, making navigation more intuitive. The blurring technique also helps differentiate important areas by lessening the visual emphasis on less relevant parts of the network.

Regarding filtering, the new tool combines enhanced functionalities to enhance the strong features of Cytoscape and Graphia, resulting in more accurate data manipulation and exploration. The core strength of plugin compatibility allows users to seamlessly extend functionality. The visual appearances are greatly enhanced by blending Graphia and Arena 3D's detailed 2D/3D rendering with innovative, dynamic focus methods, leading to more transparent and informative network displays.

Scalability and performance have been improved to efficiently manage large datasets, similar to Gephi's abilities, but with additional visual enhancements that ensure clarity in intricate networks. The user-friendly interface of the new tool can be customized extensively. It includes advanced search functions and feedback features to facilitate in-depth analysis and exploration.

In general, this innovative tool excels by combining advanced visualization techniques with strong analytical capabilities, establishing a new benchmark for network visualization and analysis. A detailed comparison of the new tool introduced in this study with existing tools, as evaluated in Chapter 5, for the general factors is provided in table 27.

Table 27. The detailed comparison of the new tool introduced in this study with existing tools evaluated in chapter 5 for the general factors

No	**Factors**	**Medusa**	**Cytoscape**	**Graphia**	**Arena 3D**	**Gephi**	**New Tool (Cytoscape with Fisheye View and Blur)**
1	Filtering Tools	Moderate filtering, regex support	Rich filtration tools support columns, degrees, etc.	Node filtering, focus on filtered nodes	Rich filtering tools show/hide various elements	Direct node/connection selection	Advanced filtering with enhanced node focus using fisheye and blur
2	Plugins	No plugins	Extensive plugin support	Highly extensible with plugins	No user-defined plugins	Supports Java-based plugins	Same extensive plugin support as Cytoscape
3	Visual Styles	2D does not fit the	2D, high-level interaction	Optimized for large graphs,	3D rendering,	3D real-time rendering	Enhanced 2D/3D visualization

		screen by default	representations	2D/3D views	holistic view		with fisheye and blur techniques
4	Advanced Search	Regex-based search	Supports edges, nodes, attributes, and web search	Node search, graph centring on nodes	Direct/indirect nodes, connections, layers search	Limited search, direct selection	Advanced search with enhanced focus on nodes/edges with fisheye and blur
5	Free/Open-Source	Free for academic use	Free and open-source	Open-source and free	Free for academic use	Free and open-source	Free and open-source
6	Efficient Layout Algorithms	Multiple layouts, visual animation	Hierarchical, circular, force-directed layouts	Force-directed by default, other plugins available	Multi-layer, circular, random grid layouts	Various layouts, including ForceAtlas and Yifan Hu	Multiple layouts with enhanced visualization using fisheye and blur techniques
7	Scalability	Scales well with large networks	Moderate capability for large datasets	Highly scalable, handles millions of edges	Increases complexity with network size	Optimized for large datasets	High scalability with enhanced performance for large networks
8	Different File Formats	Supports CSV, TXT, TSV	Supports NDex, CSV, TSB, URL, databases	Supports TXT, CSV, TSV, GraphML	Single text file with various info	Supports CSV, GDF, GML, GEXF, Pajek NET, Graphviz Dot	Supports the same formats as Cytoscape, with additional visualization enhancements
9	Text Mining	Limited, regex-based search	Advanced text mining via	Efficient search tool, limited	Efficient search tool, limited text mining	Semantic network analysis, stemming,	Advanced text mining with enhanced

			StringApp plugin	default text mining		lemmatizatio n	node focus using fisheye and blur
10	User Input & Customizati on	Limited customizati on	Highly customizabl e and extendable	Highly customiza ble and extendable	Limited customizati on	Highly customizable and extendable	Highly customizable , with enhanced visualization options using fisheye and blur techniques
11	Graph Analysis	Supports various clustering methods	Rich graph analysis via plugins like Network Analyser	Various graph analyses, including clustering	Limited graph analysis	Generates summary statistics	Comprehensi ve graph analysis with enhanced focus using fisheye and blur
12	Feedback to Users	Export to multiple formats	Show status tool, export to NDex, web pages, images	Tooltips, alert boxes for process updates	Export to jpeg, Pajek, Medusa, VRML	Tooltips, export as image or PDF	Tooltips and alerts with enhanced user feedback using fisheye and blur techniques
13	Strength	Simple visualizatio ns, data type conversion	Advanced visualization , flexible display	Optimized for large networks, high-quality 3D rendering	Powerful 3D visualizatio n, node clustering	Multi-task architecture visualizes large, complex networks	Enhanced visualization and interaction using fisheye and blur techniques
14	Runtime Performanc e	Fairly quick	Fairly slow	Very fast	Slow, especially for large networks	Fast layout rendering	Fast performance with enhanced visualization techniques

							for large networks
15	User-friendliness	Simple interface	Customizable UI, tooltips	Very user-friendly, offline tutorials	Limited UI flexibility, no tooltips	Nice UI, draggable, zoomable, resizable visuals	User-friendly interface with advanced interaction using fisheye and blur techniques

The integration of fisheye view and blur techniques in the Cytoscape framework results in a breakthrough in network visualization and analysis. Contrary to current tools like Medusa, Graphia, Arena 3D, and Gephi, which excel in information coding, flexibility, and spatial organization, the new tool enhances user engagement and data visibility. The fisheye view lets users concentrate on particular nodes and edges while maintaining awareness of the overall network, and the blur technique decreases the visual importance of less significant parts of the network, enhancing navigation ease. Improved filtering features enhance data handling while plugin support remains strong. The tool's latest feature guarantees scalability and performance, efficiently managing large datasets with dynamic visual improvements. Comprehensive tutorials, tooltips, and advanced search functions optimize user customization and feedback. Overall, this new tool establishes a higher benchmark through the integration of advanced visualization methods alongside strong analytical capabilities to provide clearer and more elaborate network insights. The detailed comparison of the new tool introduced in this study with existing tools evaluated in chapter 5 for the heuristic factors is provided in table 28.

Table 28. The detailed comparison of the new tool introduced in this study with existing tools evaluated in chapter 5 for the heuristic factors

No	**Factors**	**Medusa**	**Cytoscape**	**Graphia**	**Arena 3D**	**Gephi**	**New Tool (Cytoscape with Fisheye View and Blur)**
1	Information Coding	Displays multiple edges using Bezier	Converts expression data into	Provides top-notch 3D graph representatio	Effectively converts network data into	Facilitates clustering, spatialising, navigating,	Enhances information coding with fisheye and

		curves. Supports predefined clustering.	visual attributes.	ns and overview mode.	3D visual information, representing multiple layers for readability.	and manipulating networks.	blur techniques, improving focus and readability of complex networks.
2	Flexibility	The interface is not easily customizable.	A highly adaptable interface supports different data types and command line tools.	Highly flexible with multiple plugin support and customizable interface.	The interface is not adaptable, with limited zoom and immovable panes.	Customizable interface with resizable and minimizable panes.	Highly flexible, combining Cytoscape's adaptability with advanced visualization features for customized user experiences.
3	Orientation and Help	It provides a tutorial page and tooltips/icons for guidance.	Rich user manual and tutorial page with tooltips.	Offers offline and online tutorials and informative websites.	No help tab or tooltips and outdated website information.	Online documentation and tutorials are available.	Comprehensive orientation with updated tutorials, tooltips, and advanced visual guides using fisheye and blur techniques.
4	Minimal Actions	A few actions are needed to load and visualize the network.	Many steps are required for loading data and styling.	Several steps to load CSV and TSV files.	Few actions are needed to visualize the network after data loading.	Many steps are required to read data and run layout algorithms.	Streamlined actions are needed for network loading and visualization, with an enhanced focus on specific areas using advanced techniques.
5	Prompting	Uses tooltips and icons for guidance.	Provides tooltips and alert boxes.	Well, it prompts users with tooltips and prompt boxes.	No prompting is available, risking unintended user actions.	Uses prompt windows and buttons with tooltips for tasks.	Advanced prompting with interactive tooltips and alerts enhanced by fisheye and blur techniques

							for better user guidance.
6	Consistency	Naming conventions and file formats align with conventional tools.	Consistent notations with conventional ones.	Consistent notations and well-named objects.	Different data input formats do not accept traditional CSV/TSV files or databases.	Consistent notations with general usage (e.g., edges and nodes).	Maintains consistent notations and file formats, enhancing with dynamic visual focus techniques for improved data interpretation.
7	Spatial Organisation	Effectively organizes clusters but requires zooming.	Well-organized network graphs are detachable to new windows.	Highly organized with intuitive node arrangement.	Well-represented 3D network with predefined clusters visualized easily.	Nicely arranged nodes with resizable attributes.	Enhanced spatial organization using fisheye and blur, allowing better focus and navigation of large networks.
8	Recognition Rather than Recall	Tools and tabs are easily understandable and expected.	Easily recognizable tools without needing tooltips.	Conventional icons are used for easy recognition.	The usage of tools is easily understood with expected names.	Tools and tabs are easily recognized based on names and icons.	Enhances recognition with interactive visual cues, reducing the need for recall and making user navigation more intuitive.
9	Remove the Extraneous	Simple layout with maximized visualization space.	Extraneous panes can be minimized.	Simple UI without distracting panes.	Simple UI without extraneous panes. Network visualization is undistracted.	Extraneous panes are present but minimizable.	The streamlined interface minimizes distractions, leveraging fisheye and blur techniques for a cleaner, focused visualization.
10	Dataset Reduction	Nodes can be selected, viewed, and	Data reduction and table	It supports data reduction	Data is easily reduced for	Data is reduced by selecting	Advanced data reduction

		deleted easily.	loading capabilities .	with tools like %-NN, edge reduction, and k-NN.	specific nodes, connection s, or layer visualizatio n.	nodes or connections of interest.	with enhanced focus on specific network areas using fisheye and blur techniques for better clarity.

6.11. Summary

This section has provided background regarding the focus and context of visualization and examined different visualization techniques, including zooming and panning, overview + details, treemaps, heatmaps, focus + context cartograms, semantic zooming and fisheye view. This study focused on the fisheye view because it is valuable for visualising complex biological networks, particularly in terms of data preservation, clutter reduction, and meeting specific research needs. The second part of this chapter focused on the fisheye view, including its application, usability, benefits and drawbacks, technology, and architecture, including HTML, Papa Parse, and CSS.

Then, the fisheye view was used in the visualization of complex biological networks. The design and implementation of the Blurfisheye visualization tool were described, particularly its dataset visualization and all nodes in a simple dataset. The Blurfisheye tool was then evaluated, which involved collaborating with domain experts from the beginning of the design process. The data required for this evaluation were collected using survey and interview techniques to gain information regarding the usability of the tool and get users' feedback. Areas for improvement were identified, including the ability to highlight nodes with a specific number of interactions, note additions, and UI blockers for intense operations. In terms of research contribution, the detailed exploration of the fisheye view presented here can provide researchers with insights into how to better implement and use the technique to overcome the current limitations when visualising complex biological networks. Furthermore, researchers can understand the benefits of data preservation and clutter reduction. Also, the evaluation of Blurfisheye in this chapter has contributed to research through the validation of its effectiveness, uncovering potential therapeutic proteins and provision of valuable educative information to comprehend biological networks.

The fisheye view is the selected visualization tools that overcame current limitations in the five visualization tools evaluated in objective 3 and support human cognition and data exploration, using multiple coordinated views and interactivity guided by interestingness metrics. Section 6.2, particularly subsection 6.2.3, answers the research question 4.1. Also, section 6.3 answers research question 4.2. These answers, therefore, do justice to objective 4.

After the evaluation of the designed Fisheye view visualization tool, it was upgraded to the Blurfisheye visualization tool based on the testing. Thus, section 6.5 answers research question 5.1 on the usability of the implemented tool. While subsections 6.6.1 to 6.6.3. answer the research question 5.2. these, therefore, did justice to objective 5.

Chapter 7: Conclusion and Future Works

7.1. Conclusion

The study reviewed different visualization tools for complex biological networks, including Cytoscape, Medusa, Osprey, ProViz, CN-Plot, MAPMAN, Graphia, Gephi, Arena3D, and CellNetVis. This helped me understand the strengths and weaknesses of each tool by comparing them in terms of open source, file formats, layouts, scalability, editing, system, and URL. This can help other researchers with comparative analysis and help them choose the most relevant project tools. Furthermore, the study identified and evaluated the general and heuristic factors to determine their importance for inclusion in complex biological network tools. The result of the mixed methods of qualitative and quantitative analysis of the evaluations identified crucial features, with particular emphasis on the role of filtering tools and the need for efficient and accessible plugins in graph visualization. Following that, these significant factors were used to evaluate the most common visualization tools for complex biological networks. This contributed to the visualization research field by providing a broad and systematic framework for visualization tool evaluations and providing insights and recommendations to improve future tool designs and development. Out of all the visualization tools, Medusa, Cytoscape, Graphia, Gephi and Arena3D contributed the most to the advancement of research by empowering the scientists with insights to visualise biological networks. Despite the usefulness of these five major visualization tools, there was a need for enhanced visualization tools because biological networks like protein-protein interaction (PPI) networks, gene regulatory networks, and metabolic pathways, among others, are inherently complex. Hence, the introduction of the fisheye view is useful for showing complex information, as specific areas of interest are highlighted while displaying the surrounding context as well. However, in the evaluation of the Blurfisheye visualization tool, some important features were lacking, hence their addition to the updated Blurfisheye visualization tool, which allowed the researcher to gain insights into how to implement and use the technique more effectively.

This study examines the usability of interestingness measures and interactive visualization with complex biological networks. Five objectives were formulated to measure the main aim of this work.

The first goal involved identifying tasks and patterns relevant to the analysis of biological networks based on a literature review and interviews/consultations with biologists. This encompassed aspects such as arranging nodes to form desired patterns, deducing new patterns

from existing patterns, gaining insights into dataset issues, node arrangement, enlarging a node to get more details, importing datasets, understanding panel settings, and selecting visualised nodes. Other aspects include focusing on nodes by zooming and panning, changing the fisheye radius, the appearance of nodes/edges, understanding when a dataset file is being processed and when it is complete, and recognising tool pitfalls. All these tasks and patterns contribute to the utility of biological networks as visualization tools.

The second objective was to define a set of interestingness metrics grouped into general factors and heuristic factors based on identified tasks and patterns. These metrics play a crucial role in data analysis and knowledge discovery, providing valuable tools for data scientists, researchers, and decision-makers in different fields. As big data evolves, it is crucial to define and refine these metrics to extract meaningful information and insights from large datasets; this is needed to make informed and data-backed decision-making.

The third objective was to evaluate the usability and limitations of existing network visualization methods for biological networks. These methods have advanced the understanding of intricate network structures. However, the evaluation highlighted the need for more intuitive, user-friendly, and efficient tools to make biological network analysis more accessible to researchers and practitioners. The study also revealed a need for more research and development in this area to address the limitations regarding data integration, scalability, and visualization of dynamic network behaviours.

The fourth objective focused on designing and developing a visualization tool that overcomes current limitations and supports human cognition and data exploration, using multiple coordinated views and interactivity guided by interestingness metrics. This tool contributes significantly to data analysis and decision-making, empowering users by presenting data in a more accessible and engaging way and guiding them based on the most relevant and valuable information. The tool developed in this study has the potential to yield new insights, which could help to drive innovation not only in the healthcare industry but also across many other sectors.

The fifth and final objective was to evaluate and upgrade the network visualization tool based on user testing. Rigorous testing and analysis supported the refinement, optimisation, and enhancement of the tool, with the aim of meeting users' expectations and evolving needs. This iterative approach demonstrates a commitment to quality, user satisfaction, and adaptability to evolving specifications, paving the way for a more comprehensive and user-friendly network.

7.2. Limitations of the Study

These are the two primary limitations of the study that should be addressed in future research. The developed tool should be tested for security and reliability before being adopted as an industry standard and made public for use by other researchers, developers, and practitioners. This would ensure future collaboration, knowledge sharing, and potential improvements based on experts' feedback.

The developed tool should be tested against other visualization tools to assess its usability. This should provide information about how well the tool works in terms of user interaction, efficiency, and effectiveness. It would also give the researcher the opportunity to highlight areas where the tool excels and where it could be improved.

7.3. Future Work

The following work is recommended in the future to extend this study:
A larger number of specialists should be interviewed to assess general and heuristic factors, as agreement among larger experts would increase confidence in the evaluation results. This would allow the researcher to identify common patterns and areas of agreement.

The developed tool should be tested for security and reliability to ensure that it meets industry standards and is suitable for public use. The code could be made available on GitHub to solicit public feedback and suggestions for future improvements to the tool.

The developed visualization tool should be compared to other tools to determine its usability. This is because comparative usability studies ensure that the developed tool is useful and relevant to users. This would help guide decisions on whether to use the tool, improve it, or iterate on it.

References

Adamson, K., Prion, S. (2013). Reliability: measuring internal consistency using Cronbach's α. J. Ecns; 12(1). Doi: 10.1016/j.ecns.2012.12.001.

Adu, J., Owusu, M. F., Martin-Yeboah, E., Pino Gavidia, L. A., Gyamfi, S. (2022). A discussion of some controversies in mixed methods research for emerging researchers. *Methodological Innovations*. https://doi.org/10.1177/20597991221123398

Ali, S. M., Gupta, N., Nayak, G. K., Lenka, R. K. (2016). Big data visualization: Tools and challenges. In 2016, the 2nd International Conference on Contemporary Computing and Informatics (IC3I), IEEE. 656-660

Ali, S., Gupta, N and Lenka, R. 2016. Big data visualization: tools and challenges. Available at>https://www.researchgate.net/publication/311920984_Big_Data_Visualization_Tools_and_Challenges>[Accessed 200824]

Aljadeff, J., Stern, M., Sharpee, T., (2015). Transition to chaos in random networks with cell-type-specific connectivity. Physical review letters, *114*(8), 88-101.

Amerstorfer, C. M. (2021). Student Perceptions of Academic Engagement and Student-Teacher Relationships in Problem-Based Learning. *Frontiers in Psychology*, *12*, 713057. https://doi.org/10.3389/fpsyg.2021.713057

Apuke, O. (2017). Quantitative research methods: A synopsis approach. Arabian Journal of Business and Management Review (Kuwait Chapter), 6(2), 40–47.

Arapakis, I., Barreda-Angeles, M. and Pereda-Baños, A., 2017. Interest as a proxy of engagement in news reading: Spectral and entropy analyses of EEG activity patterns. *IEEE Transactions on Affective Computing*, *10*(1), pp.100-114.

Arnoldi, J.F., Bideault, A., Loreau, M., Haegeman, B. (2018). How ecosystems recover from pulse perturbations: A theory of short-to long-term responses. Journal of theoretical biology, 43(6), 79-92.

Aspers, P., Corte, U. (2019). What is qualitative in qualitative research? Qualitative Sociology, 4(2), 45-56.

Bahari SF (2012) Qualitative versus quantitative research strategies: Contrasting epistemological and ontological assumptions. *Jurnal Teknologi* 52: 17–28.

Baitaluk, M., Sedova, M., Ray, A., Gupta, A. (2006). Biological Networks: Visualization and analysis tool for systems biology. Nucleic Acids Research, 34 (2), 466–471.

Balzer, M., Deussen, O. (2005). Voronopi treemaps. In Proceedings of the 2005 IEEE Symposium on Information Visualization. Minneapolis, MN, USA: INFOVIS. 49-56. doi: 10.1109/INFVIS.2005.1532128.

Bastian, M., Heymann, S., Jacomy, M. (2009). Gephi: An open-source software for exploring and manipulating networks. Proceedings of the International AAAI Conference on Web and Social Media, *3*(1), 361–362.

Batada, N. N. (2004). CNplot: Visualizing pre-clustered networks. Bioinformatics, *20*(9),1455–1456.

Batagelj, V., Mrvar, A. (1998). Pajek – A program for large network analysis. Connections, 21(2), 47–57.

Battista, G.D., Eades, P., Tamassia, R. and Tollis, I.G., (1998). Graph drawing: algorithms for the visualization of graphs. United States, Prentice Hall PTR. P.397

Behrisch, M., Blumenschein, M., Kim, N et al. 2018. Quality metrics for information visualization. Eurovis; 37(3),1-38

Bell, G. W., Lewitter, F. (2006). Visualising networks. Methods in Enzymology, 411, 408–421.

Benton, M.L., Abraham, A., LaBella, A.L., Abbot, P., Rokas, A. Capra, J.A. (2021). The influence of evolutionary history on human health and disease. Nature Reviews Genetics, *22*(5),.269-283.

Bertini, E., Lam, H and Isenberg, P. 2011. Empirical studies in information visualization; seven scenarios. Available at>https://www.researchgate.net/publication/51855553_Empirical_Studies_in_Information_Visualization_Seven_Scenarios>[Accessed 200824]

Bhatnagar, V., Al-Hegami, A.S. and Kumar, N., 2008. Novelty as a measure of interestingness in knowledge discovery. *International Journal of Computer and Information Engineering*, *2*(9), pp.3268-3273.

Bhowmick, S., Biswas, N., Kalita, P. C., Sorathia, K. (2023). Design and evaluation of AMAZE: A multi-finger approach to select small and distant objects in dense virtual environments. Displays, 80 (4), 102539. https://doi.org/10.1016/j.displa.2023.102539.

Biesecker LG. (2013). Hypothesis-generating research and predictive medicine. *Genome Res*Jul 01;23(7):1051–3. doi: 10.1101/gr.157826.113. http://genome.cshlp.org/cgi/pmidlookup?view=long&pmid=23817045 .

Bjrk, S and Holmquist, L. 2002. A framework for focus+context visualization. Available at> https://www.researchgate.net/publication/2894547_A_Framework_for_FocusContext_Visualization>[Accessed 200824]

Blumenthal, J., Megherbi, D.B Lussier, R. (2020). Unsupervised machine learning via Hidden Markov Models for accurate clustering of plant stress levels based on imaged chlorophyll fluorescence profiles & their rate of change in time. Computers and Electronics in Agriculture, 17(4), 1050-1064.

Bobi, A., Missaoui, R. and Ibrahim, M.H., 2023. Enhancing Actionable Formal Concept Identification with Base-Equivalent Conceptual-Relevance. *arXiv preprint arXiv:2312.14421*.

Boyle, J., Kreisberg, R., Bressler, R., Killcoyne, S. (2012). Methods for visual mining of genomic and proteomic data atlases. BMC Bioinformatics, 13(58). https://doi.org/10.1186/1471-2105-13-58.

Braun, V., Clarke, V. (2012). Thematic analysis. In Encyclopedia of Critical Psychology. Teo, T. (eds), New York, NY, Springer. https://doi.org/10.1007/978-1-4614-5583-7_311

Breitkreutz, B. J., Stark, C., Tyers, M. (2002). Osprey: A network visualization system. Genome Biology, 4(*3)*, 1–6.

Brodbeck, D., Mazza, R. and Lalanne, D. (2009). Interactive visualization survey. In Human Machine Interaction: Research Results of the MMI Program. Berlin, Heidelberg: Springer 27-46.

Caldarola, E. G., and Rinaldi, A. M. (2017). Big data visualization tools: A survey of the new paradigms, methodologies, and tools for large data sets visualization. In Proceedings of the 6th

International Conference on Data Science, Technology and Applications (DATA 2017). SCITEPRESS – Science and Technology Publications, Lda, 296–305.

Case guard, (2022). Can the human eye perceive how many frames per second? <https://caseguard.com/articles/how-many-frames-per-second-can-the-human-eye-see/#:~:text=The%20visual%20cues%20in%20the,to%2060%20frames%20per%20second.> [Accessed 24 June 2022].

Celotto, E., Ellero, A. and Ferretti, P., 2019. Asymmetry degree as a tool for comparing interestingness measures in decision making: the case of Bayesian confirmation measures. *Neural Advances in Processing Nonlinear Dynamic Signals 27*, pp.289-298.

Chen, L., Wang, R., Li, C., Aihara, K. (2010). Modeling biomolecular networks in cells: Structures and dynamics. London; Springer Science & Business Media. P.343. https://doi.org/10.1007/978-1-84996-214-8

Cheong, S.H., Si, Y.W. (2020). Force-directed algorithms for schematic drawings and placement: A survey. Information Visualization, *19*(1), 65-91.

Cline, M. S., Smoot, M., Cerami, E., Kuchinsky, A., Landys, N., Workman, C., et al. (2007).: Integration of biological networks and gene expression data using Cytoscape. Nat Protoc, *2*(10), 2366-82. Doi: 10.1038/nprot.2007.324.

Cockburn, A., Karlson, A., Bederson, B. (2008). A review of overview+detail, zooming and focus+context interfaces. ACM Computing Surveys, *41*:1. https://doi.org/10.1145/1456650.1456652.

Coyte, K.Z., Schluter, J. and Foster, K.R., (2015). The ecology of the microbiome: networks, competition, and stability. Science, *350*(6261), 663-666.

Cytoscape, (2022). 10. Finding and Filtering Nodes and Edges — Cytoscape User Manual 3.9.1 documentation.<https://manual.cytoscape.org/en/latest/Finding_and_Filtering_Nodes_and_Edges.html#narrowing-filters> [Accessed 17 June 2022].

Dawadi, S., Shrestha, S., Giri, R. A. (2021). Mixed-methods research: A discussion on its types, challenges, and criticisms. Journal of Studies in Education, 2(2), 25-36

Dayama, N.R., Todi, K., Saarelainen, T. and Oulasvirta, A., (2020), April. Grids: Interactive layout design with integer programming. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. 1-13.

De Gemmis, M., Lops, P., Semeraro, G. and Musto, C., 2015. An investigation on the serendipity problem in recommender systems. *Information Processing & Management*, *51*(5), pp.695-717.

De Paula, R. A. (2019). Visualization techniques. In the Encyclopaedia of Big Data Technologies, eds. S. Sakr and A. Y. Zomaya, Springer. https://doi.org/10.1007/978-3-319-77525-8_84.

DeJonckheere, M., Vaughn, L. M. (2019). Semistructured interviewing in primary care research: A balance of relationship and rigour. *Family Medicine and Community Health*, *7*(2). https://doi.org/10.1136/fmch-2018-000057.

DeSalle, R. (2018). Our senses*:* An immersive experience. New Haven: Yale University Press. https://doi.org/10.12987/9780300231649. P. 352

Dewaele, J. -M., MacIntyre, P. D. (2016). "Foreign language enjoyment and foreign language classroom anxiety: The right and left feet of the language learner," in *Positive psychology in SLA*, eds P. D. MacIntyre, T. Gregersen, and S. Mercer (Bristol: Multilingual Matters), 215–236. doi: 10.21832/9781783095360-010.

Doncheva, N.T., Morris, J.H., Holze, H., Kirsch, R., Nastou, K.C., Cuesta-Astroz, Y., (2022). Cytoscape stringApp 2.0: analysis and visualization of heterogeneous biological networks. Journal of Proteome Research, *22*(2),.637-646.

Dudzic-Gyurkovich, K. (2022). Study of Centrality Measures in the Network of Green Spaces in the City of Krakow. *Sustainability*, *15*(18), 13458. https://doi.org/10.3390/su151813458.

Dunwoodie, K., Macaulay, L., Newman, A. (2023). Qualitative interviewing in work and organisational psychology: Benefits, challenges and guidelines for researchers and reviewers. Applied Psychology, *72*(2), 863–889.

Dush, L. 2021. Drawing into being: charter graphics and their functions. Journal of Business and Technical Communication, 36(2), 165–189.

Dwivedi, D., Santos, A. L. D., Barnard, M. A., Crimmins, T. M., Malhotra, A., Rod, K. A., ... & Weintraub-Leff, S. (2022). Biogeosciences perspectives on integrated, coordinated, open, networked (ICON) science. *Earth and Space Science*, *9*(3), e2021EA002119.

Faysal, A. M., Arizfuzzaman, S. (2018). A comparative analysis of large-scale network visualization tool. Published in *IEEE International Conference on Big Data (Big Data)* 4837–4843.

Fetters, M. D., Curry, L. A., Creswell, J. W. (2013). Achieving Integration in Mixed Methods Designs—Principles and Practices. *Health Services Research*, *48*(6 Pt 2), 2134-2156. https://doi.org/10.1111/1475-6773.12117.

Forsell, C. Johansson, J. (2010). A heuristic set for evaluation in information visualization. In Proceedings of the International Conference on Advanced Visual Interfaces, AVI ’10. New York, USA, 199–206.

Franconeri, S. L., Padilla, L. M., Shah, P., Zacks, J. M., Hullman, J. (2021). The Science of Visual Data Communication: What Works. *Psychological Science in the Public Interest*. https://doi.org/10.1177/15291006211051956.

Franconeri, S. L., Padilla, L. M., Shah, P., Zacks, J. M., Hullman, J. (2021). The science of visual data communication: What works. Psychological Science in the Public Interest. 22(3), https://doi.org/10.1177/15291006211051956.

Freeman, T. C., Goldovsky, L., Brosch, M., Van Dongen, S., Mazière, P., Grocock, R. J.(2007). Construction, visualization, and clustering of transcription networks from microarray expression data. PLoS Computational Biology, *3*(10), 206.

Gansner, E.R., Koren, Y. (2006), September. Improved circular layouts. In International Symposium on Graph Drawing. Berlin, Heidelberg: Springer. 386-398.

Garima, M. Kiran, U.V. (2014). Impact of marital status on the mental health of working women. Journal of medical science and clinical research, *2*(10), 2594-2605.

Genially Blog, 2022. Could someone please explain what interaction is? <https://blog.genial.ly/en/what-is-interactivity/#:~:text=Interactivity%20refers%20to%20the%20communication,the%20person%20that’s%20using%20it.> [Accessed 24 June 2022].

Gerasch, A., Faber, D., Küntzer, J., Niermann, P., Kohlbacher, O., Lenhof, H. P., et al. (2014). BiNA: A visual analytics tool for biological network data. PloS One, *9*(2), 873-897.

Gibson, H., Faith, J., Vickers, P.2013. A survey of two-dimensional graph layout techniques for information visualization. Information visualization, *12*(3-4), 324-357.

Gigerenzer, G., Gaissmaier, W. (2011). Heuristic decision-making. Annual review of psychology, 62(1), 451-482.

Gilb, T and Brodie, L. 2005. Chapter 7: design ideas and design engineering. Available at>https://www.researchgate.net/publication/344939705_Chapter_7_Design_Ideas_and_Design_Engineering>[Accessed 200824]

Gillespie, T., Surles-Zeigler, M. C., Kokash, N., Grethe, J. S., Martone, M. (2022). Representing normal and abnormal physiology as routes of flow in ApiNATOMY. Frontiers in Physiology, 13 (795303). https://doi.org/10.3389/fphys.2022.795303.

Gkitsakis, D., Kaloudis, S., Mouselli, E., Peralta, V., Marcel, P. and Vassiliadis, P., 2024. Cube query interestingness: Novelty, relevance, peculiarity and surprise. *Information Systems*, p.102381.

Grilli, J., Barabás, G., Michalska-Smith, M.J. Allesina, S. (2017). Higher-order interactions stabilise dynamics in competitive network models. Nature, 548(7666), 210-213.

Grzybowski, A., Kupidura-Majewski, K. (2019). What is colour, and how is it perceived? Clinics in Dermatology, *37*(5), 392-401. https://doi.org/10.1016/j.clindermatol.2019.07.008

Gunčaga, J., Budai, L., Kenderessy, T., (2020). Visualization in geometry education as a tool for teaching with better understanding and teaching. Mathematics and Computer Science, 18(4), 337-346.

Gupta, M.K. and Chandra, P., 2020. A comprehensive survey of data mining. *International Journal of Information Technology*, *12*(4), pp.1243-1257.

Hashemi-Pour, C., Brush, K and Burns, E. 2022. What is data visualization and why is it important? Available at>https://www.techtarget.com/searchbusinessanalytics/definition/data-visualization>[Accessed 200824]

Hastings, J. S., Madrian, B. C., Skimmyhorn, W. L. (2013). FINANCIAL LITERACY, FINANCIAL EDUCATION AND ECONOMIC OUTCOMES. *Annual Review of Economics*, *5*, 347. https://doi.org/10.1146/annurev-economics-082312-125807.

He, Q. P., Wang, J. 2020. Application of systems engineering principles and techniques in biological big data analytics: A review. Processes, *8*(8), 951.

Hearst, M.A. and Rosner, D., 2008, January. Tag clouds: Data analysis tool or social signaller? In Proceedings of the 41st Annual Hawaii International Conference on System Sciences *(HICSS 2008)* IEEE. 160-169

Heberle, H., Carazzolle, M. F., Telles, G. P., Meirelles, G. V.,] Minghim, R. (2017). CellNetVis: A web tool for visualising biological networks using a force-directed layout constrained by cellular components. *BMC Bioinformatics*, *18*(10), 25–37.

Himeur, Y., Varlamis, I., Kheddar, H., Amira, A., Atalla, S., Singh, Y., Bensaali, F. and Mansoor, W., 2023. Federated learning for computer vision. *arXiv preprint arXiv:2308.13558*.

Holten, D. H. R. (2009). Visualization of graphs and trees for software analysis. [Ph.D. Thesis]. Technische Universiteit Eindhoven. https://doi.org/10.6100/IR642975.

Hooper, S. D., Bork, P. (2005). Medusa: A simple tool for interaction graph analysis. Bioinformatics, *21*(24), 4432–4433.

Hu, Y., Nöllenburg, M. (2019). Graph visualization. In Encyclopedia of big data technologies, eds S. Sakr, A. Y. Zomaya. Springer. P. 58. https://doi.org/10.1007/978-3-319-77525-8_324.

Huang, X., Lai, W. (2006). Clustering graphs for visualization via node similarities. Journal of Visual Languages & Computing, *17*(3), 225–253. https://doi.org/10.1016/j.jvlc.2005.10.003.

Hue, L., Beauloye, C., Bertrand, L. (2015). Principles in the Regulation of Cardiac Metabolism. *The Scientist's Guide to Cardiac Metabolism*, 57-71. https://doi.org/10.1016/B978-0-12-802394-5.00005-4.

Hussein, N., Alashqur, A. Sowan, B., 2015. Using the interestingness measure lift to generate association rules. Journal of Advanced Computer Science & Technology, *4*(1), 156.

Huynh, X.H., Guillet, F. and Briand, H., 2005, October. A data analysis approach for evaluating the behavior of interestingness measures. In *International Conference on Discovery Science* (pp. 330-337). Berlin, Heidelberg: Springer Berlin Heidelberg.

Iragne, F., Nikolski, M., Mathieu, B., Auber, D., Sherman, D. (2005). ProViz: Protein interaction visualization and exploration. Bioinformatics, *21*(2), 272–274.

Isenberg, P., Elmqvist, N., Scholtz, J., Cernea, D., Ma, L., Hagen, H. (2011). Collaborative visualization: Definition, challenges, and research agenda. *Information Visualization*. https://doi.org/10.1177/1473871611412817.

Jalali-Heravi, M. Zaïane, O.R., (2010), March. A study on interestingness measures for associative classifiers. In Proceedings of the 2010 ACM Symposium on Applied Computing. 1039-1046.

Janecek, P., Schikel, V., Pu, P. (1970). Concept expansion using semantic fisheye views. https://www.researchgate.net/publication/37454898_Concept_Expansion_Using_Semantic_Fisheye_Views.

Jing, X., Patel, V. L., Cimino, J. J., Shubrook, J. H., Zhou, Y., Liu, C., Lacalle, S. D. (2022). The Roles of a Secondary Data Analytics Tool and Experience in Scientific Hypothesis Generation in Clinical Research: Protocol for a Mixed Methods Study. *JMIR Research Protocols*, *11*(7). https://doi.org/10.2196/39414.

Juenemann, J. (2023). How to avoid tooltip annotations to your looker studio reports. Available at> https://measureschool.com/data-studio-tooltip-annotations/>[Accessed 260724].

Junker, B. H., Koschützki, D., Schreiber, F. (2006). Exploration of biological network centralities with CentiBiN. BMC Bioinformatics, *7*:219. https://doi.org/10.1186/1471-2105-7-219 .

Jusufi I 2013 Multivariate networks: visualization and interaction techniques.

Kabir, S. M. (2016). Methods of data collection. In the book Basic Guidelines for Research: An Introductory Approach for All Disciplines. Edition: First Chapter: 9; Publisher: Book Zone Publication, Chittagong-4203, Bangladesh. 201-275

Khaire, U. M., & Dhanalakshmi, R. (2022). Stability investigation of improved whale optimization algorithm in the process of feature selection. *IETE Technical Review*, *39*(2), 286-300.

Kim, S., Mukhiddinov, M. (2022). Data anomaly detection for structural health monitoring based on a convolutional neural network. Sensors, 23(20), 852-5. https://doi.org/10.3390/s23208525.

Koh, G. C., Porras, P., Aranda, B., Hermjakob, H., Orchard, S. E. (2012). Analysing protein–protein interaction networks. Journal of Proteome Research, *11*(4), 2014–2031.

Kohl, M., Wiese, S., Warscheid, B. (2011). Cytoscape: software for visualization and analysis of biological networks. Methods in molecular biology (Clifton, N.J.), 69(6), 291–303. https://doi.org/10.1007/978-1-60761-987-1_18

Köhler, J., Baumbach, J., Taubert, J., Specht, M., Skusa, A., Rüegg, A., et al.(2006). Graph-based analysis and visualization of experimental results with ONDEX. Bioinformatics, 22(11), 1383–1390.

Kong, X., Shi, Y., Yu, S., Liu, J., Xia, F. (2019). Academic social networks: Modeling, analysis, mining and applications. *Journal of Network and Computer Applications*, *132*, 86-103. https://doi.org/10.1016/j.jnca.2019.01.029.

Kumar, S. (2023). Surround-view fisheye camera perception for automated driving: Overview, survey, and challenges. IEEE Transactions on Intelligent Transportation Systems 99:1-22. DOI:10.1109/TITS.2023.3235057

Li, M., Gao, H., Wang, J. Wu, F.X., (2019). Control principles for complex biological networks. Briefings in Bioinformatics, *20*(6), 2253-2266.

Li, Z., Zhao, B., Wang, D., Wen, Y., Liu, G., Dong, H., et al. (2014). DNA nanostructure-based universal microarray platform for high-efficiency multiplex bioanalysis in biofluids. ACS Applied Materials & interfaces, *6*(20), 17944-17953.

Lindfors, E., van Dam, J.C., Lam, C.M.C., Zondervan, N.A., Martins dos Santos, V.A. and Suarez-Diez, M., (2018). SyNDI: synchronous network data integration framework. BMC Bioinformatics, 19(1), 1-15.

Lisboa, P.J., Saralajew, S., Vellido, A., Fernández-Domenech, R. and Villmann, T., 2023. The coming of age of interpretable and explainable machine learning models. *Neurocomputing*, *535*, pp.25-39.

Luciani, T., Burks, A., Sugiyama, C., Komperda, J., Marai, G. E. (2018). Details-first, Show Context, Overview Last: Supporting Exploration of Viscous Fingers in Large-Scale Ensemble Simulations. *IEEE Transactions on Visualization and Computer Graphics*. https://doi.org/10.1109/TVCG.2018.2864849.

Luo, W., (2019). User choice of interactive data visualization format: The effects of cognitive style and spatial ability. Decision Support Systems, *122*:113061. https://doi.org/10.1016/j.dss.2019.05.001

Lynch, M. (2000). Visualization: Representation in Science. *International Encyclopedia of the Social & Behavioral Sciences*, 16288-16292. https://doi.org/10.1016/B0-08-043076-7/03181-8.

Mao, Z., Jiang, Y., Min, G., Leng, S., Jin, X., Yang, K. (2017). Mobile social networks: Design requirements, architecture, and state-of-the-art technology. *Computer Communications*, *100*, 1-19. https://doi.org/10.1016/j.comcom.2016.11.006.

Marcílio-Jr, W. E., Eler, D. M., Paulovich, F. V., Rodrigues-Jr, J. F., Artero, A. O. (2021). ExplorerTree: A focus+context exploration approach for 2D embeddings. Big Data Research, *25*:100239. https://doi.org/10.1016/j.bdr.2021.100239.

Maxwell, J. A. (2016). Expanding the history and range of mixed methods research. Journal of Mixed Methods Research, 10(1), 12–27. https://doi.org/10.1177/1558689815571132.

Maya, B, (2023). 10 common challenges of data visualization & their solutions. Available at> https://synodus.com/blog/big-data/challenges-of-data-visualization/>[Accessed 160224]

McCurdy, N., Dykes, J., Meyer, M. 2016. Action design research and visualization design. BELIV '16: Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization. Pp 10 – 18. https://doi.org/10.1145/2993901.2993916

McGarry, K., 2005. A survey of interestingness measures for knowledge discovery. *The knowledge engineering review*, *20*(1), pp.39-61.

McIntyre, R.S., Rosenblat, J.D., Nemeroff, C.B., Sanacora, G., Murrough, J.W., Berk, M., et al., (2021). Synthesising the evidence for ketamine and ketamine in treatment-resistant depression: an international expert opinion on the available evidence and implementation. American Journal of Psychiatry, *178*(5), 383-399.

Milenković, T., Memišević, V., Bonato, A. Pržulj, N., (2011). Dominating biological networks. PloS one, *6*(8), 230-236.

Millan, P.P., (2013). Visualization and analysis of biological networks. Methods in molecular biology (Clifton, N.J.), 1021, 63–88. https://doi.org/10.1007/978-1-62703-450-0_4

Miryala, S.K., Anbarasu, A. Ramaiah, S. (2018). Discerning molecular interactions: a comprehensive review on biomolecular interaction databases and network analysis tools. Gene, 642Is, 84-94. https://doi.org/10.1016/j.gene.2017.11.028

Mousavian, Z., Khodabandeh, M., Sharifi-Zarchi, A., Nadafian, A. Mahmoudi, A. (2021). StrongestPath: a Cytoscape application for protein–protein interaction analysis. BMC Bioinformatics, *22*(1), 352. https://doi.org/10.1186/s12859-021-04230-4

Muelder, C.W., Ma, K. (2013). Large-scale graph visualization and analytics; Computer 46(7), 39-46.

Munzner, T. 2009. A nested model for visualization design and validation. Available at>https://deeplearning.lipingyang.org/wp-content/uploads/2019/04/nestedmodel09_Tamara-Munzner.pdf

Mutiara, A.B., Wirawan, S., Yusnitasari, T. Anggraini, D. (2023). Expanding Louvain Algorithm for Clustering Relationship Formation. International Journal of Advanced Computer Science and Applications, *14*(1), 701-708.

Muzio, G., Borgwardt, K. (2021). Biological network analysis with deep learning. *Briefings in Bioinformatics*, *22*(2), 1515-1530. https://doi.org/10.1093/bib/bbaa257.

Myers, O.D., Sumner, S.J., Li, S., Barnes, S. Du, X., (2017). One step forward for reducing false positive and false negative compound identifications from mass spectrometry metabolomics data: new algorithms for constructing extracted ion chromatograms and detecting chromatographic peaks. Analytical chemistry, 89(17), 8696-8703.

Nadj, M., Maedche, A., Schieder, C. (2020). The effect of interactive analytical dashboard features on situation awareness and task performance. *Decision Support Systems*, *135*, 113322. https://doi.org/10.1016/j.dss.2020.113322.

Naveed, N., Gottron, T., Kunegis, J. and Alhadi, A.C., 2011, June. Bad news travel fast: A content-based analysis of interestingness on twitter. In *Proceedings of the 3rd international web science conference* (pp. 1-7).

Nguyen, T., Dao, T., Pham, D., Duong, T. (2024). Exploring the Molecular Terrain: A Survey of Analytical Methods for Biological Network Analysis. *Symmetry*, *16*(4), 462. https://doi.org/10.3390/sym16040462.

Nielsen J, Mack RL (1994). Executive Summary. In: Nielsen J, Mack RL, editors. Usability inspection methods. 1st ed. New York, NY: John Wiley & Sons Inc; 1–24.

Nishida, K., Maruyama, J., Kaizu, K., Takahashi, K. Yugi, K., (2023). transomics2cytoscape: An automated software for interpretable 2.5-dimensional visualization of trans-omic networks. bioRxiv, 2023:3. PPRID: 10.1101/2023.03.08.531686v1

Novak, R., Bahri, Y., Abolafia, D.A., Pennington, J. Sohl-Dickstein, J., (2018). Sensitivity and generalisation in neural networks: an empirical study [Preprint]. Available at: https://arxiv.org/abs/1802.08760

Nusrat, S., Kobourov, S. (2016). The state of the art in cartograms. Computer Graphics Forum, *35*(3), 619–642. https://doi.org/10.1111/cgf.12932.

O.Nyumba, T., Wilson, K., Derrick, C. J., Mukherjee, N. (2017). The use of focus group discussion methodology: Insights from two decades of application in conservation. *Methods in Ecology and Evolution*, *9*(1), 20-32. https://doi.org/10.1111/2041-210X.12860.

Okka, A., Dogrusoz, U., Balci, H. (2021). CoSEP: a compound spring embedder layout algorithm with support for ports. Information Visualization, *20*(2-3), 151-169.

Ono, K., (2021). User Documentation <https://cytoscape.org/documentation_users.html> [Accessed 21 June 2022].

Otten, J., Cheng, K, Drewnowski, A. (2015). Infographics and public policy; using data visualization to convey complex information. Health Affairs 34(11):1901-1907. DOI:10.1377/hlthaff.2015.0642.

Pan, A., Lahiri, C., Rajendiran, A., Shanmugham, B. (2016). Computational analysis of protein interaction networks for infectious diseases. *Briefings in Bioinformatics*, *17*(3), 517-526. https://doi.org/10.1093/bib/bbv059.

Paul, T.J. Kollmannsberger, P., (2020). Biological network growth in complex environments: A computational framework. PLOS Computational Biology, *16*:11, https://doi.org/10.1371/journal.pcbi.1008003

Pavlopoulos, G. A., O'Donoghue, S. I., Satagopam, V. P., Soldatos, T. G., Pafilis, E., Schneider, R. (2008). Arena3D: Visualization of biological networks in 3D. BMC Systems Biology, 2(104), 1–7.

Pavlopoulos, G. A., Paez-Espino, D., Kyrpides, N. C., Iliopoulos, I. (2017). Empirical comparison of visualization tools for larger-scale network analysis. Advances in Bioinformatics, 2017(1278932), https://doi.org/10.1155/2017/1278932

Pavlopoulos, G. A., Wegener, A. L., Schneider, R. A. (2008). Survey of visualization tools for biological network analysis. Biodata Mining, 1:12. https://doi.org/10.1186/1756-0381-1-12

Peng, S., Zhou, Y., Cao, L., Yu, S., Niu, J., Jia, W. (2018). Influence analysis in social networks: A survey. *Journal of Network and Computer Applications*, *106*, 17-32. https://doi.org/10.1016/j.jnca.2018.01.005.

Petropoulos, F., Apiletti, D., Assimakopoulos, V., Babai, M. Z., Barrow, D. K., Ben Taieb, S., et al. (2022). Forecasting: Theory and practice. International Journal of Forecasting, *38*(3), 705-871. https://doi.org/10.1016/j.ijforecast.2021.11.001

Piñero, J., Saüch, J., Sanz, F. Furlong, L.I., (2021). The DisGeNET cytoscape app: Exploring and visualising disease genomics data. Computational and structural biotechnology journal, 19(11), 2960-2967.

Pinney, J. W., Shirley, M. W., McConkey, G. A., Westhead, D. R. (2005). MetaSHARK: Software for automated metabolic network prediction from DNA sequence and its application to Plasmodium falciparum and Eimeria tenella genomes. Nucleic Acids Research, *33*(4), 1399–1409.

Pon, R.K., Cárdenas, A.F., Buttler, D.J. and Critchlow, T.J., 2011. Measuring the interestingness of articles in a limited user environment. *Information processing & management*, *47*(1), pp.97-116.

Praneenararat, T., Takagi, T., Iwasaki, W. (2011). Interactive, multiscale navigation of large and complicated biological networks. Bioinformatics, *27*(8), 1121–1127. https://doi.org/10.1093/bioinformatics/btr083.

Qin, X., Luo, Y., Tang, N., Li, G., (2020). Making data visualization more efficient and effective: a survey. The VLDB Journal, 29, 93-117.

Rahman, S. (2016). The Advantages and Disadvantages of Using Qualitative and Quantitative Approaches and Methods in Language “Testing and Assessment” Research: A Literature Review. Journal of Education and Learning*;* 6:1. DOI:10.5539/jel.v6n1p102

Ramos, P.I.P., Arge, L.W.P., Lima, N.C.B., Fukutani, K.F. De Queiroz, A.T.L., (2019). Leveraging user-friendly network approaches to extract knowledge from high-throughput omics datasets. Frontiers in genetics, 10:1120. doi: 10.3389/fgene.2019.01120

Rebolledo, R., Navarrete, S.A., Kéfi, S., Rojas, S. Marquet, P.A., (2019). An open-system approach to complex biological networks. SIAM Journal on Applied Mathematics, *79*(2), 619-640.

Rehman, M. H. U., Liew, C. S., Wah, T. Y., Khan, M. K. (2017). Towards next-generation heterogeneous mobile data stream mining applications: Opportunities, challenges, and future research directions. *Journal of Network and Computer Applications*, *79*, 1-24. https://doi.org/10.1016/j.jnca.2016.11.031.

Reuter, J. A., Spacek, D. V., Snyder, M. P. (2015). High-throughput sequencing technologies. Molecular Cell, *58*(4), 586–597.

Röhlig, M., Prakasam, R. K., Stüwe, J., Schmidt, C., Stachs, O., Schumann, H. (2019). Enhanced grid-based visual analysis of retinal layer thickness with optical coherence tomography. Information, 10(9), 266. https://doi.org/10.3390/info10090266.

Rohr, J.R., Salice, C.J. Nisbet, R.M., (2016). The pros and cons of ecological risk assessment based on data from different levels of biological organisation. Critical Reviews in Toxicology, 46(9), 756-784.

Sajjad Kabir, S. (2016). Methods of data collection. In book: Basic Guidelines for Research: An Introductory Approach for All Disciplines (pp.201-275). Edition: First Chapter: 9. Publisher: Book Zone Publication, Chittagong-4203, Bangladesh.

Salem, M.S.Z., (2018). Biological networks: An introductory review. Journal of Proteomics and Genomics Research, *2*(1), 41-111.

Samariya, D. and Thakkar, A., 2023. A comprehensive survey of anomaly detection algorithms. *Annals of Data Science*, *10*(3), pp.829-850.

Sarker, I.H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN COMPUT. SCI.* **2**, 160 https://doi.org/10.1007/s42979-021-00592-x.

Schaffer, D., Zuo, Z., Greenberg, S., Bartram, L., Dill, J., Dubs, S et al. (1996). Navigating hierarchically clustered networks through fisheye and full-zoom method. ACM Transactions on Computer-Human Interaction, *3*(2), 162–188.

Schneider, M. V. (Ed.). (2013). Silico systems biology. Methods in Molecular Biology, vol. 1021 Humana Press, 189–199.

Schulz, H.J., Hadlak, S. Schumann, H., (2010). The design space of implicit hierarchy visualization: A survey. IEEE Transactions on Visualization and computer graphics, 17(4), 393-411.

Sedlmair, M., Meyer, M and Munzner, T. 2012. Design study methodology; reflections from the trenches and the stacks. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 18(12): 2431-2440, 2012.

Sedlmair, M., Meyer, M., Munzner, T. (2012). Design study methodology: Reflections from the trenches and the stacks. IEEE Transactions on Visualization and Computer Graphics, 18(12), 2431–2440.

Selvarangam, K. Kumar, K.R., (2014), November. Interestingness of measures: a statistical perspective. In 2014 International Conference on Contemporary Computing and Informatics (IC3I). IEEE, 209-213.

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., et al. (2003). Cytoscape: A software environment for integrated models of biomolecular interaction networks. Genome Research, 13(11), 2498–2504.

Sharma, D.K., Dharmaraj, A., Al Ayub Ahmed, A., Suresh Kumar, K., Phasinam, K. Naved, M., (2022), June. A Study on the Relationship Between Cloud Computing and Data Mining in Business Organizations. In Proceedings of Second International Conference in Mechanical and Energy Technology: ICMET 2021, India. Singapore: Springer Nature Singapore. 91-99

Sia, J., Zhang, W., Jonckheere, E., Cook, D., Bogdan, P. (2022). Inferring functional communities from partially observed biological networks exploiting geometric topology and side information. *Scientific Reports*, *12*. https://doi.org/10.1038/s41598-022-14631-x.

Siro, C., Aliannejadi, M. and De Rijke, M., 2023. Understanding and predicting user satisfaction with conversational recommender systems. *ACM Transactions on Information Systems*, *42*(2), pp.1-37.

Sjödin, D., Parida, V., Palmié, M., Wincent, J. (2021). How AI capabilities enable business model innovation: Scaling AI through co-evolutionary processes and feedback loops. *Journal of Business Research*, *134*, 574-587. https://doi.org/10.1016/j.jbusres.2021.05.009.

Smith-Miles, K., Lopes, L. (2012). Measuring instance difficulty for combinatorial optimisation problems. *Computers & Operations Research*, *39*(5), 875-889. https://doi.org/10.1016/j.cor.2011.07.006.

Smoot, L., Mellin, J., Brinkman, C.K., Popova, I., Coats, E.R., (2022). Interrogating nitritation at a molecular level: Understanding the potential influence of Nitrobacter spp. Water Research, 224, 119074. https://doi.org/10.1016/j.watres.2022.119074

Spohr, P., (2017). Developing and evaluating a Cytoscape app for graph-based clustering. [Bachelor thesis], University of Düsseldorf.

Stansfield SK, Walsh J, Prata N, et al. Information to Improve Decision Making for Health. In: Jamison DT, Breman JG, Measham AR, et al., editors. Disease Control Priorities in Developing Countries. 2nd edition. Washington (DC): The International Bank for Reconstruction and Development / The World Bank; 2006. Chapter 54. Available from: https://www.ncbi.nlm.nih.gov/books/NBK11731/ Co-published by Oxford University Press, New York.

Stone, L., (2018). The feasibility and stability of large complex biological networks: a random matrix approach. Scientific reports, 8(1), 8246.

Storey, M., Best, C., Michand, J. (2001) "SHriMP views: an interactive environment for exploring Java programs," Proceedings 9th International Workshop on Program Comprehension. IWPC 2001, Toronto, ON, Canada, 2001, 111-112, doi: 10.1109/WPC.2001.921719.

Su, G., Morris, J.H., Demchak, B. Bader, G.D. (2014). Biological network exploration with Cytoscape 3. Current protocols in bioinformatics, 47(1), 8-13.

Suderman, M., Hallett, M. (2007). Tools for visually exploring biological networks. Bioinformatics, *23*(20), 2651–2659.

Sutton, R. T., Pincock, D., Baumgart, D. C., Sadowski, D. C., Fedorak, R. N., & Kroeker, K. I. (2020). An overview of clinical decision support systems: benefits, risks, and strategies for success. *NPJ digital medicine*, *3*(1), 17.

Taherdoost, H. (2021). Data collection methods and tools for research: A step-by-step guide to choosing data collection technique for academic and business research projects. International Journal of Academic Research in Management, *10*(1), 10–38.

Tan, P.N., Kumar, V. and Srivastava, J., 2002, July. Selecting the right interestingness measure for association patterns. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 32-41).

Taubert, J., Hassani-Pak, K., Castells-Brooke, N., Rawlings, C. J. (2014). Ondex Web: Web-based visualization and exploration of heterogeneous biological networks. Bioinformatics, *30*(7), 1034–1035.

Tavallali, P., Tavallali, P. Singhal, M. (2021). K-means tree: an optimal clustering tree for unsupervised learning. The Journal of Supercomputing, 77, 5239-5266.

Tenny, S., Brannan, J. M., Brannan, G. D. (2022). *Qualitative Study*. In: StatPearls Treasure Island (FL): StatPearls Publishing. https://www.ncbi.nlm.nih.gov/books/NBK470395/

Terrell, S. (2012). Mixed-methods research methodologies. Qualitative Report, 17(1), 254–265.

Thimm, O., Bläsing, O., Gibon, Y., Nagel, A., Meyer, S., Krüger, P., et al. (2004). MAPMAN: A user-driven tool to display genomics data sets onto diagrams of metabolic pathways and other biological processes. The Plant Journal, 37(6), 914–939. https://doi.org/10.1111/j.1365-313x.2004.02016.x.

Tollis, I.G., Battista, G.D., Eades, P., Tamassia, R. 1999. Advances in the theory and practice of graph drawing. Theoretical computer science, 217(2), 235-254

Tominski, C., Abello, J., Schumann, H., Ham, F. E. (2006). Fisheye tree views and lenses for graph visualization. In proceedings 10th international conference on information visualization (IV 2006, London, UK, July 5-7, 2006). Los Alamitos, CA, IEEE Computer society. 17-24, https://doi.org/10.1109/IV.2006.54

Turetken, O and Schuff, D. 2002. The use of fisheye view visualization in understanding business process. Available at> https://www.researchgate.net/publication/221408584_The_Use_Of_Fisheye_View_Visualizations_In_Understanding_Business_Process>[Accessed 200824]

Turetken, O., Schuff, D. (2002). The use of fisheye view visualization in understanding business processes. Gdansk, Poland, ECIS, 322-330

Tutz, G., 2022. Ordinal regression: A review and a taxonomy of models. Wiley Interdisciplinary Reviews: Computational Statistics, 14(2), 1545. https://doi.org/10.1002/wics.1545

Ugwu, C., Eze, V. (2023). Qualitative research. IDOSR Journal of Computer and Applied Sciences, 8(1), 20–35

Väätäjä, H., Varsaluoma, J., Heimonen, T., Tiitinen, K., Hakulinen, J., Turunen, M., et al. (2016). Information visualization heuristics in practical expert evaluation. In Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization. ACM, 36–43.

Vaillant, B., Lenca, P. and Lallich, S., 2004. A clustering of interestingness measures. In *Discovery Science: 7th International Conference, DS 2004, Padova, Italy, October 2-5, 2004. Proceedings 7* (pp. 290-297). Springer Berlin Heidelberg.

Valsamidis, S., Kontogiannis, S., Kazanidis, I. and Karakos, A., 2011. E-learning platform usage analysis. *Interdisciplinary Journal of E-Learning and Learning Objects*, *7*(1), pp.185-204.

Van Den Brand, J., Chen, L., Peng, R., Kyng, R., Liu, Y.P., Gutenberg, M.P., et al. (2023). A deterministic almost-linear time algorithm for minimum-cost flow. ArXiv. /abs/2309.16629, 503-514.

Van der Geest, T. Van Dongelen, R. (2009). What is beautiful is useful: visual appeal and expected information quality. Conference: Professional Communication Conference, 2009. IPCC 2009. IEEE International. DOI:10.1109/IPCC.2009.5208678.

Ware, C. (2013). Information visualization perception for design (3rd edition). http://b3.stmik-banjarbaru.ac.id/data.bc/15.%20Information%20Retrieval/15.%20Information%20Retrieval/

2013%20Information%20Visualization%20Perception%20for%20Design.pdf>[Accessed 07-12-2023].

West, D., Allman, B., Hunsaker, E., Kimmons, R. (2020). Visual Aesthetics: The Art of Learning. In R. Kimmons & S. Caskurlu (Eds.), The Students' Guide to Learning Design and Research. EdTech Books. https://edtechbooks.org/studentguide/visual_aesthetics

West, J., Bhattacharya, M. (2016). Intelligent financial fraud detection: A comprehensive review. *Computers & Security*, *57*, 47-66. https://doi.org/10.1016/j.cose.2015.09.005.

White, H. D. McCain, K. W. (1998). Visualising a discipline: An author co-citation analysis of information science, 1972–1995. Journal of the American Society for Information Science, 49(4), 327–355.

Whiting LS (2008). Semi-structured interviews: guidance for novice researchers. *Nurs Stand* ;22:35–40. 10.7748/ns2008.02.22.23.35.c6420 .

Williams, R., Scholtz, J., Blaha, L.M., Franklin, L., Huang, Z. (2018). Evaluation of visualization heuristics. In Human–Computer Interaction. Theories, Methods, and Human Issues: 20th International Conference, HCI International 2018, Las Vegas, NV, USA, July 15–20, 2018, Proceedings, Part I. Springer International Publishing. 208–224.

Wong, B. (2012). Visualising biological data. Nature America, *9*(12), 1131–1161.

Wu, E and Chang, R. 2024. Design-specific transforms in visualization. Available at> https://arxiv.org/html/2407.06404v1>[Accessed 200824]

Xin, D., Shen, X., Mei, Q. and Han, J., 2006, August. Discovering interesting patterns through user's interactive feedback. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 773-778).

Yeung, N., Cline, M. S., Kuchinsky, A., Smoot, M. E., Bader, G. D. (2008). Exploring biological networks with Cytoscape software. Current Protocols in Bioinformatics, 23(1), 8–13.

Zhou, G. Xia, J., (2018). OmicsNet: a web-based tool for the creating and visualising biological networks in 3D space. Nucleic acids research, 46(1), 514-522.

Zhou, Z., Pan, Y., Lemieux, G.G., Ivanov, A., (2023). MEDUSA: A Multi-resolution Machine Learning Congestion Estimation Method for 2D and 3D Global Routing. ACM Transactions on Design Automation of Electronic Systems; 1;4. DOI:10.1145/3590768

Zudilova-Seinstra, E., Adriaansen, T. Liere, R.V., (2009). Overview of interactive visualization. Springer London, 3-15.

Appendix A

Survey questions for evaluating the general and heuristics factors.

Section A: Demographic Information

Section B: Knowledge and Experience of Visualization Tools

Question 1:

Do you think network Visualization tools aid network analysis?

- ☐ Yes
- ☐ No
- ☐ Maybe

Question 2:

Which of the following network Visualization tools are you aware of?

- ☐ Cytoscape
- ☐ Gephi
- ☐ Pajek
- ☐ ProViz
- ☐ Osprey
- ☐ Medusa
- ☐ ONDEX
- ☐ Others (please identify)

Question 3:

Which of the following network Visualization tools have you used?

- ☐ Cytoscape
- ☐ Gephi
- ☐ Pajek
- ☐ ProViz
- ☐ Osprey
- ☐ Medusa
- ☐ ONDEX
- ☐ Others (please identify)

Question 4:

How long have you been using the network Visualization tool(s)?

- ☐ 0–5 years
- ☐ 6–10 years
- ☐ 11–15 years
- ☐ 16–30 years
- ☐ More than 30 years

Question 5:

Do you think network Visualization is a relevant skill for a data scientist/analyst?

- ☐ Yes
- ☐ No
- ☐ Maybe

Section C: Requirements of Network Visualization Tools

This section aims to identify which of these factors are considered most important in good network Visualization tools. A range of statements on network Visualization tools and their features are provided below. Please respond based on the level at which you agree with each of the statements.

C1: General Factors

Factor 1: Filtering Tools

	Strongly Agree	**Agree**	**Neutral**	**Disagree**	**Strongly Disagree**
A good network Visualization tool should have filtering tools to reduce the amount of data.					
Data analysts often want to use filtering tools when visualising networks.					
Visualization filtering tools should be easily accessible.					
Node filtering and regular expressions should be possible with a network Visualization tool.					

Factor 2: Plugins

	Strongly Agree	**Agree**	**Neutral**	**Disagree**	**Strongly Disagree**
The availability of plugins is an indicator of a good network Visualization tool.					
Network Visualization tools should have plugins that are easy to install.					
Network Visualization tools with plugins have more functionalities than those without plugins.					
Network Visualization tools with plugins are generally preferred to those without plugins.					

Factor 3: Visual styles

	Strongly Agree	**Agree**	**Neutral**	**Disagree**	**Strongly Disagree**
The first thing people look for in a network Visualization tool is its visual style.					
Aesthetics is very important in network Visualizations.					
Network Visualization tools that support 3D Visualization are preferred to those that support only 2D Visualization.					
A good network Visualization tool draws graphs in ways that improve understanding and network communication.					
Network Visualization tools that support node Visualization with layers are preferred to those that do not use layers.					

Factor 4: Advanced Search

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
A good network Visualization tool should centre the screen on a searched object in the network.					
Network Visualization tools should support a graph search by node and edge attributes.					
A good network Visualization tool should be capable of searching for web pages.					
A good network Visualization tool should be capable of c searches using regular expressions.					

Factor 5: Free/Open-Source

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Open-source network Visualization tools are preferred to closed source tools.					
Open-source network Visualization tools generally have more functionalities than those that are closed source.					
Free network Visualization tools are preferred to commercial ones.					
The source codes of open-source network Visualization tools are useful.					
Open-source network Visualization tools have more online support than those that are closed source.					

Factor 6: Efficient Layout Algorithms

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
A good network Visualization tool should quickly represent graphs in different layouts.					
A good network Visualization tool should support a multi-layer layout.					
A good network Visualization tool should support layouts that tell the whole story in a graph.					
A good network Visualization tool should be able to visualise complex graphs.					

Factor 7: Scalability

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization tools should be fast to load data.					
Network Visualization tools should be able to visualise large networks.					
Network Visualization tools should scale with the size of nodes in a network.					
Network Visualization tools should scale with the size of edges in a network.					
Network Visualization tools should scale with representing graph visuals.					

Factor 8: Different File Formats

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization tools should support CSV files.					
A good network Visualization tool should support the file formats of other Visualization tools.					
A good network Visualization tool should be able to export data to the formats of other Visualization tools.					
A good network Visualization tool should not have an unconventional file format.					
Network Visualization tools should support an adjacency matrix.					

Factor 9: Text Mining

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
A good graph Visualization tool should be able to do lemmatisation.					
A good graph Visualization tool should be able to do stemming.					
A good graph Visualization tool should be able to perform **semantic analysis.**					
A good graph Visualization tool should support text mining tools by default.					

Factor 10: User Input & Customisation

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization applications should enable user customisation.					
Network Visualization applications should be capable of being expanded.					
Functionalities possible with a mouse should also be possible with a touchpad in network Visualization tools.					
Users should have the ability to drag and drop objects with network Visualization tools.					
Users should have the ability to change the default layout of a network Visualization tool.					

Factor 11: Graph Analysis

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization applications should be able to do spanning tree.					
Network Visualization applications should be able to detect cycles.					
Network Visualization applications should be able to find the shortest path.					
Network Visualization applications should be able to do summary statistics.					
Network Visualization applications should support graph analysis by default.					

Factor 12: Feedback to Users

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization applications should be able to export graph visuals as image files.					
Network Visualization applications should have tooltips.					
Network Visualization applications should be able to print results as PDF files.					
Network Visualization applications should have a progress bar to update users on processes.					

Factor 13: Strength

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization applications should have strength in visualising large and complex networks.					
The main strength of a network Visualization tool affects users' preferences.					
Network Visualization applications should have strength in supporting many layouts.					

Factor 14: Runtime Performance

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization applications should have low computation time.					
Network Visualization applications should load data and export output on time.					
Network Visualization applications should be capable of switching quickly between layouts.					
Runtime performance is an important factor in the choice of a network Visualization tool.					

Factor 15: User-friendliness

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
The screen of a good network Visualization tool should be able to be maximized for visuals.					
A good network Visualization tool should have tooltips and movable panes.					
A good network Visualization application should have self-explanatory tool names.					
A good network Visualization application should have an easy-to-use interface.					

C2: Heuristic Factors

Heuristic Factor 1: Information Coding

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization tools should use layers to differentiate nodes.					
Network Visualization tools should enable the use of different line types and colours to represent edges.					
Network Visualization tools should be able to use different colours to represent node clusters.					
Network Visualization tools should be able to use different symbols to represent nodes.					

Heuristic Factor 2: Flexibility

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization tools should be able to load different data types.					
Network Visualization tools should support plugins, movable planes and different layouts.					
Network Visualization tools should be flexible.					

Heuristic Factor 3: Orientation and Help

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization applications should have manuals.					
Network Visualization applications should have easily accessible tutorials and tooltips.					
Network Visualization applications should have an online community for help.					
Network Visualization tools should have helpful error messages.					

Heuristic Factor 4: Minimal Actions

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Changing network layouts in graph Visualization tools should require few steps.					
Network Visualization tools should require few steps to style networks.					
Network Visualization tools should require few steps to visualise a network reasonably.					
Network Visualization tools should not require several installation steps.					

Heuristic Factor 5: Prompting

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Prompt boxes should serve as a guide to users before allowing them to take any action.					
Error messages should suggest possible solutions.					
Deletion of nodes should be confirmed before being executed.					

Heuristic Factor 6: Consistency

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Notations and icons used in graph Visualization tools should be consistent with the conventional ones.					
Notations and icons used in graph Visualization tools should be consistent throughout the application.					
Users should be pre-informed of any changes to features in future versions of the application.					
Plugins should be consistent with the application.					

Heuristic Factor 7: Spatial Organisation

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network graphs should be detachable to other windows.					
Network visuals should be capable of full-screen display.					
Network visuals should preferably be 3D.					
Network visuals should use as much of the screen space as possible.					
Nodes and edges should be easily differentiated in graph visuals.					

Heuristic Factor 8: Recognition Rather than Recall

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization buttons should be easily recognised.					
Nodes of interest should be easily recognised in the network visual.					
The names of network Visualization tools should be consistent with their functions.					

Heuristic Factor 9: Removing the Extraneous

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization tools should not have distracting panes.					
Network Visualization tools should not have pop-ups.					
Tools should not be duplicated in network Visualization tools.					
The user interface of network Visualization tools should be simple.					

Heuristic Factor 10: Dataset Reduction

	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Network Visualization tools should support edge or node reduction through selection.					
Network Visualization tools should support data pruning.					
Network Visualization tools should support automated dimension reduction.					
Network Visualization tools should support automated clustering.					

Appendix B

Interview questions for evaluating the general and heuristics factors.

C1: General Factors

Factor 1: Filtering Tools

Question 1: Do you think that it is important that a graph Visualization tool should be able to filter out the connections of a single node?

Question 2: Do you think that it is important that a graph Visualization tool should be able to filter out the connections of a single edge?

Question 3: How important is regular expression in filtering networks when using graph Visualization tools?

Question 4: At what point will filtering of network be important?

Question 5: Do you think that it is important that a graph Visualization tool should be able to filter out the groups of nodes or edges?

Factor 2: Plugins

Question 1: Do you think that it is a good idea to reduce the size of a graph Visualization tool to make some functionalities available as plugins?

Question 2: Do you think that graph Visualization tools with plugins are preferred to those without plugins?

Question 3: What do you think are the downside of using plugins in graph Visualization tools?

Question 4: Do you think that plugins should be available for free?

Question 5: Are there special kinds of functionalities that should be available as plugins in graph Visualization tools?

Factor 3: Visual styles

Question 1: Which Visualization style do you prefer?

- 2D
- 3D
- Others, please specify.

Question 2: Do you think that gaining of insight into associations present in a graph is a function of the visual style?

Question 3: Should graph Visualization tools have multi layers layouts?

Question 4: Should visual style be customisable?

Question 5: What is a good visual style to you?

Factor 4: Advanced Search

Question 1: Which kind of search do you consider to be advance?

Question 2: Do you expect graph Visualization tools to be capable of searching for webpages?

Question 3: Do you think that the use of regular expression should be considered to be an advanced search?

Question 4: Do you consider the ability of a graph Visualization tool to perform advance search as being pivotal in your choice of using the Visualization tool?

Question 5: Do you think that the ability of a graph Visualization tool to search for group is advance?

Factor 5: Free/Open-Source

Question 1: Is being an open-source application something to look for in a graph Visualization tool?

Question 2: Which do you prefer? Free graph Visualization tool

- Commercial graph Visualization tool
- Others, please specify.

Question 3: What are the benefits of using open-source graph Visualization tool?

Question 4: What are the benefits of using free graph Visualization tool?

Question 5: Have you ever used the source code of a graph Visualization tool?

Factor 6: Efficient Layout Algorithms

Question 1: What is the main criterion for considering the layout algorithm of a graph Visualization tool as being efficient?

Question 2: Do you think it is important for a graph Visualization tool to have multi-layer layout?

Question 3: To you, which layout is generally the most efficient in terms of Visualization of networks?

Question 4: Do you think that the efficiency of a particular type of layout is tool dependent?

Question 5: Do you think it is good idea for graph Visualization tools to show the transition of a graph visual into a particular layout?

Factor 7: Scalability

Question 1: Since real life networks are often complex, do you think a graph Visualization tool is expected to quickly produce visuals of networks?

Question 2: Is speed of loading in data an important factor in choosing a graph Visualization tool?

Question 3: What do you think is the benchmark complexity of network a graph Visualization tool should handle?

Question 4: What do you think is the benchmark speed of loading data for graph Visualization tools?

Question 5: Do you consider scalability to be an important factor in the choice of a graph Visualization tool?

Factor 8: Different file formats

Question 1: Do you think it is a good idea for a graph Visualization tool to have its own data format?

Question 2: If graph Visualization tools should have a standard data format, what do you think it should be?

Question 3: Do you think that a graph Visualization tool should be able to export networks to other file formats?

Question 4: Do you think that a graph Visualization tool should be able to import networks stored in other file formats?

Question 5: How does file formats affect your choice of a graph Visualization tool?

Factor 9: Text mining

Question 1: Should graph Visualization tools have the ability to do semantic analysis on networks?

Question 2: Should text mining features be defaulting features or available in plugins?

Question 3: Graph Visualization tools are mainly used for visualising graphs; do you think that text mining feature is needed?

Question 4: Does text mining ability affect your choice graph Visualization tool?

Question 5: What is the main text mining technique you think every graph Visualization tool should have?

Factor 10: User input & Customisation

Question 1: Do you think that functionalities like zooming that are possible with mouse be possible with touchpad when using graph Visualization tools?

Question 2: To what extent do you think graph Visualization tools should allow users' customisation?

Question 3: What are the main types of inputs you think graph Visualization tools should support?

Question 4: Do you think that graph Visualization tools should have moveable panes?

Question 5: Is user input and customisation an important factor to consider in one's choice of graph Visualization tool?

Factor 11: Graph Analysis

Question 1: Graph Visualization tools are widely known for their usage as network Visualization. Do you think they should still be able to be used for graph analysis?

Question 2: Which graph analysis do you think graph Visualization tools should be capable of doing?

Question 3: Do you think that graph analysis features be available in plugins or in the graph Visualization tools?

Question 4: Which technique in graph analysis do you use often?

Question 5: Do you think that graph analysis is an important factor to consider in one's choice of graph Visualization tool?

Factor 12: Feedback to users

Question 1: Do you think that it is important for graph Visualization tools to have progress bar?

Question 2: Which output formats do you think graph Visualization tools should be capable of exporting results to?

Question 3: Do you think that graph Visualization tools should have helpful error messages?

Question 4: Do you think that graph Visualization tools should be able of exporting graphs and reports to Word document. format?

Question 5: Do you think that uses' feedback is an important factor to consider in one's choice of graph Visualization tool?

Factor 13: Strength

Question 1: Do you think that the strength of graph Visualization tools affects users' preference?

Question 2: Do you think that graph Visualization tool should have strength in one area or in many?

Question 3: What metric do you think should be used in comparing the strength of graph Visualization tools?

Question 4: Should the types of layouts available in a graph Visualization tool be an important factor in telling of the strength of a graph Visualization tool?

Question 5: What is the most important criterion in determining the strength of a graph Visualization tool?

Factor 14: Runtime performance

Question 1: Since real world networks are often large and complex, do you think runtime speed is important in choosing a graph Visualization tool?

Question 2: To you, what is the benchmark speed of graph Visualization tools in computing layout algorithms? Fast.

Question 3: How fast should graph Visualization tools switch from one layout to another?

Question 4: How fast do you expect a graph Visualization tool to load in data?

Question 5: Do you think that runtime performance is an important factor in one's choice of a graph Visualization tool?

Factor 15: User-friendliness

Question 1: Do you think that it is important for the screen for showing graph visuals be maximised in graph Visualization tools? why?

Question 2: When using graph Visualization tools, are tooltips important?

Question 3: Is having self-explanatory button names user friendly in graph Visualization tools?

Question 4: Do you think that having moveable panes is important for graph Visualization tools?

Question 5: Should graph Visualization tools have simple interface or an interface with many tools available on the go?

C2: Heuristic Factors

Heuristic Factor 1: Information Coding

Question 1: Do you think that graph Visualization tools should be able to swap source and target nodes?

Question 2: Do you think graph Visualization tools should be able to use different symbols to represent nodes?

Question 3: Do you think that graph Visualization tools should be able to use different line types to represent edges?

Question 4: Do you think that graph Visualization tools should be able to use different colours to represent edges? Why?

Question 5: To you, what is the best way to code information in graph Visualization tools?

Heuristic Factor 2: Flexibility

Question 1: What do you look out for before considering a graph Visualization tool as being flexible?

Question 2: Is compatibility with different data types of an important feature in graph Visualization tools?

Question 3: Do you think that graph Visualization tools should support all possible layouts by default?

Question 4: To what extent should the flexibility of graph Visualization tools be?

Question 5: Do you think that flexibility is an important factor to consider in one's choice of graph Visualization tool?

Heuristic Factor 3: Orientation and Help

Question 1: Do you think that orientation for a graph Visualization tool be available offline?

Question 2: Do you think that graph Visualization tools should have online communities to reach out to for help?

Question 3: Do you think graph Visualization tools should have orientation on first start up?

Question 4: Which of these orientation forms do you think is best?

- Textual
- Audio
- Video
- Combination of any two listed above. Please specify.
- All three listed above
- Others. Please specify.

Question 5: Do you think that orientation and help are important factors to consider before choosing a graph Visualization tool?

Heuristic Factor 4: Minimal actions

Question 1: Is the number of steps taken before performing a task important when using graph Visualization tools?

Question 2: To you, what is the benchmark number of steps to be taken before outputting visuals of network when using graph Visualization tools?

Question 3: Do you think graph Visualization tools should have minimal installation steps?

Question 4: Do you think that graph Visualization tools should have minimal steps before importing data?

Question 5: Do you think that minimal actions is an important factor to consider when choosing a graph Visualization tool?

Heuristic Factor 5: Prompting

Question 1: Promptings can slow down work sometimes; do you think they are needed in graph Visualization tools?

Question 2: Are alert boxes distracting when using graph Visualization tools? why?

Question 3: Do you think it is important for graph Visualization tools to have confirmatory boxes before deletion of certain objects?

Question 4: Do you think that error messages should suggest position solution to errors in graph Visualization tools?

Question 5: Do you think prompt box should disable users from using the graph Visualization application unless they have attended to the prompt in the prompt box?

Heuristic Factor 6: Consistency

Question 1: Do you think that notations used in graph theory be used in graph Visualization tools?

Question 2: Edges are often represented with lines; do you think that graph Visualization tools should stick to this?

Question 3: Do you think that icons be used as buttons in graph Visualization tools?

Question 4: How should graph Visualization tools handle cases where many notations are being used for a particular object in graph theory?

Question 5: Do you think that consistency affects one's choice of graph Visualization tool?

Heuristic Factor 7: Spatial Organisation

Question 1: How important is zoomed to fit in graph Visualization tools?

Question 2: Do you think it is a good idea to make graph Visualization tools to have spate windows for visual outputs?

Question 3: Do you think that graph visuals be given the maximum screen space as possible in graph Visualization tools?

Question 4: To you, what is a good spatial organisation in graph Visualization tools?

Question 5: Do you think that spatial organisation is an important factor to consider when choosing a graph Visualization tool?

Heuristic Factor 8: Recognition rather than recall

Question 1: Should icons be used instead of names of tools in graph Visualization tools? why?

Question 2: Do nodes having different colour from edges make graph more readable?

Question 3: Does the symbol of nodes affect graph readability in graph Visualization tools?

Question 4: Does the line thickness of edges affect the readability of graph Visualization tools?

Question 5: Does default displaying the names of nodes improve recognition?

Heuristic Factor 9: Remove the extraneous

Question 1: Is it a good idea for many features to be made available on the screen as opposed to be in menus when using graph Visualization tools?

Question 2: Do you think that certain tools should be accessible through more than one way in graph Visualization tools? why?

Question 3: Do you think pop ups are a good idea in graph Visualization tools?

Question 4: Do you think Ads are distracting in graph Visualization tools?

Heuristic Factor 10: Data set reduction

Question 1: Real world data are often large, is data reduction a good idea in graph Visualization tools?

Question 2: Should graph Visualization tools support node reduction?

Question 3: Do you think that information captured in pruned network are a reflective of that in the whole network?

Question 4: If all graph Visualization tools must be able to reduce the size of data, which data set reduction algorithm do you think is the must have?

Question 5: Do you think that data set reduction is an important factor to consider when choosing a graph Visualization tool?

Appendix C

Table 29: Filtering tools.

S/N	Interview extract	Codes	Themes
1	Very important(5x)	Strongly agreed	Essential
2	Very important(5x)	Strongly agreed	Essential
3	Important(3x) Regular(2x)	Agreed	Essential
4	At all points(5x)	Strongly agreed	Essential
5	Yes, very useful. (5x)	Strongly agreed	Essential

Table 30: Plugins.

S/N	Interview extract	Codes	Themes
1	Yes (4x) | Maybe, I am not sure (1x)	Strongly agreed	Essential
2	Yes (3x) | I don't know (2x)	Agreed	Essential
3	Complexity (4x) | I don't know(1x)	Strongly disagreed	Not essential
4	Yes, should be available (5x)	Strongly agreed	Essential
5	Yes, others should be added (5x)	Strongly agreed	Essential

Table 31: Visual styles.

S/N	Interview extract	Codes	Themes
1	2D (3x) | 3D (2x)	2D most preferred	Essential
2	Yes (4x) | No (1x)	Strongly agreed	Essential
3	Yes (4x) | Maybe (1x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Good charts (5x)	Strongly agreed	Essential

Table 32: Advanced search.

S/N	Interview extract	Codes	Themes
1	Very important (3x) | I don't know (2x)	Agreed	Essential
2	Yes (2x) | No (3x)	Disagreed	Not essential
3	Yes (2x) | No (3x)	Disagreed	Not essential
4	Yes (2x) | No (3x)	Disagreed	Not essential
5	Yes (4x) | No (1)	Strongly agreed	Essential

Table 33: Free/Open source

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential
2	Free graph visualization tool	Strongly agreed	Essential
3	Accessibility and Economical	Strongly agreed	Essential
4	Accessibility and Economical	Strongly agreed	Essential
5	No (5x)	Strongly disagreed	Not essential

Table 34: Efficient and layout algorithms.

S/N	Interview extract	Codes	Themes
1	Ease to use (4x) | Accuracy (1x)	Strongly agreed	Essential
2	Yes (4) | Maybe (1x)	Strongly agreed	Essential
3	No specific trend	Neutral	Neutral
4	Yes (5x)	Strongly agreed	Essential
5	Yes (4x) | Not necessary(1x)	Strongly agreed	Essential

Table 35: Scalability.

S/N	Interview extract	Codes	Themes

1	Yes (4x) | Maybe (1x)	Strongly agreed	Essential
2	Yes (5x)	Strongly agreed	Essential
3	Ease of use (4x) | Not sure (1x)	Strongly agreed	Essential
4	Very fast (5x)	Strongly agreed	Essential
5	Yes (5x)	Strongly agreed	Essential

Table 36: Different file format.

S/N	Interview extract	Codes	Themes
1	Yes (2x) | No (3x)	Disagreed	Not essential
2	Common format (5x)	Strongly agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Matters (4x) | Doesn't matter (1x)	Strongly agreed	Essential

Table 37: Text mining.

S/N	Interview extract	Codes	Themes
1	Yes (3x) | No (2x)	Agreed	Essential
2	Yes (4x) | No (1x)	Strongly agreed	Essential
3	Matters (3x) | Doesn't matter (2x)	Agreed	Essential
4	Yes (1x) | No (4x)	Strongly disagreed	Not Essential
5	No specific trend	-	-

Table 38: User input and customisation.

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential

2	As much as possible (3x) | Minimum (2x)	Agreed	Essential
3	Common formats (5x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Yes (4x) | No (1x)	Strongly agreed	Essential

Table 39: Graph analysis.

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential
2	Correlation, PCA, Clustering (5x)	Very Important	Essential
3	In the visualization tool (5x)	Strongly agreed	Essential
4	Correlation, Regression, PCA, Ordination	Very Important	Essential
5	Yes (5x)	Strongly agreed	Essential

Table 40: Feedback to users.

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential
2	Common formats (5x)	Strongly agreed	Essential
3	Yes (4x) | No (1x)	Strongly agreed	Essential
4	Yes, common formats (5x)	Strongly agreed	Essential
5	Yes (4x) | No (1x)	Strongly agreed	Essential

Table 41: Strength.

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential
2	Many strengths (4x) | a few (1x)	Strongly agreed	Essential

3	Ease of use (5x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Charts (3x) | Speed (2x)	Strongly agreed	Essential

Table 42: Runtime performance.

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential
2	Fast (3x) | I'm not sure (2x)	Agreed	Essential
3	Very fast (4x) | Not sure (1x)	Strongly agreed	Essential
4	Seconds (3x) | Minute (2x)	Agreed	Essential
5	Yes (5x)	Strongly agreed	Essential

Table 43: User friendliness.

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential
2	Yes (5x)	Strongly agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Yes (4x) | Maybe (1x)	Strongly agreed	Essential
5	Simple interface (5x)	Strongly agreed	Essential

Appendix D

Table 44: Information coding.

S/N	Interview extract	Codes	Themes
1	Yes (4x) | No (1x)	Strongly agreed	Essential
2	Yes (5x)	Strongly agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Different colours (3x) | Don't know (2x)	Agreed	Essential

Table 45: Flexibility.

S/N	Interview extract	Codes	Themes
1	Ease to use (5x)	Strongly agreed	Essential
2	Yes (5x)	Strongly agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Very flexible (5x)	Strongly agreed	Essential
5	Yes (4x) | Not really (1x)	Strongly agreed	Essential

Table 46: Orientation and help.

S/N	Interview extract	Codes	Themes
1	Yes (4x) | Maybe (1x)	Strongly agreed	Essential
2	Yes (5x)	Strongly agreed	Essential
3	Yes (4x) | Not necessary (1x)	Strongly agreed	Essential
4	Textual, audio, video (5x)	Strongly agreed	Essential
5	Yes (4x) | No (1x)	Strongly agreed	Essential

Table 47: Minimal actions.

S/N	Interview extract	Codes	Themes
1	Yes (5x)	Strongly agreed	Essential
2	4-5 steps (3) | Not sure (2)	Agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Yes (5x)	Strongly agreed	Essential

Table 48: Prompting.

S/N	Interview extract	Codes	Themes
1	Yes (4x) | Not necessary (1)	Strongly agreed	Essential
2	Yes (3x) | No (2)	Agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Yes (1x) | No (4x)	Strongly disagreed	Not essential

Table 49: Consistency.

S/N	Interview extract	Codes	Themes
1	Yes (3x) | Not sure (2x)	Agreed	Essential
2	Yes (4x) | Not sure (1x)	Strongly agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Selection (3x) | Clustering (2)	Agreed	Essential
5	Yes (4x) | No (1x)	Strongly agreed	Essential

Table 50: Spatial organisation.

S/N	Interview extract	Codes	Themes
1	Very important (4x) | Important (1x)	Strongly agreed	Essential
2	Yes (5x)	Strongly agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Ease to use (5x)	Strongly agreed	Essential
5	Yes (5x)	Strongly agreed	Essential

Table 51: Recognition rather than recall.

S/N	Interview extract	Codes	Themes
1	Yes (2x) | No (3x)	Disagreed	Not essential
2	Yes (5x)	Strongly agreed	Essential
3	Yes (5x)	Strongly agreed	Essential
4	Yes (5x)	Strongly agreed	Essential
5	Yes (4x) | No (1x)	Strongly agreed	Essential

Table 52: Remove the extraneous.

S/N	Interview extract	Codes	Themes
1	Yes (4x) | Maybe (1x)	Strongly agreed	Essential
2	Yes (4x) | No (1x)	Strongly agreed	Essential
3	Yes (2x) | No (3x)	Disagreed	Not essential
4	Yes (5x)	Strongly agreed	Essential

Table 53: Data set reduction.

S/N	Interview extract	Codes	Themes
1	Yes (4x) | No (1x)	Strongly agreed	Essential
2	Yes (5x)	Strongly agreed	Essential
3	Yes (3x) | Maybe (2x)	Agreed	Essential
4	Not sure (4x) | PCA(1x)	Neutral	Essential
5	Yes (5x)	Strongly agreed	Essential

Appendix E

Survey for the Usability evaluation.

Table 54: information coding

Information coding				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to use layers to differentiate nodes	0	5	5	**5**
It is easy to use different line types and colours to represent edges.	0	4	4	4
It is easy to use different colours to represent node clusters	0.52	4.3	4	**5**
Total	0.17	4.4	4.33	4.7

Table 55: Flexibility

Flexibility				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to arrange nodes.	0	5	5	**5**
It is easy to enlarge a node to get more details.	0	5	4	**5**
It is easy to import datasets into the system.	0	5	4	**5**
Total	0	5	4.33	**5**

Table 56:Orientation and help

Orientation and Help				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to understand how to change the settings using the panel on the right	0.52	4.33	4	5
The help feature can be understood easily	0.52	4.33	4	5
The help feature is detailed enough.	0.82	4.33	3	5

Total	0.56	4.33	3.67	5

Table 57:Minimal actions

Minimal Actions				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to select any of the visualised node.	0.4	4.8	4	5
it is easy to focus on nodes by zooming and panning.	0.5	4.7	4	5
It is easy to change the radius of the fisheye view and the appearance of nodes/edges.	0.5	4.7	4	5
Total	0.3	4.7	4.3	5

Table 58:Prompting

Prompting				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to know when a dataset file is still being processed.	0.4	4.8	4	5
It is easy to know when a change of layout is fully processed.	0	4	4	4
It is easy to know the pitfalls in the tool	0	4	4	4
Total	0.1	4.3	4	4.3

Table 56: Consistency

Consistency				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to fix inconsistencies in the features of the tool, such as colouring.	0.5	4.3	4	5
It is easy to predict the time it will take to render a dataset	0.4	4.2	4	5
It is easy to find inconsistencies in a dataset.	0.6	4	3	5

Total	0.5	4.2	3.7	5

Table 59:Spatial organisation

Spatial Organisation				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to see the differences in the nodes in the fisheye view.	0.5	4.7	4	5
The nodes in the fisheye effect are properly sized	0.4	4.8	4	5
The way the nodes are arranged makes it easier to see patterns	0	5	5	5
Total	0.3	4.8	4.3	5

Table 60:Recognition rather than recall

Recognition rather than recall				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to notice differences between datasets using the blurfisheye-visualization tool	0	5	5	5
It is easy to notice when a dataset has changed using the blurfisheye-visualization tool	0.5	4.5	4	5
It is easy to identify similarities between datasets using blurfisheye-visualization tool	0.4	4.2	4	5
Total	0.3	4.6	4.3	5

Table 61:Removing extraneous.

Removing the Extraneous				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to find unwanted nodes in the dataset.	0.6	4	3	5
It is easy to remove parts of the tools that are considered irrelevant.	0	4	4	4
It only takes a short time to remove unwanted features using the tool	0	4	4	4

Total	0.2	4	3.7	4.3

Table 60:Dataset reduction

Dataset Reduction				
	Standard Deviation	Mean	Minimum	Maximum
It is easier to render parts of a large dataset instead of all of it.	0.5	4.7	4	5
It is easy to find the irrelevant elements in a dataset.	0.4	4.2	4	5
It is easy to partition a dataset into several sections	0	4	4	4
Total	0.3	4.3	4	4.7

Table 62:Time

Time				
	Standard Deviation	Mean	Minimum	Maximum
The time taken to render large datasets is impressive.	0.4	3.8	3	4
It takes a short time to find patterns in a dataset.	0.4	4.2	4	5
It only takes a short time to remove unwanted features using the tool	0.5	4.7	4	5
Total	0.3	4.2	4	4.7

Table 63:Insight

Insights				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to judge the accuracy of a dataset when it is visualised using blurfisheye-visualization tool.	0.4	4.2	4	4
It is easy to find faults in a dataset when it is visualised using blurfisheye-visualization tool	0.5	4.3	4	5

It is easy to find inspirations in a dataset when it is visualised using the blurfisheye-visualization tool	0.4	4.2	4	5
Total	0.3	4.2	4	4.7

Table 64:Essence

Essence				
	Standard Deviation	Mean	Minimum	Maximum
It is easy to make sense of random values	0	4	4	4
It is easy to reach a conclusion from visualised data	0.4	4.2	4	5
It is easy to get the message in a dataset	0.5	4.3	4	5
Total	0.3	4.2	4	4.7

Table 65:Confidence

Confidence				
	Standard Deviation	Mean	Minimum	Maximum
Visualised network boost confidence in data when using the blurfisheye-visualization tool	0.4	4.2	4	4
Visualised data affects the possibility of accepting the data when using the blurfisheye-visualization tool	0.4	4.2	4	5
It is easy to trust data that can be seen when using the blurfisheye-visualization tool	0.5	4.7	4	5
Total	0.2	4.3	4	4.7

Evaluation Tasks for the usability test

The Task have general information but did not give too detailed to make it more rigid in examination. For example, task 1 focuses on studying the selection and drag feature of the nodes. The task did not tell the user to select a certain node rather told to select any node and record their experience.

Task 1

Load any of the datasets or select one of the sample data provided in the system and click and drag any of the nodes.

Please follow the steps to achieve the task:

- **Step 1:** Open the portal.
- **Step 2:** Select a data set from the list of data sets or upload one.
- **Step 3:** When it is fully loaded, click any of the nodes.
- **Step 4:** Drag the node that you clicked across the view.

Task 2

Select 4 nodes from a complex dataset (Yeast probabilistic Network) and change the color of each node.

Please follow the steps to achieve the task:

- **Step 1:** Open the portal.
- **Step 2:** Select the dataset labeled "Yeast probabilistic Network" from the list of datasets.
- **Step 3:** When it has fully rendered, go to the side bar and select any four (4) nodes from the dropdown menu labeled 'Select node'.
- **Step 4:** In the same side bar, click the colored shape labelled 'color of selected node' and pick a different color from the color palette.

Task 3

Select a node from a complex network, apply fisheye effect to the selected node, increase the scope of the fisheye effect, and change the color of the nodes and edges.

Please follow the steps to achieve the task:

- **Step 1:** Open the portal.
- **Step 2:** Select the dataset labeled "Yeast probabilistic Network" from the list of datasets.
- **Step 3:** When it has fully rendered, go to the side bar and select a single node from the dropdown menu labeled 'Select node'.
- **Step 4:** When the graph has rerendered, click any node other than the selected node.
- **Step 5:** Go to the side bar and increase the value of the range bar labeled range radius to 160.
- **Step 6:** Change the value of the range bar to 240.

Task 4

Select two nodes from dataset "Yeast probabilistic Network", click on either of the nodes, drag the clicked node across the canvas, then increase the scope of the fisheye effect**.**

Please follow the steps to achieve the task:

- **Step 1:** Open the portal.

- **Step 2:** Select the dataset labeled “Yeast probabilistic Network” from the list of datasets.
- **Step 3:** When it has fully rendered, go to the side bar and select a single node from the dropdown menu labeled ‘Select node’.
- **Step 4:** When the graph has rerendered, go to the side bar and click Concentric.
- **Step 5:** Click any of the selected nodes.
- **Step 6:** Drag the clicked node to the edge of the circle formed by the other node.
- **Step 7:** Go to the sidebar and increase the range radius bar to 140.

Task 5

Read and understand the help information in the side bar.

Please follow the steps to achieve the task:

- **Step 1:** Open the portal.
- **Step 2:** Click the Icon with question mark.
- **Step 3:** Read the instructions provided.

Task 6

Build a network with more than 7000 edges.

Please follow the steps to achieve the task:

- **Step 1:** Open the portal.
- **Step 2:** Select the dataset named Probabilistic Functional Integrated Network.
- **Step 3:** Go to the list of nodes and select nodes.
- **Step 4:** Continue selecting nodes from the select node menu until the number of edges exceed 7000. You can see the current number of edges on the side bar.
- **Step 5:** Click any of the rendered nodes to show all the connections of the clicked node clearly.

Task ended.

Appendix F

Interview for the evaluation of user testing.

HF1: Information coding

Table 66: HF1 information coding

Questins	Q1: To what extent was the outcome of the visualization relevant to your needs?	Q2: What challenges did you face while attempting to visualise a dataset the way you wanted it?	Q3: In your opinion, out of the five-layout algorithm, which one passed the most information about a data set?
Participants	Application allowed for good visualization. Flexible and easily tuneable	Web lagging, clunkiness of the interface. Lack of what node represented and drugging a node to slowdown the web	Grid, breadfirst, goose and circle

HF2: Flexibility

Table 66: HF2 Flexibility

Questins	Q1: To what extent did you utilize zooming and panning?	Q2: How many nodes did the visualised dataset have?	Q3: How many adjustments were you able to make to the visualised network after rendering it? Kindly state areas you felt limited.
Participants	Nice touch to the visualization. Simple, but more difficult when panning due to smaller white space	Number depends on the selected radius. 1986,3000 or 19623	Speed of loading the web Lag problem

HF3: Orientation and help

Table 67:HF3 Orientation and help

Questins	Q1: How long did it take you to understand the information displayed when you clicked the help button?	Q2: What did you understand were the criteria that must be met by any dataset to be uploaded?	Q3: Which information do you recommend should be included in the help instruction?
Participants	Straight forward	txt file	Information prompt Layout explanation

HF4: Minimal Actions

Table 68:HF4 Minimal Actions

Questins	Q1: Was it challenging to achieve the can fisheye view the way that you wanted it? If yes, kindly share your challenges.	Q2: Was it hard to change the look of the visualised network? If yes, in which areas did the tool let you down?	Q3: From your experience with the tool, do you think the tool can easily visualise any dataset that meets the criteria?
Participants	No	Undo button will make it easy More sensitivity of range radius	Yes, but need minor adjustment

HF5: Prompting

Table 69:HF5 Prompting

Questins	Q1: Which prompting do you consider most relevant in the system? Briefly explain what makes it so relevant.	Q2**:** Which prompting did you consider unnecessary in the system? Kindly explain why you consider it unnecessary.	Q3: Is there any prompting that you would like to be included? If so, please provide examples
Participants	No prompting Cluster of relevant genes Node and edges highlights	Concentric and circle None	No Ability to add notes and groups

HF6: Consistency

Table 70:HF6 Consistency

Questins	Q1: Did you observe any inconsistencies in the tool? If yes, in what way did they affect you experience with the tool?	Q2: Did you notice any difference in the rendering time of datasets?	Q3: Do you consider the visualizations of any datasets to be inconsistent with what is expected? If no, briefly point out the inconsistencies.
Participants	No	Yes Dataset can be enhanced	No

HF7: Spartial organisation

Table 71:HF7 Spatial organisation

Questins	Q1: which of the layout algorithms was the fisheye view most useful?	Q2: Which of the layout algorithms is do you consider producing the most visually appealing renderings?	Q3: Were you able to arrange the nodes in a manner that suits your purpose? If no, what changes would you recommend?
Participants	Grid, goose and breadfirst	Cose	yes

HF8: recognition rather recall

Table 702:HF8 Recognition rather recall

Questins	Q1: Did you notice anything in the data that would have gone unnoticed if not for the visualization?	Q2: Do you consider recognition of information more efficient that recalling of information? If yes, how has this tool helped in that area?	Q3: In what ways can this tool be improved to help users recognise unexpected patterns in datasets?
Participants	Possible node cluster Better analysis of concentric	Useful in recognition and remembering data	Pattern and relationship recognition

	Data part of the visualization		

HF9: Removing the extraneous

Table 73:HF9 removing the extraneous

Questins	Q1: Was there any part of the tool you wanted to remove but could not? If yes, briefly describe it	Q2: Having used the tool, do you think that there are any unnecessary features that serve no real purpose? If yes, kindly state them.	Q3: Did you find it easy to find unnecessary elements in your dataset? Kindly share your experience.
Participants	No	No	Yes

HF10: Dataset reduction

Table 74: HF10 Dataset reduction

Questins	Q1: Do you consider it necessary to render datasets in small chunks?	Q2: Did you use the tool to partition a dataset into sections? If yes, what were your challenges? If no, why not?	Q3: What features would recommend that could make it easier to reduce a dataset?
Participants	No	Yes	Snipping feature Easy deletion of uninteresting nodes

HF11: Time

Table 75:HF1 Time

Questins	Q1: Do you consider it to be time consuming to use this tool? If yes, kindly state your reasons.	Q2: How many seconds did it take to render the largest dataset you have? Do you consider the time to be good enough.	Q3: Which aspects of the tool took the most of your time? In what ways can the time spent in those aspects be reduced?
Participants	No	More than 30 secs	Loading the dataset

			Layout changing and rendering

HF12: Insights

Table 76: HF12 Insights

Questins	Q1: What were the issues you were able to figure out with the aid of the visualised network?	Q2: Were you able to see new patterns emerging from any of the visualised networks? If so, what type of patterns?	Q3: To what extent can you act on any of the issues?
Participants	Cluster of related genes Protein interaction for heart disease treatment	Cluster of important nodes	Test target protein Further experimentation

HF13: Essence

Table 71: HF13 Essence

Questins	Q1: Were you able to see special meaning in any of the datasets that you visualised? If yes, briefly talk about it.	Q2: What could be added to the tool to make it easier for users to find meaning in datasets?	Q3: Were there any features or layouts in the tool that you think would not be useful for gaining insight from the data? Kindly state your reason.
Participants	No	Interaction Analysis of specific edges Two layouts to be side by side	No

HF: Confidence

Table 72:HF14 Confidence

Questins	Q1: Do you think it is rational to confidently make a decision based on the visualization of a dataset? Kindly state your reasons.	Q2: To what extent can you make binding decision about a dataset based on the visualization of the dataset?	Q3: In your opinion, what could be done to increase users' confidence in the visualizations

			produced by this tool?
Participants	Identify cluster. Dataset is clear	Interactions are clear. Extensive	Design software intuition based on users need. Easy accessibility of data on edges. Re-downloading of specific view of interest.

Appendix G

The styles referenced in the algorithm

inRadiusStyle:

'background-color': nodeColor,

'opacity': 1,

'border-color': '',

'border-width': '',

'border-style': '',

'color': nodeLabelColor,

'text-outline-color': nodeLabelColor,

'text-outline-width': .5,

'font-size': 12,

'height': '32px',

'width': '32px'

blurredStyle:

'background-color': nodeColor,

'opacity': 0.2,

'border-color': '',

'border-width': '',

'text-outline-width': 0,

'font-size': 0,

'border-style': '',

'height': '24px',

'width': '24px'

selectedNodeStyle:

'background-color': selectedColor,

'opacity': 1

clickedStyle:

'opacity': 1,

'border-color': clickedBorderColor,

'border-width': '4px',

'font-size': 12,

'border-style': 'solid',

'text-outline-width': .5,

'height': '50px',

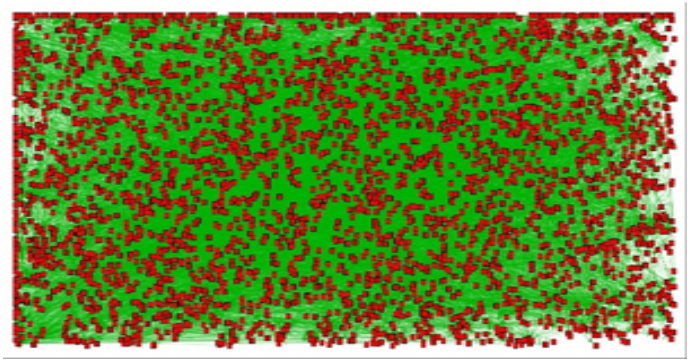
'width': '50px',

NOTE: The parameters listed below are variable because their values can be changed from the tool bar: nodeColor, nodeLabelColor, nodeLabelColor, clickedBorderColor, selectedColor.

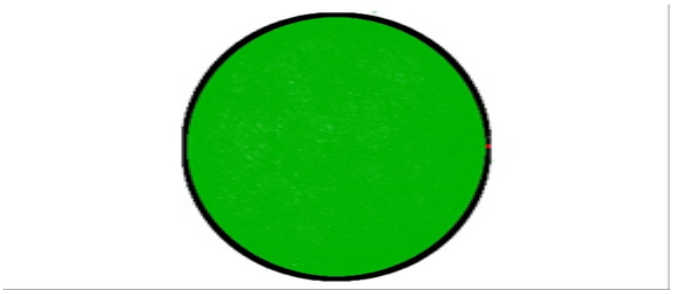
Appendix H

Medusa layouts

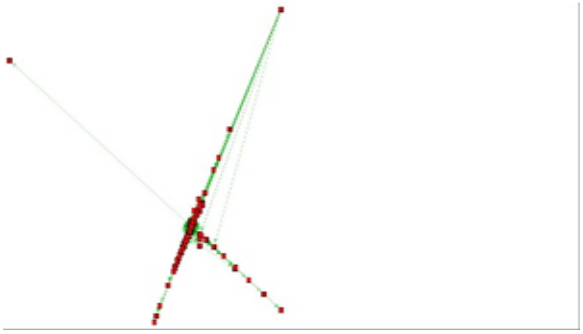
Random



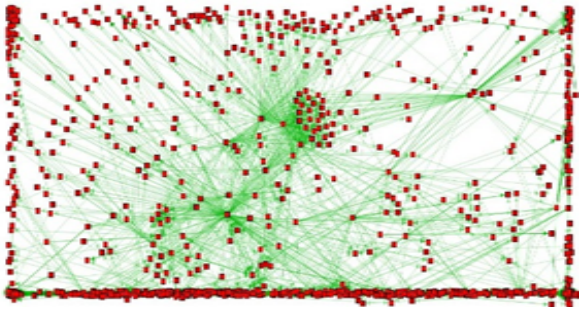
Circular



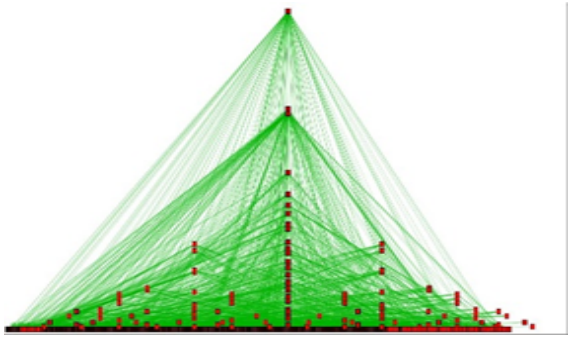
Grid



Fruchterman-Reingold



Hierarchical



K-Means

Spectral clustering