A task-based evaluation methodology for visual representation of dynamic networks

Pablo Camarillo-Ramirez
Western Institute of Technology and Higher Education
Tlaquepaque, Jalisco, Mexico
ng724453@iteso.mx

Francisco Cervantes-Alvarez
Western Institute of Technology and Higher Education
Tlaquepaque, Jalisco, Mexico
fcervantes@iteso.mx

Luis F. Gutiérrez-Preciado
Western Institute of Technology and Higher Education
Tlaquepaque, Jalisco, Mexico
lgutierrez@iteso.mx

ABSTRACT

Current evaluation approaches for visualization strategies of dynamic networks are focused on maintaining the mental map of the network over the time or keeping a certain shape to make it easy to navigate, however the available tools for analyzing temporal network have not been evaluated in terms of how easy to use they are to perform exploratory data analysis tasks with dynamic networks. In this work we present an evaluation methodology that guides the usability assessment of software tools used to analyze dynamic networks by using the standard ISO 9241-11. This methodology has been applied successfully with two popular open source tools used to analyze temporal networks.

KEYWORDS

dynamic networks, graph drawing, usability, drawing evaluation

1 INTRODUCTION

Due to its impact on business and data analysis, the analysis of networks has become one of the most prominent research areas in recent years. The function of a network is to represent links between entities, revealing the structure and nature of relationships in data. Network visualization is one of the main means of exploratory graph analysis [24] and it has becomes relevant for business when network visualization supports the decision-making process[5]. For those problems with connected data which is represented as network, a good visual representation is highly required to perform successfully exploratory data analysis (EDA). To determine whether a network drawing technique is good or not, several approaches have been proposed such as those approaches focussed on characteristics of network layout [7], [8], [10], clusters in graph [17] or network shape [6]. All of these strategies are focussed on visualizing static networks only. Another type of networks that recently are becoming relevant in the EDA field are those that changes over time, known as dynamic networks. The most common ways to visualize a dynamic network includes animations, timeline of changes or a hybrid visualization [4]. For these kind of networks, most of the existent evaluation strategies are focused on preserving the mental map over the time [23],[3]. As a matter of fact, Beck et al. [4] conclude that most of these evaluation approaches are not necessarily involving users, hence the motivation of this paper to propose a user-centred evaluation methodology rather than network structure or aesthetics properties.

EDBT 2020, March 30-April 2, 2020, Copenhagen, Denmark

ACM ISBN .
https://doi.org/10.1145/nnnnnnn.nnnnnnn

The importance of evaluating the usability of a software lies on managing the potential risks that can arise from inappropriate outcomes of interaction. For instance, an undesired outcome of EDA tasks might be the waste of computing resources or user time to perform analysis tasks. In this work we present an evaluation methodology focussed on the usability of tools that support EDA tasks with dynamic networks.

The rest of this paper is organized as follows. In the Section 2 we explore works that inspired the development of the methodology proposed in this work. We describe briefly EDA and how a connected data structure can be useful to perform this kind of analysis. In the Section 3 we provide details about the methodology proposed. In the Section 4 it is shown how the methodology described is used to evaluate the usability of Gephi and Cytoscape for temporal tasks on dynamic networks. Finally in the Section 5 we conclude about the advantages and improvements needed to the methodology proposed based on the results presented.

2 RELATED WORK

2.1 Exploratory Data Analysis with dynamic networks

The exploratory data analysis consists on finding answers to numerous questions about data [2]. To obtain these answers analysts use mainly software tools. In [22] authors conclude that EDA is about hypothesis generation rather than hypothesis testing. This definition, however, does not take into account the questions that analysts may have and the process to solve them. Another well-known work that defines EDA is the *Information Seeking Mantra* by Ben Shneiderman [19] that generalizes the EDA process into three steps: (1) Overview first, (2) filter, and then (3) details-on-demand. In summary, this definition indirectly states that EDA is the process to find what items are *interesting* and deserve further examination. According to Andrienko and Andrienko [2], visualization systems are frequently employed to support EDA tasks.

2.2 Task taxonomies for temporal EDA tasks

A task can be understood as an entity formed by two components: target and constraints. A target refers to the unknown information to be obtained, and the constraints points out to the known conditions that system needs to fulfill; a task therefore involves finding a target given a set of constraints.

According to [13] task taxonomies play a vital role in the design and evaluation of visualization systems, because they reveal and categorize the application needs. This categorization supports the process to design a system that provides an appropriate visual representation of a dynamic network to complete exploratory tasks.

There are many works explaining different aspects of an exploratory task on a static network. Lee et al. [14] define a graph visualization task taxonomy and classified the tasks as: (1) Topology-based (adjacency, accessibility, common connection, connectivity) (2) Attribute-based (On the nodes and On the links), (3) Browsing (Follow path and Revisit) and (4) Overview, a compound exploratory task to get estimated values quickly.

Shneiderman and Aris [20] define a task taxonomy of networks as a collection of task associated to (1) Basic networks (unlabeled nodes and undirected links), (2) Node/Link labels, (3) Directed networks, and (4) Node/Link attributes.

Along with these entities, authors propose a list of tasks specifically associated to basic networks (count number of nodes, compute degree for every node, find betweenness centrality, etc.), but they conclude there are an unlimited number of tasks that could be defined.

On the other hand, for temporal analysis, Yi et al. [25] propose a task classification that visualization techniques should support to perform temporal social network analysis (TSNA): temporal changes at the global level, temporal changes at the subgroup level and temporal associations among nodal and level attributes.

2.2.1 A task taxonomy for network evolution analysis. For temporal analysis of networks, analysts are interested in three different targets: entities, properties, and temporal features. Constraints are the (limited) resources such as display size or I/O devices used to perform exploratory tasks [1]. Entities include node/link, group or network. The properties include both structural properties and domain attributes. Finally, temporal features consist of those features that answer the question about the network's evolution. In fact, Jae-wook et al. [1] take these three dimensions to define a design space [1] to formulate a task taxonomy for temporal networks. This design space and some examples of temporal tasks are shown in the Table 1.

2.2.2 A task taxonomy for temporal graph visualization. Another taxonomy proposed for tasks on a temporal network is presented in [13]. This approach covers not only temporal networks, but also static networks, multivariate graphs, and graph comparison. The main idea of that work is to extend the Andrienko framework [2]. The Andrienko framework consists in data model and task framework. The task framework applies the task definition previously mentioned (targets and constraints). The data model identifies the data items that might participate as target or constraint. However, one of the main limitations of the Adrienko's framework is that it does not consider graph data. For example, the information of an edge is difficult to model under the data model presented by such framework. The extension proposed by Kerracher et al. [13] includes the structural tasks that considers the questions associated to relational tasks for the networked data.

2.3 Evaluation approaches of dynamic network visualization

In this section we will discuss some of the most popular strategies to analyze the quality of the visual representation of the dynamic networks. We can distinguish two main approaches to evaluate visualization systems for dynamic networks: those focussed on the importance of maintaining the mental map and those concentrated on profiling the visualization in terms of network structure or layout. The most common way to visualize a network is by using a *node-link* diagram to represent entities and their connections. Another way to visualize a network is by using adjacency matrices where the nodes are represented as rows and columns and a colored intersection encodes an edge. The approaches discussed in this section are concentrated on the mental map preservation by using *node-link* diagrams.

One of the most used criterion to determine whether a visual representation algorithm of a network is *good* or not is if it can preserve the mental map. The intention of the mental map preservation is to keep the network layout over time in order to offload the cognitive effort required to comprehend the information contained in the network [3].

One of the works focussed on maintaining the mental map of a temporal network is the Hyperbolic temporal layout proposed by Cengiz and Balcisoy [23] that represents the evolution of relations among network actors and structural patterns of a social network.

On the other hand, Archambault and Purchase [3] have conducted some experiments focussed on the human factors in temporal network drawing rather than algorithmic considerations. They found that preserving the mental map is not always helpful when performing tasks on dynamic networks.

2.4 Usability evaluation

The term usability can be understood as the software capability of being used. One of the most important benefits of having a software highly usable might be a little time on performing a task.

We can distinguish two approaches that might help us to outline the evaluation methodology proposed in this paper: A consolidated model called Quality in Use Integrated Measurement (QUIM) proposed by Seffah et al. [18] and the standard ISO 9241-11 [11].

2.4.1 Quality in Use Integrated Measurement. The model described in [18] includes 10 usability factors: (1) Efficiency, (2) Effectiveness, (3) Productivity, (4) Satisfaction, (5) Learnability, (6) Safety, (7) Trustfulness, (8) Accessibility, (9) Universality, and (10) Usefulness. These factors are decomposed into 26 sub-factors which are further-decomposed into 127 specific usability metrics. Authors proposal included an editor tool [2] that supports the activities to obtain usability measurement. Unfortunately this editor is not longer available.

2.4.2 ISO 9241-11. The aforementioned model was inspired by analyzing several standards, frameworks and models previously proposed. One of these standards is the ISO 9142-11 [11]. This standard measures the usability of a software (or hardware) in terms of efficiency, effectiveness and satisfaction in a context of use. The context of use can be understood as the users, tasks equipment (software and materials), and the physical and social environment in which a product is used.

2.5 Software tools and libraries to visualize networks

2.5.1 Cytoscape. Cytoscape[3] is an open source software for visualizing complex networks. It is a software developed by Cytoscape Consortium and it is founded by the U.S. National Institute of General Medical Sciences (NIGMS). Its main goal was

[1] A design space is a multidimensional combination and interaction of input variables and process parameters that have been demonstrated to provide assurance of quality.

[2] http://rana.cs.concordia.ca/odusim

[3] https://cytoscape.org

Table 1: Design space of temporal taxonomy proposed by Jae-wook et al. [1]

			Entities Node or Link / Group / Network
	Individual temporal features	Single Occurrences	Examine Network’s Clustering Coefficient
		Birth or Death	Find when the tendency #freebiefriday appears
		Replacement	Find when changes the in-degree of #firdayfeeling tendency
Temporal features	Shape of changes features	Growth & Contraction	Observe the Network’s growth (forward)
		Convergence & Divergence	Observe if the Clustering Coefficient converges at some time point
		Stability	Compare the stability states between starting and ending point
		Repetition	Observe the repeated relationship between tendencies #fridaymotivation and #fridayfeeling
		Peak or Valley	Observe the Clustering Coefficient peaks or valleys for the entire network
	Rate of changes features	Fast & Slow	Observe the speed of tendencies creation
		Accelerate & Decelerate	Identify the acceleration for tendencies creation

to offer a tool for biological research, however nowadays it is a general tool for complex network analysis and visualization. The architecture of Cytoscape offers the capability to increase functionalities by developing adding plugins. Currently there are ten available apps in the Cytoscape marketplace under the *Network dynamics* category.

2.5.2 Gephi. Gephi [16] is another open software tool useful to explore and understand graphs. It is an interactive visualization and exploration platform for many kinds of networks and complex systems, dynamic and hierarchical graphs. The goal is to help data analysts to form a hypothesis, intuitively discover patterns, isolate structure singularities or faults during data sourcing. Its last version supports visualize dynamic networks by using a continuous representation of connected data.

2.5.3 NTDV. The Network Dynamic Temporal Visualization [21] is a package for language R to visualize dynamic networks. Its last version was released on May 2019 and it provides capabilities to analyze and visualize networks such as birth, death, and reincarnation of objects in the network over time. It supports discrete and continuous representation for time, which allows to visualize many kinds of datasets with temporal connected data. The NDTV package generates network movies or interactive HTML5 animations, timelines and other visualizations ways of dynamic networks.

2.5.4 KeyLines. KeyLines [4] is a SDK developed by Cambridge Intelligence company for building web applications to perform network visualization. One of the main features of this SDK is the capability that offers to manage dynamic networks with its time bar. With this time bar, users can filter data by time and date, observe network evolution and perform any EDA task. Another key feature of KeyLines is the map mode that enables the functionality to visualize networks on maps, and thus perform spatial analysis.

[4]https://cambridge-intelligence.com/keylines/

2.5.5 ReGraph. Part of the suite provided by Cambridge Intelligence, ReGraph [5] is a library of React components and analysis functions for client-side network visualization.

3 EVALUATING VISUALIZATION OF DYNAMIC NETWORKS

For EDA with temporal networks, only the experiments described in [3] take into account the user experience of a visual representation of temporal networks. These experiments are focussed on the importance of the mental map preservation for dynamic graph drawing. We propose a new methodology based on ISO 9241-11 to evaluate the usability of a visualization system in terms of effectiveness, efficiency and satisfaction in a context of use. This methodology can be summarized as follows:

- Establish the context of use: (1) obtain or generate the time-evolving network in format required by the software tool to be evaluated, (2) Define a subset of EDA tasks that the software tool should be capable to perform, (3) select a group of users or analysts that should complete the EDA tasks, and (4) fix the layout algorithm that will be observed every time slice.
- Analyze EDA tasks selected: (1) measure time every user takes to complete the task (if he/she does), (2) apply a satisfaction questionnaire after finishing every one of these tasks, and (3) compute Effectiveness and Efficiency metrics.

3.1 Effectiveness of dynamic network visualization tools

The effectiveness metric can be obtained by using the completion rate equation 1. In our context, given an EDA task, it is asked to a set of analysts to complete the task under same conditions. The

[5]https://cambridge-intelligence.com/regraph/

more EDA tasks are completed, the higher is the effectiveness score for this task.

$$Effectiveness = \frac{N}{T} \quad (1)$$

Where N represents the number of tasks completed successfully and T stands for the total number of tasks undertaken.

3.2 Efficiency of dynamic network visualization tools

One of the main motivations to evaluate the usability of the current software tools that support EDA tasks is to measure the time employed to complete an EDA tasks with a dynamic network. Said that, we compute the efficiency of a software tool for dynamic network analysis in terms of the time needed to complete a task. In the equation 2 if is shown how the Efficiency can be calculated.

$$Efficiency = \frac{\sum_{j=1}^{R}\sum_{i=1}^{N}\frac{n_{ij}}{t_{ij}}}{NR} \quad (2)$$

Where N is the number of tasks, R is the number of users, if the user successfully completes the $i-th$ task $n_{ij} = 1$ otherwise $n_{ij} = 0$ and t_{ij} represents the time spent by $j-th$ user to complete the $i-th$ task.

3.3 User satisfaction of dynamic network visualization tools

The strategy suggested to assess the user satisfaction is to apply the ASQ questionnaire [15] after completing every EDA task. This questionnaire surveys the user satisfaction in terms of task difficulty, time spent to complete the task and usefulness of the documentation provided by the software to complete the task. What we propose is to change the original 7-point scale to a 5-point scale because after expose the original questionnaire to some users, they suggested us to reduce the number of options. The 5-point scale resultant is as follows:

(1) Strongly agree
(2) Agree
(3) Neutral
(4) Disagree
(5) Strongly disagree

4 CASE OF STUDY WITH CYTOSCAPE AND GEPHI

In order to show how the methodology proposed can be applied we are going to evaluate two open source tools that support the EDA of temporal networks: [16] and Cytoscape. These tools were selected because both are open source projects and once evaluated they can be improved by the open source community itself. The methodology proposed is a guide to obtain effectiveness, efficiency and satisfaction in a context of use.

4.1 Establish the context of use

4.1.1 Generate dynamic network. The dynamic network presented to the users represents the evolution over 99 minutes of a sample of 142 posts on Twitter [6]. Every node represent a tendency or hashtag mentioned in the post: two tendencies are related or connected if they are mentioned in the same post. In order to add dynamics to this dataset, the timestamp is used to create the time points and thus all tendencies created or connected with a shared timestamp are observed in the same time point. With this approach it is possible to generate a dynamic network from this sample of *tweets*. In the Figure 1 it is shown the static data model of the network that is being visualized.

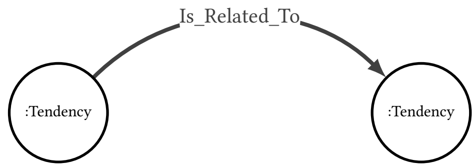


Figure 1: Data model of the dataset

4.1.2 Define a subset of EDA tasks. Based on the task taxonomy for network evolution analysis [1] we define the next subset of tasks as part of the context of use for the usability tests. This taxonomy has been selected to perform the case study because of its clear categorization of tasks and the number of examples provided by the original authors. The tasks selected are the intersection of those tasks that can be completed by using the two tools we are evaluating in these study:

(1) **BD01**: Determine the time point when the tendency #freebiefriday appears (Birth\Death)
(2) **GrCtr01**: Observe the Network's growth (Growth & Contraction)
(3) **GrCtr02**: Observe the Network's contraction (backward) (Growth & Contraction)

4.1.3 Select a group of analysts. The users selected to complete the EDA tasks are people that is involved (or interested) in network analysis. Specifically, the population selected is interested on analyzing networks that changes over time. Eleven users performed the EDA tasks in the given context of use.

4.1.4 Fix the layout algorithm. For every tool it was fixed a different layout algorithm. For Cytoscape it was fixed the Kamada-Kawai [12] algorithm and for Gephi it was fixed the Frunchterman Reingold algorithm [9].

4.2 Analyze EDA tasks

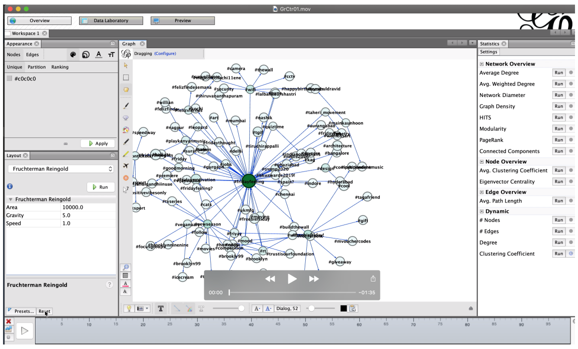
Once established the context of use, the core of the study is the observation of the user experience on performing the aforementioned EDA tasks by using two different software tools.

4.2.1 Measure time to complete every task. The entire session was recorded, from the begin of the tasks until the user was notified that heRshe has completed the task. The goal of recording every session is not only to measure the time spent but also to observe whether the user could or not complete the given task. By observing the duration of the recording, it can be obtained the time employed for every user to complete the task (See Figure 2)

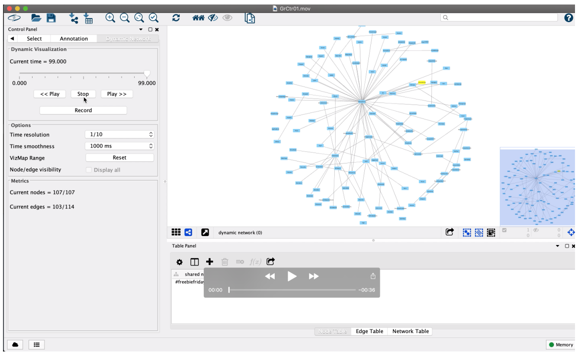
4.2.2 Apply satisfaction questionnaire. After competing every task, all users were asked to complete the questionnaire mentioned in the section 3.

4.2.3 Compute metrics. By using the equations 2 and 1, the Efficiency and Effectiveness metrics can be computed respectively.

[6] https://www.trackmyhashtag.com/historical-twitter-data



(a) Observing network evolution with Gephi



(b) Observing network evolution with Cytoscape

Figure 2: Case of study to measure time spent on EDA tasks

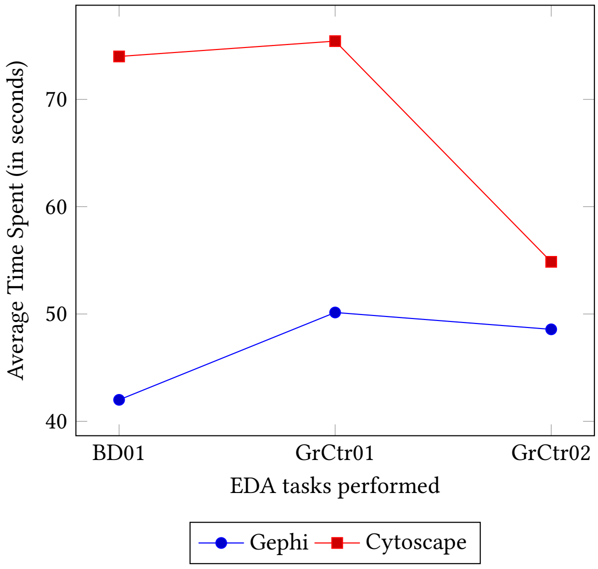


Figure 3: Temporal analysis efficiency for Cytoscape and Gephi

4.3 Results

All users could complete successfully the three tasks analyzed with both Cytoscape and Gephi. The task efficiency (with the given context of use) is 100%: all users completed the tasks with a reasonable amount of time.

For the task Efficiency, there is a clear difference between Cytoscape and Gephi for all tasks . In general, it can be observed that Cytoscape is less efficient than Gephi: 43.24% for BD01, 33.52% for GrCtr01, and 11.46% for GrCtr02 task. In the Figure it 3 can be observed the average time spent to complete the analyzed tasks.

Analyzing results obtained from the task that involves finding when appears a specific tendency in the network's timeline (BD01), we can observe that 57% of users agree with the ease of completing this task by using Gephi. For the same task, users spent in average 43.24% less time to fulfil the task with Gephi. This tendency is consistent with the rest of tasks and their satisfaction results.

One interesting finding in the task that involves observing the network contraction over time (GrCtr02). For the analyzed tasks, this is the only task where users expressed a better satisfaction of using Cytoscape. In terms of task difficulty, 57% of users consider easy to complete the GrCtr02 task, meanwhile 43% disagree with the difficulty to complete this task with Gephi, even when the average time to complete this task by using Gephi was 11% better than the time spent with Cytoscape. Probably this results is caused because the user interface of Cytoscape clearly shows the options to complete this tasks and Gephi requires more inputs to get the same animation.

Another way to interpret the results obtained is to analyze the satisfaction results for every software tool independently. For example, for EDA tasks performed by using Gephi, it can be observed a correlation between satisfaction expressed with the time spent for every task and the how difficult users found every task. When users expressed a positive experience (or neutral), they also agree with the time spent to complete these tasks.

Finally, we can analyze the satisfaction results in terms of information provided by the software interface to complete temporal EDA tasks. If we compare the results obtained from Cytoscape and Gephi, there is a notorious difference between the user satisfaction between these two tools. For Cytoscape users expressed a neutral or positive experience. However, for Gephi at least 14% of users expressed they strongly disagree with the information provided to complete the analyzed tasks.

5 CONCLUSIONS AND FUTURE WORK

The methodology presented in this work shows an effective way to evaluate software tools that supports EDA with temporal networks based on the user experience. Results obtained shows that the evaluation can be performed independently by analyzing correlations between satisfaction and effectiveness data. In addition to, this methodology can be used to compare two or more software tools and to guide the improvement process of them.

The set of tasks proposed in the taxonomies analyzed in Section 2 do not consider large graphs and we consider that a new taxonomy (or extension) should be proposed to cover EDA with large temporal networks. The future task taxonomy needs take into account the navigation capabilities offered by devices used to fulfil EDA tasks such as touch-screen devices.

Regarding to the software that supports EDA, it should consider that temporal tasks do not depend of a good animation. For instance, to analyze the shape of changes another visual components like timeline charts are might be helpful. Actually, for labeled graphs many visual tools are required to navigate, explore and analyse successfully temporal data.

Table 2: Satisfaction results for temporal tasks

		BD01		GrCtr01		GrCtr02	
		Cytoscape	Gephi	Cytoscape	Gephi	Cytoscape	Gephi
Task difficulty	Strongly agree	0%	0%	29%	0%	57%	0%
	Agree	14%	57%	29%	71%	29%	57%
	Neutral	29%	43%	14%	0%	0%	0%
	Disagree	43%	0%	29%	29%	14%	43%
	Strongly disagree	14%	0%	0%	0%	0%	0%
Time spent to complete the task	Strongly agree	14%	14%	29%	29%	43%	14%
	Agree	29%	29%	43%	43%	29%	43%
	Neutral	29%	57%	14%	0%	29%	29%
	Disagree	14%	0%	14%	29%	0%	14%
	Strongly disagree	14%	0%	0%	0%	0%	0%
Information provided to complete the task	Strongly agree	0%	0%	14%	0%	14%	0%
	Agree	29%	29%	14%	71%	29%	43%
	Neutral	57%	29%	57%	14%	57%	29%
	Disagree	14%	29%	14%	0%	0%	14%
	Strongly disagree	0%	14%	0%	14%	0%	14%

ACKNOWLEDGMENTS

To those users and analysts that help us to complete the usability tests. To the National Science and Technology Council of Mexico for the scholarship granted to the authors to develop this study.

REFERENCES

[1] Jae-wook Ahn, Catherine Plaisant, and Ben Shneiderman. 2013. A task taxonomy for network evolution analysis. *IEEE transactions on visualization and computer graphics* 20, 3 (2013), 365–376.

[2] Natalia Andrienko and Gennady Andrienko. 2005. *Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach.* Springer-Verlag, Berlin, Heidelberg, DE.

[3] Daniel Archambault and Helen C. Purchase. 2013. The "Map" in the mental map: Experimental results in dynamic graph drawing. *International Journal of Human-Computer Studies* 71, 11 (2013), 1044 – 1055. https://doi.org/10.1016/j.ijhcs.2013.08.004

[4] Fabian Beck, Michael Burch, Stephan Diehl, and Daniel Weiskopf. 2017. A Taxonomy and Survey of Dynamic Graph Visualization. *Comput. Graph. Forum* 36, 1 (Jan. 2017), 133–159. https://doi.org/10.1111/cgf.12791

[5] Richard Brath and David Jonker. 2015. *Graph analysis and visualization: discovering business opportunity in linked data.* John Wiley & Sons, Indianapolis, IN, US.

[6] Peter Eades, Seok-Hee Hong, An Nguyen, and Karsten Klein. 2017. Shape-Based Quality Metrics for Large Graph Visualization. *J. Graph Algorithms Appl.* 21, 1 (2017), 29–53.

[7] Y. Frishman and Ayellet Tal. 2004. Dynamic Drawing of Clustered Graphs. In *IEEE Symposium on Information Visualization.* IEEE, Austin, TX, US, 191–198. https://doi.org/10.1109/INFVIS.2004.18

[8] Y. Frishman and A. Tal. 2008. Online Dynamic Graph Drawing. *IEEE Transactions on Visualization and Computer Graphics* 14, 4 (July 2008), 727–740. https://doi.org/10.1109/TVCG.2008.11

[9] Petr Gajdoš, Tomáš Ježowicz, Vojtěch Uher, and Pavel Dohnálek. 2016. A parallel Fruchterman–Reingold algorithm optimized for fast visualization of large graphs and swarms of data. *Swarm and Evolutionary Computation* 26 (2016), 56 – 63. https://doi.org/10.1016/j.swevo.2015.07.006

[10] T. E. Gorochowski, M. di Bernardo, and C. S. Grierson. 2012. Using Aging to Visually Uncover Evolutionary Processes on Networks. *IEEE Transactions on Visualization and Computer Graphics* 18, 8 (Aug 2012), 1343–1352. https://doi.org/10.1109/TVCG.2011.142

[11] ISO. 2018. *Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts.* Standard. International Organization for Standardization, Geneva, CH.

[12] T. Kamada and S. Kawai. 1989. An Algorithm for Drawing General Undirected Graphs. *Inf. Process. Lett.* 31, 1 (April 1989), 7–15. https://doi.org/10.1016/0020-0190(89)90102-6

[13] N. Kerracher, J. Kennedy, and K. Chalmers. 2015. A Task Taxonomy for Temporal Graph Visualisation. *IEEE Transactions on Visualization and Computer Graphics* 21, 10 (Oct 2015), 1160–1172. https://doi.org/10.1109/TVCG.2015.2424889

[14] Bongshin Lee, Catherine Plaisant, Cynthia Sims Parr, Jean-Daniel Fekete, and Nathalie Henry. 2006. Task Taxonomy for Graph Visualization. In *Proceedings of the 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization (BELIV '06).* ACM, New York, NY, US, 1–5. https://doi.org/10.1145/1168149.1168168

[15] James Lewis. 1991. Psychometric evaluation of an after-scenario questionnaire for computer usability studies: The ASQ. *SIGCHI Bull.* 23 (01 1991), 78–81. https://doi.org/10.1145/122672.122692

[16] Bastian Mathieu, Sebastien Heymann, and Mathieu Jacomy. 2009. Gephi: An Open Source Software for Exploring and Manipulating Networks. In *Proceedings of the Third International ICWSM Conference (2009).* The AAAI Press, San Jose, CA, US, 361–362. http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154

[17] Amyra Meidiana, Seok-Hee Hong, Peter Eades, and Daniel Keim. 2019. A Quality Metric for Visualization of Clusters in Graphs. In *Graph Drawing and Network Visualization*, Daniel Archambault and Csaba D. Tóth (Eds.). Springer International Publishing, Cham, 125–138.

[18] Ahmed Seffah, Mohammad Donyaee, Rex Kline, and Harkirat Padda. 2006. Usability measurement and metrics: A consolidated model. *Software Quality Journal* 14 (06 2006), 159–178. https://doi.org/10.1007/s11219-006-7600-8

[19] B. Shneiderman. 1996. The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages.* IEEE, Boulder, CO, US, 336–343. https://doi.org/10.1109/VL.1996.545307

[20] Ben Shneiderman and Aleks Aris. 2006. Network Visualization by Semantic Substrates. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (Sept. 2006), 733–740. https://doi.org/10.1109/TVCG.2006.166

[21] T. Butts, Carter and Leslie-Cook, Ayn and N. Krivitsky, Pavel and Bender-deMoll, Skye and Almquist, Zack . 2019. Network Dynamic Temporal Visualization. http://statnet.org

[22] John W. Tukey. 1977. *Exploratory Data Analysis.* Addison-Wesley, Reading, MA, US.

[23] Uraz Cengiz Turker and Selim Balcisoy. 2014. A visualisation technique for large temporal social network datasets in Hyperbolic space. *Journal of Visual Languages & Computing* 25, 3 (2014), 227 – 242. https://doi.org/10.1016/j.jvlc.2013.10.008

[24] Tatiana Von Landesberger, Arjan Kuijper, Tobias Schreck, Jörn Kohlhammer, Jarke J van Wijk, J-D Fekete, and Dieter W Fellner. 2011. Visual analysis of large graphs: state-of-the-art and future research challenges. *Computer graphics forum* 30, 6 (2011), 1719–1749.

[25] Ji Soo Yi, Niklas Elmqvist, and Seungyoon Lee. 2010. TimeMatrix: Analyzing Temporal Social Networks Using Interactive Matrix-Based Visualizations. *International Journal of Human–Computer Interaction* 26, 11-12 (2010), 1031–1051. https://doi.org/10.1080/10447318.2010.516722 arXiv:https://doi.org/10.1080/10447318.2010.516722