On the Integration of Graph Exploration and Data Analysis: The Creative Exploration Toolkit

Stefan Haun[1], Tatiana Gossen[1], Andreas Nürnberger[1],
Tobias Kötter[2], Kilian Thiel[2], and Michael R. Berthold[2]

[1] Data and Knowledge Engineering Group,
Faculty of Computer Science, Otto-von-Guericke-University, Germany
http://www.findke.ovgu.de
[2] Nycomed-Chair for Bioinformatics and Information Mining
University of Konstanz, Germany
http://www.inf.uni-konstanz.de/bioml

Abstract. To enable discovery in large, heterogenious information networks a tool is needed that allows exploration in changing graph structures and integrates advanced graph mining methods in an interactive visualization framework. We present the Creative Exploration Toolkit (CET), which consists of a state-of-the-art user interface for graph visualization designed towards explorative tasks and support tools for integration and communication with external data sources and mining tools, especially the data-mining platform KNIME. All parts of the interface can be customized to fit the requirements of special tasks, including the use of node type dependent icons, highlighting of nodes and clusters. Through an evaluation we have shown the applicability of CET for structure-based analysis tasks.

1 Introduction

Today's search is still concerned mostly with keyword-based searches and the closed discovery of facts. Many tasks, however, can be solved by mapping the underlying data to a graph structure and searching for structural features in a network, e.g. the connection between certain pages in Wikipedia[1] or documents closely related to a specific document, which may be defined by the exploration task itself, i.e. documents mentioning each other, documents which are term-related, etc. Exploring a hyperlink structure in a graph representation enables these tasks to be fulfilled much more efficiently. On the other hand, graph visualization can handle quite large graphs, but is rather static, i.e. the layout and presentation methods calculate the graph visualization once and are not well suited for interactions, such as adding or removing nodes. For example, one of the well known graph layout methods, the *Spring Force Layout*, can yield very

[1] http://www.wikipedia.org

M.R. Berthold (Ed.): Bisociative Knowledge Discovery, LNAI 7250, pp. 301–312, 2012.

chaotic results when it comes to small changes in the graph, leading to a completely different layout if just one node is removed [13]. Since a user's memory is strongly location-based [16] and relies on the node positions during interaction with the graph, such behavior is not desirable.

With the *Creative Exploration Toolkit (CET)*, we present a user interface with several distinct features: Support of interactive graph visualization and exploration, integration of a modular open source data analytics system, and easy configuration to serve specific user requirements.

In the following sections, we describe these features in more detail. We start with a short overview on the state of the art in graph interaction and visualization (Sect. 2), describe the explorative user interface (Sect. 3) and the XMPP communication (Sect. 4.1), discuss the integrated (graph)mining methods (Sect. 4), present a first evaluation of the tool by a user study (Sect. 5) and finally discuss some future work.

2 State of the Art in Graph Interaction and Visualization

Related work can be found in the field of graph visualization and graph layouting. Cook and Holder, although mostly concerned with graph mining, provide a good overview on the state of the art and current systems [3]. For a general overview there are several surveys on graph visualization available (c.f. [14], [4], [18]). According to [5], there are three major methods for graph layouting: *force-directed*, *hierarchical* and *topology-shape-metrics*, where the force directed method introduced by [7] is most used today, despite its disadvantageous behavior in interactive systems [14]. Special visualizations can be used to accommodate data specific features such as time lines: [6] introduces a 2.5D time-series data visualization, which uses stacks to represent time-dependent advances in data series. A large number of visualization systems is available. Approaches tailored to web searching and the visualization of hypermedia structures can be found among the web meta-search clustering engines (Vivsmo[2], iBoogie[3], SnakeT[4], WhatsOnWeb[5]) and in the field of semantic wikis (iMapping Wiki [10]).

However, existing layout and visualization methods do not take continuous graph development in the exploration scenario or the heterogeneousity of visualized information networks and their data sources into account. Besides the grave differences between data mining tools and human-computer interaction (see [11]) and the aforementioned shortcomings in continuous visualization of changing graph structures, a loosly-coupled, but efficient integration between network-providing services and visualization tools is often not available.

[2] http://vivismo.com/
[3] http://www.iboogie.com/
[4] http://snaket.di.unipi.it
[5] http://whatsonweb.diei.unipg.it

3 The Creative Exploration Toolkit

The Creative Exploration Toolkit (CET) is a user interface that visualizes the graph—derrived from an information network—and allows interaction with it. The global design, shown in Figure 1, consists of

- a *dashboard* at the top, where the controls are located,
- a *logging area*, below, to show information on running processes and the tool status,
- a *sidebar* on the right which displays detailed information about a node,
- and the *workspace* in the center, which is used for visualization.

We currently use the *Stress Minimization Layout* [15] to determine the initial graph layout, followed by an overlap removal [8]. Nodes can be moved to create certain arrangements, selected for further action, and expanded by double-clicking them. On expansion the surrounding graph structure—obtained from the data provider—is added to the visualization. Additionally, the user may issue keyword-based queries. The corresponding results consists of graphs and can be visualized as well. Subsequent query results are added to the graph, enabling the user to explore the graph itself and the structures between the query results. Additionally, there is support for node handling such as a list of all available or all marked nodes, a keyword search for specific nodes and an attribute editor for the selected node, allowing to manually add, change and delete properites.

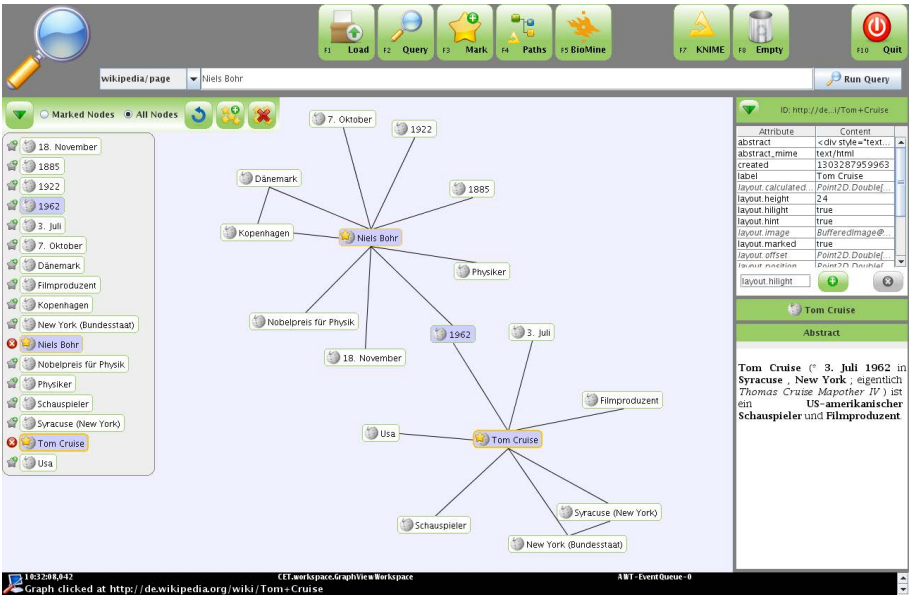

Fig. 1. Screenshot of the Creative Exploration Toolkit (CET), showing an exploration subgraph from the Wikipedia provider

While CET takes care of graph visualization and presentation, domain-specific semantics are not supported. To give an example: The shortest path between two nodes is displayed by highlighting all nodes on the path. However, the user interface is not aware of the path-property, but only displays the highlight attribute of the nodes, while the actual calculation takes place in the underlying data analysis platform described in the next section. The user interface is therefore very flexible when it comes to tasks from different domains.

4 Network and Algorithm Providers

While the CET provides interaction and visualization, it relies on external tools to provide graphs and algorithms. We are working on a selection of providers, including the KNIME platform and Wikipedia—both presented here—as well as a provider for the MusicBrainz[6] network and Personal Information Management (PIM) data.

4.1 Communication between CET and Other Tools

When an interactive process is spread over several nodes, i.e. databases and computation services, it is necessary to keep track of the current status and to be able to propagate changes in the request or outcome very quickly. The Extensible Messaging and Presence Protocol (XMPP)[7] has originally been developed for the Jabber instant messenger, i.e. for the fast and cheap exchange of small messages. From this application, an XML-based real-time messaging protocol has emerged, which now offers numerous extensions (XEPs) for several tasks, including the exchange of large data portions and Remote Method Invocation (RMI) [17].

We definied a unified text message format to allow communication between the tools. This format is also human-readable, which allows for easy debugging and tracing of any communication as well as sending man-made messages during development. A library encapsulates message creation/parsing as well as process management. As the XMPP is an asynchronous protocol, a respective software design is needed. In contrast to a web application one cannot send a request and wait for a response, but has to define a communication context—here as an XMPP process—which groups messages between two or more clients. As an advantage the messages are cheap enough to handle progress messages on a very fine level, allowing to use UI elements such as progress bars even on remotely executed calculations.

4.2 The KNIME Information Mining Platform

KNIME [2], the Konstanz Information Miner, was initially developed by the Chair for Bioinformatics and Information Mining at the University of Konstanz,

[6] `http://www.musicbrainz.org`
[7] `http://www.xmpp.org`

Germany. KNIME is released under an open source license (GPL v3[8]) and can be downloaded free of charge[9]. KNIME is a modular data exploration platform that enables the user to visually create data flows (often referred to as pipelines), selectively execute some or all analysis steps, and later investigate the results through interactive views on data and models. The KNIME base version already incorporates hundreds of processing nodes for data I/O, preprocessing and cleansing, modeling, analysis and data mining as well as various interactive views, such as scatter plots, parallel coordinates and others. It integrates all analysis modules of the well known Weka data mining environment and additional plugins allow, among others, R-scripts[10] to be run, offering access to a vast library of statistical routines.

KNIME has been extended to allow for the flexible processing of large networks via a network plugin, which is available on the KNIME Labs homepage[11]. The network plugin provides new data types and nodes to process (un)weighted and (un)directed multigraphs as well as hypergraphs within KNIME. It further supports the handling of vertex, edge and graph features. Networks can either be processed in memory or in a relational database which enables large networks to be handled within KNIME.

The plugin provides nodes to create, read and write networks. It also provides nodes for filtering and analyzing networks. The created networks can be visualized directly in KNIME or in other existing network visualization tools e.g. visone[12]. Nodes to convert a network into other data types such as matrices and various data tables allow already existing nodes to be used within KNIME. Due to this seamless integration, KNIME can be applied to model complex network processing and analysis tasks.

CET offers a very generic access to KNIME, enabling the user to make arbitrary calls without adapting the user interface. CET can be configured to directly call a KNIME workflow via a configuration and execution dialog. which provides a list of all available workflows and parameters for a selected workflow, which can be edited by the user. Essentially, all information that would be sent by the user interface can be provided to start a KNIME workflow. The result is then visualized in the graph. New analysis methods can therefore be integrated easily into CET by simply adding a new workflow providing the corresponding functionality.

Figure 2 shows an example of a workflow computing the network diameter. In this workflow, first all nodes with a certain feature value are filtered to take only those into account that have been selected and marked by the user. Second, degree filters are applied on nodes and edges to filter unconnected nodes. The shortest paths of all node pairs are subsequently computed and a feature is

[8] http://www.gnu.org/licenses/gpl.html
[9] http://www.knime.org
[10] http://www.r-project.org
[11] http://tech.knime.org/knime-labs
[12] http://visone.info

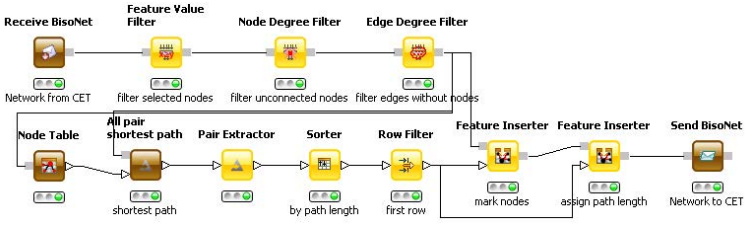

Fig. 2. An example KNIME workflow for calculating the network diameter which is called from CET

assigned consisting of the path length to those nodes of the longest of shortest paths. Finally the graph is sent back to the CET.

4.3 Wikipedia

The Wikipedia client provides query methods needed to explore the Wikipedia structure graph. To obtain the structure, a Wikipedia database dump[13] has been crawled and its structure stored into a MySQL database containing

- the pages—which constitute the nodes in the graph structure—with the page URL, their title and a flag stating whether a page is a redirect to another page. For pages with content, the text part before the first heading is stored as an abstract. However, this only holds the source code from the Wiki page, not any rendered contents, i.e. to display the abstract as it can be seen from Wikipedia, a MediaWiki conversion must be applied.
- links between the pages, including the source page, the destination and an alternative text given for a link, as the MediaWiki markup allows to state any text to be displayed for the link content.

An XMPP service has been set up to provide access to the Wikipedia structure. There are commands to query pages by their name, get the list of all pages connected to a specific page (expansion) and find all links connecting a list of pages (completion). The completion step is needed to fill a graph after nodes have been added, as a simple expansion only views outgoing links with respect to a node. To find incoming links, i.e. links which are outgoing from another node, all pages in the displayed subgraph have to be revisited. Additionally, the abstract of a page can be acquired to be displayed in the side-bar.

5 Evaluation

In this section we describe the evaluation of the CET in the form of a small user study. With this study we prove the applicability of the CET for graph exploration tasks and show how basic functionality of KNIME can support the user by these activities. In this case study we concentrate on one possible scenario

[13] `http://en.wikipedia.org/wiki/Wikipedia:Database_download`

for graph exploration which is knowledge discovery, ex. *bisociations* discovery or discovery of new unexpected relations in the graph data (see [1]).

The case study is carried out in the form of a controlled comparative lab experiment. Our research question is, whether graph based navigation outperforms hypertext navigation in terms of effectiveness, efficiency and user satisfaction. Thus, the target of this study is to compare knowledge discovery, using the CET, which is based on graph navigation with hypertext navigation when users are searching online using web resources e.g. Wikipedia. For a general discussion about the evaluation of exploratory user interfaces see [9].

Hypothesis: Users can make more novel discoveries or make them faster when using our graph-based interface in comparison to exploration based on hypertext navigation.

5.1 Study Design

The study consisted of a lab experiment combined with a questionnaire. By the questionnaire we collected the participants' demographic data, their computer skills and search experience, results of the search experiment using the user interface and Wikipedia, and usability assessment. For the experiment we used the German version of Wikipedia.

We designed several search tasks to reflect knowledge discovery. Especially we concentrated on bisociations discovery. There are three types of bisociations: bridging concepts, bridging graphs and structural similarity (see [1]). We concentrated on bisociations of type "bridging graphs", which can be found in Wikipedia. These bisociations contain named entities that have many "connecting domains" in common. As example, Indira Gandhi and Margaret Thatcher are concepts from the same domain (person) connected through such domains as: university (both attended Oxford), career (Female heads of government), political position (Cold War leaders), sex and century (Women in 20th century warfare). Here, the "connecting domains" are Wikipedia categories. However, the direct link between Indira Gandhi and Margaret Thatcher was missing in Wikipedia.

To be able to find such bisociations the user interface should support searching for similarities between concepts. Therefore we considered the following independent searching tasks for our study:

- Participants should find what the two concepts have in common.
- Participants should build the association chain between two concepts[14].

In the lab experiment the participants used the CET to solve two tasks of the types described above. They also did two similar tasks using online web search on Wikipedia. We employed a Latin Square blocking design [12] in our lab study to get rid of the effects like the order the users use the discovery tools which can bias the results. That means one group of our participants started with the first task set and used the CET and after that used online Wikipedia to solve

[14] To avoid confusion we omitted the term "bisociation" in the study and used "association" instead.

the second task set. Second half of the participants started with the first task set and used online Wikipedia interface and after that used the CET to solve the second task set. Our toolkit supports the user discovery process with the following features:

- *Querying the graph:* exact match and its neighbour nodes can be found.
- *Graph visualisation:* with Wikipedia articles as nodes and links between them if there is a hypertext link in the abstract of one of the articles.
- *Explanation of relations between graph nodes:* for each node a corresponding article abstract can be seen.
- *Graph navigation:* each node can be expanded by its neighbours.
- *Shortest path:* indication of the minimum intermediate concepts/nodes between two or more concepts/nodes in the graph.

Before conducting the lab experiment, the participants were given instructions on how to operate the CET. They were also given some time to try out the toolkit. When using online Wikipedia the participants used the *Firefox*[15] browser and were allowed to perform Wikipedia search as they usually do, e.g. open several tabs and use the internal search feature. The participants were allowed to use only the text in the abstract of an article. They could also follow the hypertext links found only within an abstract. This limitation arose from the limited experiment time (otherwise the users could spend much time reading and understanding information) and from the construction of our graph-based interface. Each participant was told that the target answer, he or she was supposed to find in the lab experiment, should be derived from article abstracts.

5.2 Results of the Study

Twelve users participated in our study: 66.7% (8) men and 33.3% (4) women. Their average age was about 26. The majority of participants had informatics-related profession like engineering, IT-research assistant or student. As we employed a Latin Square blocking design there were 50% of the users (6) in the first group and 50 % (6) in the second with equal percentage of women in each group. All participants categorized themselves as professional users of computer programs. Almost all participants (91.7%) used search engines (e.g. Google) every day to make their investigations. One participant uses search engine several times a week. The majority used Wikipedia to make their investigations several times a week.

The experiment consisted of two similar sets, each with two unrelated tasks (see Table 1). The participants were equally successful solving the first task set independent of the tool they used: based on hypertext navigation or graph based navigation. The second task set was more complicated than the first one. Especially the task about the association chain between amino acids and Gerardus Johannes Mulder showed that graph-based tool better supports users by knowledge discovery. All the participants managed the task using the CET while only

[15] `http://www.mozilla.com/firefox`

one third did it based on Wikipedia. One third of the participants who were supposed to solve the task based on Wikipedia even gave up and presented no solution. The participants also spent less time on task solution if using the CET in comparison to hypertext navigation with exception on one task (Table 2). One important note is that the participants mainly did not know the answers in advance[16]. The proportion of people who knew the answer before the search experiment was equal comparing two configuration groups. This information is important because it would not make sense to compare the programs if the users already knew the answers as then they could find the right answer not because they used one of the tools.

Table 1. Task solving statistic. Success rate in %.

		Wikipedia			CET		
Task set	Task description	Right answer	Wrong answer	Not solved	Right answer	Wrong answer	Not solved
1	What do Tom Cruise and Niels Bohr have in common?	83.3	16.7	0	83.3	16.7	0
	Build an association chain between computer science and Netherlands	100	0	0	100	0	0
2	What do Jean-Marie Lehn and Thomas Mann have in common?	83.3	16.7	0	100	0	0
	Build an association chain between amino acids and Gerardus Johannes Mulder	33.3	33.3	33.3	100	0	0

Table 2. Task solving statistic. Mean time spent on solving (in minutes).

Task set	Task description	Wikipedia	CET
1	What do Tom Cruise and Niels Bohr have in common?	4.00	1.33
	Build an association chain between computer science and Netherlands	1.50	1.67
2	What do Jean-Marie Lehn and Thomas Mann have in common?	2.33	1.50
	Build an association chain between amino acids and Gerardus Johannes Mulder	3.83	2.67

[16] For the task about association chain between computer science and Netherlands, two participants, who used Wikipedia, and two participants, who used CET, knew the answer before the search. This information was learned from the questionnaire.

To summarize, our hypothesis that users can make more new discoveries or achieve them faster using our graph-based interface in comparison to online web search based on hypertext navigation was supported by the study[17]. We also observed on user actions during the experiment. Participants experienced difficulties analyzing even small portions of text without the support of CET (see the example of *Tom Cruise* and *Niels Bohr* in Figure 3).

CET, which has a graph-based interface, helps users to see the connections between concepts at once (see the example of *Tom Cruise* and *Niels Bohr* in Figure 1). That is why our tool is better for knowledge discovery.

We analyzed the participants' opinion on our tool to improve it. The overall rate of the program support of the functions for information discovery was good (see Table 3). The best mean assessment (nearly very good) was for finding relations between topics. The study results show that the program does not sufficiently support the search for topics and we should work in the direction to better support this functionality.

Furthermore we evaluated the usability of the user interface. This statistic is summarized in Table 4. The overall usability assessment was good. The best mean assessment was for user support by solving the searching tasks which was nearly very good. The participants again confirmed our hypothesis that

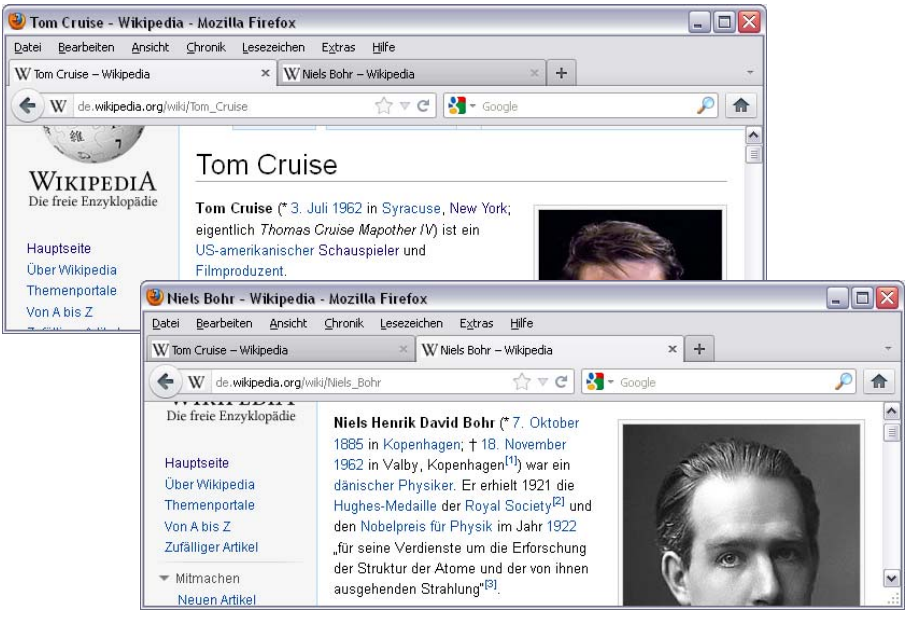

Fig. 3. Screenshots of abstracts of the Wikipedia articles on Tom Cruise and Niels Bohr. Tom Cruise was born in the year Niels Bohr died.

[17] As the participants group was relatively small we do not have a statistical proof. Further studies would be beneficial.

Table 3. User assessment how well the program supports them by knowledge discovery with a scale from 1 (very bad) to 5 (very good)

Functionality	Mean	Min	Max	St. Dev.
Topic search	3.67	2	5	0.78
Navigation between topics	4.25	3	5	0.75
Finding relations between topics	4.67	4	5	0.49
Understanding the relations between topics	4.08	2	5	1.08
Knowledge discovery	4.08	3	5	0.67

Table 4. Usability assessment with a scale from 1 (very bad) to 5 (very good)

Usability criteria	Mean	Min	Max	St. Dev.
Intuitive operation	4.00	3	5	0.74
User support by task solving	4.75	3	5	0.62
User support of knowledge discovery vs. Wikipedia	4.17	3	5	0.84

graph-based interface in comparison to online web search based on hypertext navigation better supports knowledge discovery[18].

6 Conclusion and Future Work

We presented a user interface for generic, exploratory graph visualization with special emphasis on extensibility by integration with data and graph analysis methods provided by KNIME. The presented interface allows for easy interaction with the visualized graphs. This setup is particularly interesting for researchers in the area of Data Mining and Network Analysis, as it is very simple to plug in new approaches and visualize the results, even if there is interaction involved. With a case study we proved the CET applicability for knowledge discovery on tasks requiring structural analysis of data sets.

Future work includes the enhancement of available interaction elements, eventually being able to plug in arbitrary control widgets, improvements on the communication facilities—with extensions of our XMPP library—and the integration of more data sources.

Acknowledgement. The work presented here was supported by the European Commission under the 7th Framework Programme FP7-ICT-2007-C FET-Open, contract no. BISON-211898.

[18] One participant wrote a note, that he did not use Wikipedia for discovery. That is why he could not compare these two tools. But he admitted he knew no alternative to our graph-based tool.

References

1. Berthold, M.R. (ed.): Bisociative Knowledge Discovery. LNCS (LNAI), vol. 7250. Springer, Heidelberg (2012)
2. Berthold, M.R., Cebron, N., Dill, F., Gabriel, T.R., Kötter, T., Meinl, T., Ohl, P., Sieb, C., Thiel, K., Wiswedel, B.: KNIME: The Konstanz Information Miner. In: Data Analysis, Machine Learning and Applications - Proceedings of the 31st Annual Conference of the Gesellschaft für Klassifikation e.V. Studies in Classification, Data Analysis, and Knowledge Organization, pp. 319–326. Springer, Heidelberg (2007)
3. Cook, D.J., Holder, J.B. (eds.): Mining Graph Data. John Wiley & Sons, Inc., Hoboken (2007)
4. Battista, G.D., Eades, P., Tamassia, R., Tollis, I.G.: Algorithms for drawing graphs: An annotated bibliography. Computational Geometry: Theory and Applications 4(5), 235 (1994)
5. Didimo, W., Liotta, G.: Mining Graph Data (Graph Visualization and Data Mining), ch.3, pp. 35–63. John Wiley & Sons, Inc., Hoboken (2007)
6. Dweyer, T., Rolletschek, H., Schreiber, F.: Representing experimental biological data in metabolic networks. In: 2nd Asia-Pacific Bioinformatics Conference (APBC 2004). CRPIT, vol. 29, pp. 13–20. ACS, Sydney (2004)
7. Eades, P.: A heuristic for graph drawing. Congr. Numer. 42, 149–160 (1984)
8. Gansner, E.R., Hu, Y.: Efficient, proximity-preserving node overlap removal. J. Graph Algorithms Appl. 14(1), 53–74 (2010)
9. Gossen, T., Nitsche, M., Haun, S., Nürnberger, A.: Data Exploration for Knowledge Discovery: A brief Overview of Tools and Evaluation Methods. In: Berthold, M.R. (ed.) Bisociative Knowledge Discovery. LNCS (LNAI), vol. 7250, pp. 287–300. Springer, Heidelberg (2012)
10. Haller, H., Kugel, F., Völkel, M.: iMapping Wikis - Towards a Graphical Environment for Semantic Knowledge Management. In: SemWiki (2006)
11. Haun, S., Nürnberger, A.: Supporting exploratory search by user-centered interactive data mining. In: SIGIR Workshop Information Retrieval for E-Discovery (SIRE) (2011)
12. Hearst, M.: Search user interfaces. Cambridge University Press (2009)
13. Herman, I., Melançon, G., de Ruiter, M.M., Delest, M.: Latour – A Tree Visualisation System. In: Kratochvíl, J. (ed.) GD 1999. LNCS, vol. 1731, pp. 392–399. Springer, Heidelberg (1999)
14. Herman, I., Melancon, G., Marshall, M.S.: Graph visualization and navigation in information visualization: A survey. IEEE Transactions on Visualization and Computer Graphics 6(1), 24–43 (2000)
15. Koren, Y., Çivril, A.: The Binary Stress Model for Graph Drawing. In: Tollis, I.G., Patrignani, M. (eds.) GD 2008. LNCS, vol. 5417, pp. 193–205. Springer, Heidelberg (2009)
16. Payne, S.J.: Mental models in human-computer interaction. In: Sears, A., Jacko, J.A. (eds.) The Human-Computer Interaction Handbook, pp. 63–75. Lawrence Erlbaum Associates (2008)
17. Saint-Andre, P.: Streaming XML with Jabber/XMPP. IEEE Internet Computing 9(5), 82–89 (2005)
18. Tamassia, R.: Advances in the theory and practice of graph drawing. Theoretical Computer Science 17, 235–254 (1999)