Supporting Analytical Reasoning and Presentation with Analytic Provenance

Phong H. Nguyen*
Middlesex University

1 Introduction

Analytic provenance research tries to understand a user's reasoning process by examining their interactions with a visual analytic system. Visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces [30]. The key role of visual analytics is to support analysts to derive insight from massive amounts of data and to make decision based on the derived knowledge. However, not only is the extracted knowledge important, but the analysis process that led to that knowledge and the rationale underlying the analysis are also of great significance [23, 14].

In 1996, Shneiderman already noticed the importance of studying user interactions in information visualisation by classifying *history* as one of the seven tasks in his *Task by Data Type Taxonomy* [26]. According to Shneiderman, information visualisation systems need to support users to review previous actions and correct mistakes because the information exploration process is typically long and complex. Since then, there has been more research on exploration history and analytic provenance in visualisation and related fields. In May 2011, the first workshop dedicated to analytic provenance was held in *CHI 2011* conference to develop a research agenda to better study analytic provenance and a call to action for further research. In that workshop, the following definition of analytic provenance was proposed, "the area of research that focuses on understanding a user's reasoning process through the study of their interactions with a visualisation is called Analytic Provenance" [23, p.33]. Besides understanding the user's reasoning process, many benefits can also be gained from analytic provenance such as recalling the analysis process, reusing performed analyses, supporting evidence in constructing the reasoning process, and facilitating collaboration between colleagues including dissemination, discussion and presentation (Section 2.3).

The literature of analytic provenance is reviewed on Section 2. Based on that, we develop three research questions. First, how to capture the provenance of the analytical reasoning process including both *high-level reasoning* and its *low-level interactions* performed (Section 3.1)? Second, how to effectively visualise the captured provenance to address the *scalability* and the *complexity* challenges in visualising analytic provenance (Section 3.2)? Third, how to build an analytical product that can address the *auditability* and *context-sensitivity* challenges in presentation (Section 3.3)?

2 Related Work

Typically, an *analytic provenance aware* system consists of three stages: capturing the provenance of the analysis process, visualising the captured information, and utilising the visualised provenance. As a result, we characterise the literature of analytic provenance by these stages.

*e-mail: p.nguyen@mdx.ac.uk

2.1 Capturing Analytic Provenance

The first step in capturing analytic provenance is to decide what kind of information needs to be captured. Does the system capture low-level user interactions, or high-level user intentions, or both of them? The decision may depend upon how the system subsequently uses the captured information.

Based on an empirical study, Gotz and Zhou [12] characterise visual analytic activities at multiple levels of granularity according to the semantic richness of these activities: the top-level *tasks* (high-level analytic goals), the high-level *sub-tasks* (more concrete sub-goals to fulfil the goal), the low-level *actions* (detailed analytic steps to achieve the sub-goal such as filtering or sorting data) and the bottom-level *events* (the actual interactions need to perform such as mouse-clicks or keystrokes). Figure 1 illustrates the model and an example scenario.

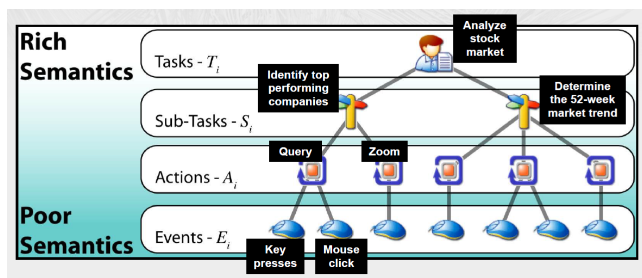


Figure 1: Illustration of the Gotz and Zhou model [12] with an example in business environment. One possible analysis task is to analyse the stock market to invest. Two proposed sub-tasks are identifying the top performing companies and finding the trend of the market this year. To accomplish the first sub-task, the analyst queries top 50 highest profit companies, which requires clicking on the query button and typing '50'. All these task, sub-tasks, actions and events are represented in the model.

Following this characterisation, a system can capture the information corresponding to one or many tiers. We describe the existing work of capturing provenance of each tier below.

2.1.1 Capturing Bottom-Level Events

Glass Box [5] can record a great deal of low-level information including keyboard/mouse events, window events, file open and save events, copy/paste events, and so on. Its objective is to capture, archive and retrieve intelligence analysis activities.

There is not much research on capturing events because it is relatively easy and limitedly useful. Simply capturing these events alone does not provide sufficient information to understand their purpose and rationale. For example, we know that a *mouse click* is captured; however, what the purpose of that click was (e.g., to sort the data?), and why the user performed that click (e.g., to find an interesting pattern from the data?) are unknown. Commonly, when analysing data with a visualisation, an analyst needs to perform many operations with trials and errors to find the answer to the problem. In that case, a series of poor-semantic and bottom-level events makes it more difficult for the analyst to recall what has been done. Therefore, more meaningful interactions also need to be captured.

2.1.2 Capturing Low-Level Actions

Taxonomy of Actions Actions in the action tier are both semantic and generic across different visual analytic systems, thus are commonly used as the semantic building blocks for the provenance of derived insight [13, 27, 12]. Gotz and Zhou [12] provide a taxonomy of actions that contains the most common analytic operations in many visual analytics systems they observed. The taxonomy classifies actions, based on their intention, into three groups: *exploration* actions (e.g., *filter* the data according to a condition), *insight* actions (e.g., *bookmark* the current visualisation), and *meta* actions (e.g., *undo/redo* a performed action).

Considering an example to distinguish between capturing meaningful actions and less-meaningful events. The action *zoom-in* has an intention of increasing the details of the display area, and thus can be used either to reduce the focus area of a map or to refine the display time range of a timeline, in a semantically equivalent manner. Moreover, this action could be accomplished by three different events: scrolling the mouse, pressing the combination of Ctrl and plus (+), and tapping on a smart device. Recording only the event; for instance, scrolling the mouse, is not sufficient to know whether its action is zoom or just a common scroll in a text editor.

Automatically Capturing Low-Level Actions During the course of analysis with a visualisation, all user interactions can be systematically recorded. The visual exploration process can be modelled using *graph* metaphor. Nodes in the graph represent *states* of the application and edges represent *actions* that transform one state into the other state. Considering an example of *bar-chart* visualisation, states are all the necessary information allowing to reconstruct the captured chart such as the *dataset* and the *colour map*; while an action could be *sorting data*. The system can support *undo* to revisit to a previous state; and if a new action is performed at that state, a new branch will be created to store that new line of inquiry.

Basically, there are two prime strategies to automatically capture the exploration process. One is capturing the initial state and all the performed actions so that they can be rerun to achieve the desired state [17]. Second is simply capturing all visualisation states after each action [1]. The former strategy suffers from potential long running time if the number of actions need to executed is high; while the latter is memory-expensive if a state contain too much information. The later is easier to implement; whereas, the former allows re-applying the analysis process with a different dataset.

2.1.3 Capturing High-Level Sub-Tasks or User Intentions

Typically, high-level sub-tasks can be either inferred from captured low-level actions or directly recorded by users.

Deriving from Captured Actions When analysts interact with a visual analytics tool, their plans and methods to analyse data could partially be reflected through their interactions with the application.

Manual Derivation Dou et al. [8] conduct a quantitative study to measure how much of a user's reasoning process can be recovered from only the captured user actions. Reasoning results decoded from the interaction logs are compared with the ground-truth reasoning from analysts' interviews; and the results show that 79 per cent of the findings, 60 per cent of the methods and 60 per cent of the strategies could be extracted from manually analysing the interaction logs. This post-analysis approach is domain-specific because ad hoc tools need to be designed to effectively discover some well-known strategies in a particular domain, detecting suspicious activities in wire transactions. Even though reasoning processes are discovered, the interaction analyses occur after the data analyses and thus cannot support analysts in real-time.

Automatic Derivation Gotz and Zhou use heuristics to automatically infer a sub-task from a series of actions [12]. One heuristic suggests that a user solves a sub-task by completing a combination of several exploration actions followed by an insight action. For example, the analyst explores the data by selecting bar-chart as a visualisation technique (*change-metaphor* action), sorting the data according to some indicator (*sort* action), and then annotating (*annotate* action) on the highest column of the chart. The heuristic considers "annotate", the insight action, as a signal of deriving insight, or solving a sub-task; and represents that sub-task as a trail of three actions "change-metaphor - sort - annotate". However, if the analyst does not annotate or bookmark visualisations, the heuristic cannot derive any sub-tasks.

Automatic derivation provides real-time support for users to quickly understand the analysis process. However, because heuristic approach could lead to a misleading user intention, it should only assist analysts and allow them to correct the derived intention.

Directly Capturing using Annotations Instead of inferring user intentions from low level actions, analysts can manually capture the insight by annotating on the visualisations of interest. Sense.us [15], a web site supporting asynchronous collaboration, allows users to annotate on visualisations, and use these annotated visualisations in discussion. GeoTime [10], a geotemporal event visualisation tool, supports embedding hypertext linked visualisations and visual annotations in an analysis story. Annotation can provide more information than simple text and graphics attachment. *Data-aware* annotation detects the subset of data belonging to the annotated area so that the data of interest remains unchanged when new visualisation metaphors are applied for further investigation [6]. Another benefit of data-aware annotation is that statistical values could be automatically generated to add more information such as the mean and the extreme values of selected data items [3].

Manual annotation provides high-fidelity; however, users often only take notes of the final state of a visualisation [12]. Therefore, intuitive annotation mechanism needs to be designed to encourage users to take notes.

2.1.4 Capturing Top-Level Tasks

Top-level tasks are highly domain-specific; therefore, it's virtually impossible to automatically derive them. The user needs to explicitly write down what the task is before solving the problem. Further enhancement has been made to allow users to document their reasoning processes; for example, recording found interesting patterns about the data, describing their causal relationships, and building a hypothesis based on these found artefacts [27, 24]. This mental model needs to be documented directly onto the same system for effective reasoning rather than keeping tacitly or recording it into an external application such as Microsoft Word (see [27] for explanation).

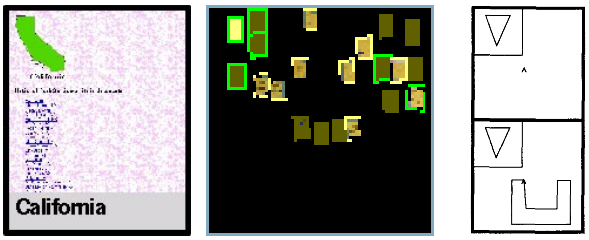
2.2 Visualising the Captured Information

Typically, events are not visualised because they do not carry much information and the number of events is high. Actions and states (the visual results of the actions) are commonly visualised together to depict the *analysis process*. Sub-tasks and tasks are illustrated in the graphical *reasoning process*.

2.2.1 Visualising the Analysis Process

Methods of visualising individual actions and states will be discussed, and followed by methods of chaining them together to visualise the entire analysis process.

Visual Representation of States Conventionally, the space for rendering a state is limited because a small portion of the display area is reserved for provenance information; whereas, the much larger portion is used for data exploration [27, 12, 17]. Therefore, a small-scale visualisation of the captured state is popular [16] (Figure 2(a)). To recall the affect of the performed action, the miniature can highlight the difference from the previous state [19] (Figure 2(b)), or combine both the former and the latter visualisations corresponding to that action [20] (Figure 2(c)).



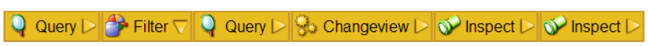
(a) The exact miniature of a web-site[16].

(b) Each rectangle represents a post-it note. Modified post-it notes are highlighted with green colour [19].

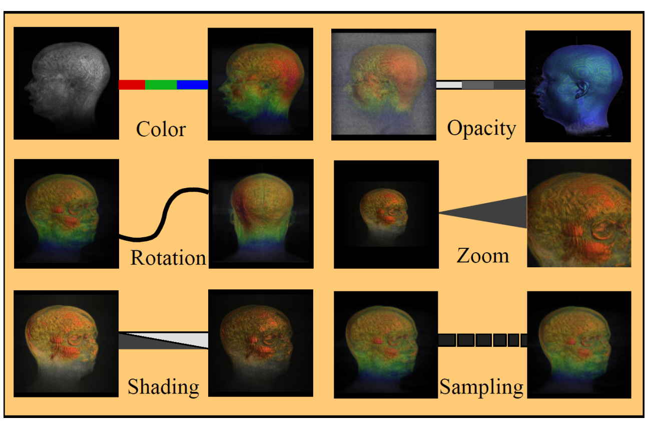
(c) In a graphical editor, combining the state before the action (above) and the state after the action (bottom) [20].

Figure 2: Examples of visual representations of states.

Visual Representation of Actions Typically, a system supports a certain number of actions; and thus allows using icons to visually distinguish different kinds of actions besides texts [12] (Figure 3(a)). Actions are also commonly represented as edges in a graph to connect two states. Therefore, graph edges can be stylised to reflect the characteristics of the represented actions [21] (Figure 3(b)).



(a) Using icons to represent actions including query, filter, change-view and inspect [12].



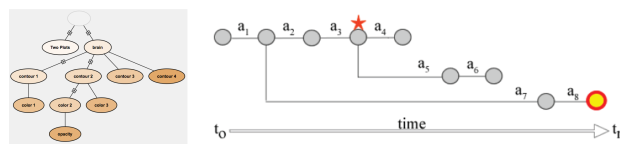
(b) Using stylish edges to represent actions including changing colour map, rotating, shading, changing opacity, zooming and sampling [21].

Figure 3: Examples of using visual representations of different types of actions.

Layout of Actions and States Typically, the system begins with an initial state (node). When the user performs an action, a new node is created for the current state, and a new edge is added to connect the previous node with the current node. Gradually, a string of nodes and edges is built in the chronological order. The system can support revisiting to the previous state. If a new action is performed at that state, a new branch will be forked to store forthcoming actions. Therefore, the analysis process has the layout of a *direct acyclic graph*, or a *tree* if revisited links are not explicitly visualised.

To not distract analysts from the primary data exploration and save space, several techniques have been proposed to reduce the display area of the provenance graph: organising trees in the right horizontal-vertical layout [27], displaying only nodes of the active branch that led to the selected visualisation [19], allowing graph nodes be expandable/collapsible on demand [1], supporting zoomable and pan-able interface [9], and applying distortion techniques to focus on more relevant states [22].

The order of actions can be interpreted through the direction of edges in the provenance graph. Moreover, exact time gap between actions can also be measured and visually encoded into the visualisation. VisTrails [1] colour-codes the background of visualisation nodes according to when they are created (Figure 4(a)); and Aruvi [27] uses the length of edges to represent the distance in terms of time between two states (Figure 4(b)). This time indication can be updated only when a new node is added, or continuously to reflect the fact that time is always flying. In the latter case, endlessly, background nodes will become lighter and edge lengths will become shorter. This *time-travel* interface is implemented in Visage [7].

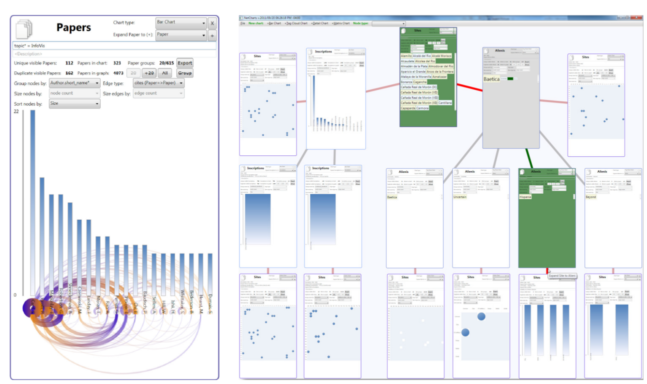


(a) The darker node refers to the later visualisations [1].

(b) The edge length represents time distance between two connecting nodes [27].

Figure 4: Examples of encoding time into provenance visualisation.

The provenance space can be unified with the data exploration space, where each node of the provenance graph represents a fully interactive visualisation. Zoom-able and pan-able interface needs to be supported to allow users either to focus on the current visualisation (Figure 5(a)) or to observe the entire context of the analysis process (Figure 5(b)).



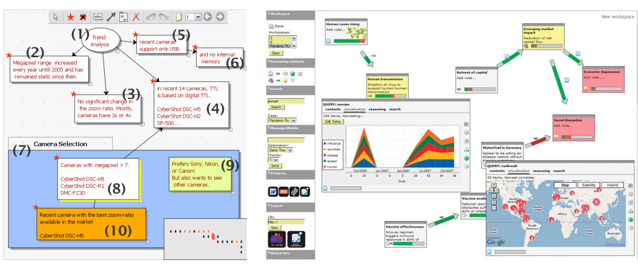
(a) Focus view on the active visualisation.

(b) Context view to see the overview of what has been done.

Figure 5: An example of integration of the provenance space into the data exploration space [9].

2.2.2 Visualising the Reasoning Process

The Aruvi system [27] allows analysts to freely compose the reasoning process by using a graphical editor. Users can take note in rectangles or ellipses, and use arrows to connect them. Conventionally, nodes can be referred as *evidence*, *assumptions* or *hypotheses*, and arrows can be referred as *causal relationships*. Nodes in the editor can be linked to the captured visualisations to help explain its reasoning, and these nodes are marked with a star to indicate the existence of the linked visualisations. Instead of using a graphical editor and an implicit convention to map drawing shapes with reasoning artefacts, Scalable Reasoning System [24] provides a more formal method to document the reasoning process. A captured visualisation can be dropped to the reasoning space to create a node. The node shows the miniature of the captured visualisation and can be tagged as an *evidence* artefact. An evidence can be converted to a *casual relationship* and its rectangular shape will become an edge. An *assumption* is a free note and can be upgraded as a *hypothesis* when it is supported by an evidence. Figure 6 shows those two examples of reasoning process visualisation.



(a) Using graphical editor to freely construct the mental model [27].

(b) A formal reasoning diagram with different types of artefacts: evidence, casual relationship, assumption and hypothesis [24].

Figure 6: Examples of visualising the reasoning process.

2.3 Utilising the Visualised Provenance

The visualised provenance can be exploited to support the analytical reasoning process and collaboration.

2.3.1 Supporting the Analytical Reasoning Process

Recalling the Analysis Process Provenance visualisation provides a visual overview of the analysis process. Therefore, it helps the analyst recall what has been done, and potentially reminds what is missing and what should be done. Provenance visualisation should not be a static picture of the past. In contrast, it should allow the analyst to freely navigate back to the desired state [7, 19, 1]. A state can be easily selected through the interface or with the help of *search and filter* mechanism when there are too many recorded states. During the analysis process, users can take notes and tag keywords on visualisations; and all these metadata are subjects to search [28]. Moreover, VisTrails supports *query-by-example* to find related visualisations [25]. Past states can be filtered to display periodically [19] or based on a particular metadata such as authors [19] and time [28].

Reusing the Performed Analyses When reviewing the provenance, analysts can insert missing actions, remove undesired actions, and reapply past actions to a new dataset [7]. The past actions can also be modified directly by changing the command statements [17], the command parameters [11], and the changing effects can be propagated along the history trail [21].

Typically, in scientific visualisation, a visualisation is considered as a rendering result of many understandable parameters. Therefore, it is feasible to compare visualisations by measuring these sets of parameter values. VisTrails, [1], a scientific visualisation workflow system, allows comparing and merging two visualisations into a new one. More specifically, *set* operations including *intersection*, *union* and *difference* can be employed to build the parameter set of the generated visualisation. In GraphTrail [9], a information visualisation tool, it is still possible to merge visualisations in terms of data. The active dataset of each visualisation is mapped with a *SQL statement*; thus, performing a SQL *union* statement will result a new visualisation with the combined data of interest.

Supporting Evidence in Constructing the Reasoning Process As discussed in Section 2.2.2, the reasoning process can be graphically documented inside the system. By capturing analytic provenance, we can attach the recorded visualisation to the reasoning evidence to support that artefact [27]. Not only the visualisation but also could all the steps that the analyst performed to generate that visualisation be helpful.

2.3.2 Supporting Collaboration

Dissemination and Discussion Visualisations can be annotated, captured and attached into the discussion forum to help peers understand the findings of author easier [15]. Captured provenance can also be embedded into a formal analysis story to visually convey idea [10]. The embedded provenance should be interactive so that audience can examine and verify what the author wrote [24]. In asynchronous collaboration, each individual colleague can capture insight and submit them to a central repository. As a result, all peers can exploit other findings and facilitate the solving problem process [2].

Presentation Analytic provenance can also be exported for presentation purpose with various published formats. Outpost [19], a tangible interface for collaborative web site design, provides a *print version* of annotated visualisations as a report. VisTrails [1], a scientific workflow and provenance management system, supports embedding the visualisation process into a paper through *Latex* format. Image Graphs [21], a volume visualisation system, builds an *animation* from selected key visualisations.

3 Research Questions

3.1 Research Question 1

How to capture the provenance of the analytical reasoning process including both high-level reasoning and its low-level interactions performed?

Low-level interactions can be automatically captured when an analyst interacts with a visualisation; however, only low-level actions are not sufficient to understand the purpose and the rationale behind those actions [12]. Therefore, high-level reasoning also needs to be captured together with its low-level interactions.

The challenge in capturing high-level user intentions is that the interpretation of user actions is domain specific. The complexity even arises because the analyst commonly keeps their thought inside during the analysis process. Currently, there are two approaches to capture high-level user intentions. The first approach is to automatically record low-level interactions (should be more meaningful than mouse and keystroke events) and use some heuristics to automatically infer the intention from the captured interactions [12]. The second approach is to let the analyst record their reasoning by taking notes, annotating on visualisations [15] or freely constructing the reasoning process [24].

We propose a hybrid approach that combines the benefits of both automatic and manual method and reduces their deficiencies.

Manual-based Commonly, the analyst annotates on the visualisation in the data analysis view (*exploration space*), and manually document their thinking in another view (*reasoning space*). We plan to use both annotation and reasoning process construction techniques to capture user intentions. However, the reasoning space will be unified with the exploration space to reduce the cognitive load when switching the context between two spaces. Novel annotation mechanism will be designed to encourage analysts to take notes and annotate on visualisations. Novel techniques to facilitate the reasoning documenting will also be designed; for example, simple drag-and-drop a note in the exploration space to the reasoning space to create an evidence for reasoning.

Automatic-based We plan to automatically capture meaningful actions described in the Action tier of the four-level-semantic model of Gotz and Zhou [12]. We use their taxonomy as the building blocks of the analytic provenance and may extend it during the development and evaluation of the project. The large number of performed analytic actions makes it difficult to follow the analyst's logic; therefore, we propose to use a heuristic to automatically group continuous and closely related actions into one higher level intention. For example, if the analyst selects the dataset about population of all cities in England and then selects bar chart as the visualisation method. Finally, the analyst brushes the highest column in the bar chart, which is London, and derives insight from that visualisation. For this three-step process, the analyst wants to find the city that has the highest population in England and this intention could be automatically derived from that series of actions. The challenge of this method is how to detect when the analyst begins and ends a series of actions that serve for one purpose, and how to interpret the analyst's intention from a given sequence of actions.

Outcome We plan to have the following research results:

O1. A seamless integration of the exploration space and the reasoning space.

O2. Novel techniques to facilitate the reasoning process construction.

O3. A novel annotation mechanism to encourage analysts to take notes and annotate on visualisations.

O4. A heuristic to automatically infer user intentions from continuous and closely related interactions.

Evaluation All the research outputs will be empirically evaluated through a controlled lab experiment followed by a semi-structured interview. Participants will be recruited to perform some analysis tasks by using our system that will implement all research solutions. The controlled lab experiment will measure the performance improvement to assess the benefit of all combined research results. The interview will help to gain qualitative feedbacks and specifically evaluate the first three research outputs. Some of the questions could be as follows. *O1*: do users find the unified exploration-reasoning space useful or confused? *O2*: does the annotation mechanism make users comfortable take notes and annotate on visualisations? *O3*: do the new techniques really leverage users to document their reasoning? Regarding to *O4*, every time the system automatically generates user intentions, it will log the original actions and ask the user to compare at the end of the analysis to quantitatively measure the accuracy of the heuristic.

3.2 Research Question 2

How to effectively visualise the captured provenance to address the scalability and the complexity challenges in visualising analytic provenance?

The challenge of visualising analytic provenance is that the analysis process may contain a huge number of interactions; whereas, the space used for visualising them is limited. Typically, an analytic provenance supported system preserves a small space for provenance visualisation, while most of the space is used for data exploration [7, 13, 12, 27]. *Graph metaphor* is a natural method to visualise analytic provenance. Nodes represent (small-scale) visualisations and edges represent actions that transform one node to another. When the analyst revisits an old visualisation and performs a new action, a new branch will be forked to store the new line of inquiry. To address the scalability and the complexity issues, several techniques have been proposed including horizontal-vertical layout [27], collapsible graphs [1], and zoom-able and pan-able interface [9].

Different from those *graphical* methods, we propose a *semantic* approach to reduce the number of nodes to be displayed. As discussed in the first research question about provenance capturing, low-level actions and high-level intentions could be (semi-)automatically recorded. This hierarchy suggests a *multi-semantic-level* provenance visualisation approach with each level corresponds to each tier in the model proposed by Gotz and Zhou [12]. The top level contains the task that the analyst documented in the reasoning space. The second level stores user intentions, which are also systematically inferred by using some heuristics or manually recorded by users. The third level contains the meaningful actions, which are automatically captured. The last level stores the detailed interactions that the user performed to execute those actions. A collapsible tree, which allows grouping sibling nodes, will be employed to support switching between different levels of semantics and providing details on demand. In addition, analysts can manually edit the high levels of the tree if they do not satisfy with the automatic derivation content, or even modify the structure of the tree.

Moreover, we think that simply visualising actions following the exact order of capturing may not be sufficient to help understand the analyst's logic. It may be useful to reorganise the actions to reflect the user intention. For example, the analyst searches for one keyword, and then searches for another keyword, and finally merges the two result sets. The visualisation that shows the two keyword searches as two parallel actions and the merge action as the combination of these two parallel branches reflects the analyst's logic more precisely than the visualisation that shows three actions in a row. The challenge of this method is detecting the dependency of performed actions.

Outcome We plan to have the following research results:

O5. A multi-semantic-level visualisation of analytic provenance.

O6. A heuristic to automatically reorganise user interactions to correctly reflect the user's logic.

Evaluation These two research outputs will be evaluated similarly as the first research question. For *O5*, the interview will ask participants whether they find the multi-semantic-level graph easy to understand what has happened during their analyses. For *O6*, every time the system automatically reorganises user interactions, it will log the original ones and ask the user to compare at the end of the analysis to quantitatively measure the accuracy of the heuristic.

3.3 Research Question 3

How to build an analytical product that can address the auditability and context-sensitivity challenges in presentation?

Deriving insight, verifying a hypothesis, and answering a question are still not the final steps of visual analysis. Analysts need to create a product that summarises the results, to present it efficiently to a variety of audience, and to disseminate that product to the people who need it [30]. Moreover, according to Chinchor and Pike [4], an analytical product should be *auditable* and *context-sensitive*. For example, for public audience, the product should be fairly simple as a summary of the analysis result. For peer colleagues, the product should include more arguments about how the result was derived. More seriously, for defending an issue, analysts need to present all the reasoning, evidence, analysis that produced these artefacts, and the quality, credibility of data used in analysis.

To the best of my knowledge, the context-sensitivity is not available in any existing work. There has been attempt to make the analytical product partially auditable such as in GeoTime [10]. Analysts can annotate on visualisations and embed them into a story. Therefore, audience can refer to the annotated visualisations to understand the narrative better. However, audience cannot go beyond that understanding such as challenging the embedded evidence, discovering how the analyst derives it, and checking the credibility of the data source.

The analytical product needs to cover its entire analysis process, which is modelled by Keim et al. [18] (illustrated in Figure 7). The process begins with collecting data from multiple sources, which may contain uncertainty and bias (*Data Collection* box in the figure). The data is either directly used to visualise (*Visualisation/Interaction* box) or applied in data mappings, transformations or mining techniques (*Transformation/Mapping* box) to find the data model/pattern. Insight could be gained either from interacting with the visualisation or from the found data model/pattern. The analyst then may build multiple hypotheses about the issue and confirm or object to these hypotheses based on derived insight. This reasoning process maps to the *Insight Synthesiser* box in the figure. Finally, the output of the process is a conclusion, or an answer to the problem.

We propose a *multi-level provenance* approach to address the auditability and context-sensitivity challenges. An analytical product should contain multiple levels of details, to be context-sensitive, and each level should map to the provenance of each stage of the visual analytics process, to be auditable. Specifically, the highest level of an analytical product contains only the conclusion of the analysis, which is derived from the Insight Synthesiser process. The second level of detail adds the provenance of that process to show how and why the conclusion was derived. This provenance includes information of which hypotheses used to derive the conclusion, which arguments used to build the hypotheses, and which evidence used to support the arguments; and is represented as the red cloud *Provenance of Conclusion* in the figure. The third level of detail adds the provenance of each evidence used in the arguments to show how and why these evidence/insight were obtained. This provenance includes model parameters if the insight came from the model and includes important analytical interactions with the visualisation if the insight came from the visualisation. This provenance is represented as the two green clouds *Provenance of Model* and *Provenance of Visualisation* in the figure. The fourth level of detail adds the data provenance of used dataset to show the source, quality and uncertainty of data. This provenance is represented as the purple cloud *Provenance of Data* in the figure.

There has been research on all these different types of provenance: data provenance [29], workflow provenance [1] for provenance of the data model, insight provenance [12] for provenance of the visualisation, reasoning process construction [24] for provenance of conclusion. However, these research works have been done separately and there has not any attempt to combine them. We propose that all four different types of provenance: reasoning provenance, insight provenance, workflow provenance, and data provenance should be linked together to enhance supporting analysts performing analytical tasks because their corresponding processes: Insight Synthesiser, Transformation/Mapping, Visualisation/Interaction and Data Collection have close relationships and are parts of a unique, complete visual analytic process. The challenge is how to communicate and visualise different types of provenance to make them as a unique product.

Outcome We plan to have the following research results:

O7. A multi-provenance-level presentation model that can address auditability and context-sensitive challenges in presentation.

O8. A method to combine, communicate and visualise different types of provenance in the model.

Evaluation These two research outputs will be evaluated similarly as the first research question. For *O7*, the interview will ask participants about the capability we offer to build the product, the detailed information in the product, and the traceability of that information. For *O8*, the interview will ask participants questions related to the visualisation of the combined provenance.

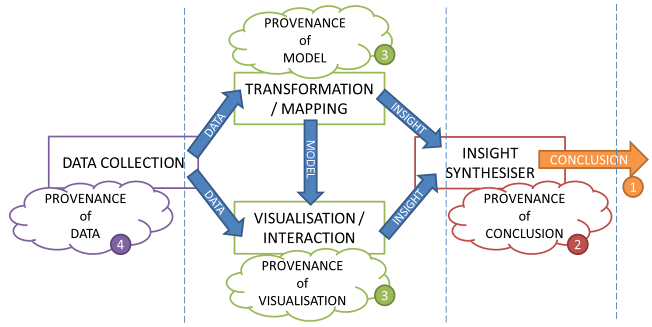


Figure 7: The Visual Analytic Process is enriched by provenance information in all sub-processes to illustrate the idea of combing four different types of provenance to build a multi-level analytical product. Level 1: analysis conclusion. Level 2: provenance of conclusion. Level 3: provenance of insight (mode/visualisation). Level 4: provenance of data.

4 PLANNING

I plan to divide my PhD project into three phases, each corresponds to one research question. Generally, each phase involves algorithm design, prototype development, user evaluation and paper writing. The publications will aim at top Visual Analytics conferences (such as IEEE VAST [1]), HCI conferences (such as ACM SIGCHI [2]), and provenance-specific venues (such as International Provenance and Annotation Workshop [3]). Prototypes will be implemented based on our group's research framework using Java and Processing. The detailed plan is as follows.

Task	Time
Find research questions and work on literature review	12/2011 - 11/2012 (12 months)
Registration	*11/2012*
RQ1: How to capture the provenance of the analytical reasoning process including both high-level reasoning and its low-level interactions performed?	12/2012 - 05/2013 (6 months)
RQ2: How to effectively visualise the captured provenance to address the scalability and the complexity challenges in visualising analytic provenance?	06/2013 - 11/2013 (6 months)
Transfer	*11/2013*
RQ3: How to build an analytical product that can address the auditability and context-sensitivity challenges in presentation?	12/2013 - 05/2014 (6 months)
Write-up	06/2014 - 11/2014 (6 months)

[1]http://www.visweek.org/
[2]http://www.sigchi.org/conferences
[3]http://www.ipaw.info/

REFERENCES

[1] L. Bavoil, S. Callahan, P. Crossno, J. Freire, C. Scheidegger, C. Silva, and H. Vo. VisTrails: Enabling Interactive Multiple-View Visualizations. In *IEEE Symposium on Information Visualization*, pages 135–142. IEEE, 2005.

[2] Y. Chen, J. Alsakran, S. Barlowe, J. Yang, and Y. Zhao. Supporting effective common ground construction in Asynchronous Collaborative Visual Analytics. In *IEEE Symposium on Visual Analytics Science And Technology*, pages 101–110. IEEE, Oct. 2011.

[3] Y. Chen, S. Barlowe, and J. Yang. Click2Annotate: Automated Insight Externalization with rich semantics. In *IEEE Symposium on Visual Analytics Science And Technology*, pages 155–162. IEEE, Oct. 2010.

[4] N. Chinchor and W. a. Pike. The science of analytic reporting. *Information Visualization*, 8(4):286–293, 2009.

[5] P. Cowley, J. Haack, R. Littlefield, and E. Hampson. Glass box: capturing, archiving, and retrieving workstation activities. In *ACM Workshop on Continuous Archival and Retrival of Personal Experence*, pages 13–18, New York, New York, USA, Oct. 2006. ACM Press.

[6] I. Denisovich. Software Support for Annotation of Visualized Data Using Hand-Drawn Marks. In *Conference on Information Visualisation*, pages 807–813. IEEE, July 2005.

[7] M. Derthick and S. F. Roth. Enhancing data exploration with a branching history of user operations. *Knowledge-Based Systems*, 14(1-2):65–74, 2001.

[8] W. Dou, D. H. Jeong, F. Stukes, W. Ribarsky, H. R. Lipford, and R. Chang. Recovering Reasoning Processes from User Interactions. *IEEE Computer Graphics and Applications*, 29(3):52–61, May 2009.

[9] C. Dunne, N. H. Riche, B. Lee, R. A. Metoyer, and G. G. Robertson. GraphTrail: Analyzing Large Multivariate and Heterogeneous Networks while Supporting Exploration History. In *ACM Conference on Human Factors in Computing Systems*, pages 1663–1672, 2012.

[10] R. Eccles, T. Kapler, R. Harper, and W. Wright. Stories in GeoTime. *IEEE Symposium on Visual Analytics Science And Technology*, 7(1):19–26, 2007.

[11] D. Gotz, Z. When, J. Lu, P. Kissa, N. Cao, W. H. Qian, S. X. Liu, and M. X. Zhou. HARVEST: An Intelligent Visual Analytic Tool for the Masses. In *International Workshop on Intelligent Visual Interfaces for Text Analysis*, pages 1–4, New York, New York, USA, Feb. 2010. ACM Press.

[12] D. Gotz and M. X. Zhou. Characterizing users visual analytic activity for insight provenance. *Information Visualization*, 8(1):42–55, Jan. 2009.

[13] D. P. Groth and K. Streefkerk. Provenance and annotation for visual exploration systems. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1500–1510, 2006.

[14] J. Heer, J. Mackinlay, C. Stolte, and M. Agrawala. Graphical histories for visualization: supporting analysis, communication, and evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1189–1196, 2008.

[15] J. Heer, F. B. Viégas, and M. Wattenberg. Voyagers and Voyeurs: Supporting Asynchronous Collaborative Visualization. *Communications of the ACM*, 52(1):87–97, Jan. 2009.

[16] R. R. Hightower, L. T. Ring, J. I. Helfman, B. B. Bederson, and J. D. Hollan. Graphical Multiscale Web Histories: A Study of PadPrints. In *ACM Symposium on User Interface Software and Technology*, pages 121–122, New York, New York, USA, Nov. 1998. ACM Press.

[17] N. Kadivar, V. Chen, D. Dunsmuir, E. Lee, C. Qian, J. Dill, C. Shaw, and R. Woodbury. Capturing and supporting the analysis process. In *IEEE Symposium on Visual Analytics Science And Technology*, pages 131–138. IEEE, 2009.

[18] D. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. *Mastering the Information Age - Solving Problems with Visual Analytics*. 2010.

[19] S. R. Klemmer, M. Thomsen, E. Phelps-Goodman, R. Lee, and J. A. Landay. Where do web sites come from? Capturing and Interacting with Design History. In *ACM Conference on Human Factors in Computing Systems*, pages 1–8, New York, New York, USA, Apr. 2002. ACM Press.

[20] D. Kurlander and S. Feiner. Editable graphical histories. In *IEEE Workshop on Visual Languages*, pages 127–134. IEEE Comput. Soc. Press, 1988.

[21] K.-L. Ma. Image graphs - a novel approach to visual data exploration. In *IEEE Conference on Visualization*, pages 81–88, Oct. 1999.

[22] C. Meng, M. Yasue, A. Imamiya, and M. Xiaoyang. Visualizing histories for selective undo and redo. In *Asian Pacific Computer and Human Interaction*, pages 459–464. IEEE Comput. Soc, 1998.

[23] C. North, R. Chang, A. Endert, W. Dou, R. May, B. Pike, and G. Fink. Analytic provenance: process+interaction+insight. In *ACM Transactions on Computer-Human Interaction*, pages 33–36. ACM, May 2011.

[24] W. Pike, J. Bruce, B. Baddeley, D. Best, L. Franklin, R. May, D. Rice, R. Riensche, and K. Younkin. The Scalable Reasoning System: Lightweight visualization for distributed analytics. *Information Visualization*, 8(1):71–84, 2009.

[25] C. Scheidegger, H. Vo, D. Koop, J. Freire, and C. Silva. Querying and creating visualizations by analogy. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1560–1567, 2007.

[26] B. Shneiderman. The Eyes Have It : A Task by Data Type Taxonomy for Information Visualizations. In *IEEE Symposium on Visual Languages*, pages 336–343, 1996.

[27] Y. B. Shrinivasan and J. J. van Wijk. Supporting the Analytical Reasoning Process in Information Visualization. In *ACM Conference on Human Factors in Computing Systems*, pages 1237–1246, New York, New York, USA, Apr. 2008. ACM Press.

[28] Y. B. Shrinivasan and J. J. van Wijk. Supporting Exploration Awareness in Information Visualization. *IEEE Computer Graphics and Applications*, 29(5):34–43, Sept. 2009.

[29] Y. L. Simmhan, B. Plale, and D. Gannon. A survey of data provenance in e-science. *ACM SIGMOD Record*, 34(3):31–36, Sept. 2005.

[30] J. J. Thomas and K. A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Center, 2005.