DOI: 10.1111/cgf.13997
Eurographics Conference on Visualization (EuroVis) 2020
M. Gleicher, T. Landesberger von Antburg, and I. Viola
(Guest Editors)

Volume 39 (2020), Number 3

Resolving Conflicting Insights in Asynchronous Collaborative Visual Analysis

Jianping Kelvin Li, Shenyu Xu, Yecong (Chris) Ye, and Kwan-Liu Ma

University of California, Davis, USA

Abstract

Analyzing large and complex datasets for critical decision making can benefit from a collective effort involving a team of analysts. However, insights and findings from different analysts are often incomplete, disconnected, or even conflicting. Most existing analysis tools lack proper support for examining and resolving the conflicts among the findings in order to consolidate the results of collaborative data analysis. In this paper, we present CoVA, a visual analytics system incorporating conflict detection and resolution for supporting asynchronous collaborative data analysis. By using a declarative visualization language and graph representation for managing insights and insight provenance, CoVA effectively leverages distributed revision control workflow from software engineering to automatically detect and properly resolve conflicts in collaborative analysis results. In addition, CoVA provides an effective visual interface for resolving conflicts as well as combining the analysis results. We conduct a user study to evaluate CoVA for collaborative data analysis. The results show that CoVA allows better understanding and use of the findings from different analysts.

1. Introduction

Exploratory visual analysis allows analysts to explore datasets based on visualizations of different data dimensions and characteristics [Kei01, JKMG07]. However, a thorough exploration of a large and complex dataset by a single person requires tremendous time and effort. Collaborative visual analytics allows a team of analysts to collectively explore large and complex datasets [HA08, IES*11]. Analysts in collaboration need to share, understand, evaluate, and build on each other's findings, which makes collaborative sensemaking a complex and demanding process. Researchers in collaborative data analysis and visualization have proposed methods for combining analysis results [CYM*10], switching between shared and private results via branching [MBM*12], maintaining team awareness [MT14], transferring knowledge [ZGI*17], and reporting results via storytelling [MHK*19]. Many systems for collaborative data analysis and visualization have also been introduced [VWVH*07, HVW07, CY13, ST14]. However, identifying and resolving conflicts in the collaborative analysis results have not been considered, and such functionality is missing in collaborative visualization systems. Conflicts in the results from different analysts are often inevitable, and methods for conflict resolution are necessary for using collaborative analytics in real-world applications. This is particularly important for collaboration that is asynchronous and geographically distributed, because the analysts cannot directly communicate with each other. Effective methods are needed for resolving conflicts along with the tasks of understanding, evaluating, and building upon the insights gained by individual analysts.

In this paper, we present CoVA, a visual analytics system with a framework designed for managing collaborative analysis results and resolving conflicting insights to better support asynchronous collaborative sensemaking. CoVA provides effective visual interfaces for recording and structuring of findings from exploratory data analysis into a node-link diagram, which we call it the Insight Graph. Insight Graph uses nodes to represent insights and edges to represent relations between insights, Provenance of the insights can be attached to the nodes and edges in the form of visualizations and annotations. Analysts can interactively investigate the Insight Graph and review the insight provenance. To detect and resolve conflicts in collaborative analysis results, we contribute a conceptual design for specifying insights and their provenance as visualizations using declarative grammar. By storing visualizations as text files with a unified format, CoVA can leverage the distributed revision control mechanism used in software development to manage the findings and detect conflicts for collaborative analysis. Furthermore, we contribute a novel framework that leverages this conceptual design to allow seamless interoperation between the system components of exploratory visual analysis, insight management, and collaboration process management. CoVA also provides a visual interface for showing and resolving the conflicts in the insights and combining the results properly. We demonstrate CoVA's effectiveness and usefulness in collaborative data analysis through two case studies with two real world datasets. In addition, a user study is conducted to assess the impact of CoVA's conflict resolution methods and visual interface to collaborative data analysis. Results show that CoVA allows better understanding of the results from collaborative data analysis, leading to more insights derived from the results.

2. Related Work

CoVA builds on prior work on insight management, collaborative visual analysis, and research that across these two areas.

2.1. Insight Management

Graph-based tools [Nov91, BB96, SvW08, ZGI*17] have been used extensively for managing the knowledge extracted from data. A summary of insight organization tools based on graph was provided by Eppler [Epp06]. These tools typically use nodes to represent concepts and edges to represents the relationships between concepts. However, these tools need to be incorporated in data analytics and visualization systems in order for them to be useful for managing insights during data exploration. Researchers have developed systems that use visual analytics for exploring data and graph-based tool for managing the findings and insights. Canas et al. developed CmapTools [CHC*04, CCH*05] to integrate knowledge management and information visualization with concept maps. Yang et al. [YXRo07] used the term nuggets to refer to valuable information and insights hidden in datasets, and they developed the Nugget Management System to facilitate insight management and rediscovery by using visualization to present insights based on similarity. Stasko et al. [SGL08] developed Jigsaw, an interactive system that shows connections and relationships between entities across documents. Chen et al. [CYR09] argued that an insight consisted with three components: a fact, a knowledge base, and subjective evaluations. In addition, Chen et al. [CBY10, CAB*11, CY13] pointed out that insight management tools should provide automated features to aid the sensemaking process, thus proposed a general framework [CY13] as well as design considerations for individual components [CYR09, CBY10, CAB*11] for collaborative insight management. In CoVA, the exploratory visual analysis component is tightly integrated with insight management, where analysts can easily create visualizations to explore data and organize the externalized insights.

Besides organizing insights, recording insight provenance [NCE*11, RESC16] is useful for developing shared understandings of insights within a team of analysts, which is crucial for collaborative sensemaking. Many analysis systems and tools record the history of analysis process for provenance tracking, including GRASPARC [BPW*93], GraphTrail [DHRL*12], and VisTrails [CFS*06]. In addition to the history of analysis process, more information can be tracked for insight provenance. Derthick and Roth [DR01] developed a data exploration system that allows branching the history of user operations with navigation across time and scenarios using a tree-structured visualization. Gotz et al. [GZ08] built a system that tracks and summarizes user activities for insight provenance. Sarvghad et al. [ST14] exploited analysis history for supporting collaborative analysis, where the data dimension coverage of previous analysis is visualized to help identify unexplored regions and suggest the next step for analysis. CoVA builds on these works to support effective management of insights, insight provenance, and the process of collaborative analysis. We use graph-based representations of insights for interactively externalizing and structuring the insights from exploratory visual analysis. CoVA also allows visualizations to be attached as insight provenance in the graph. Analysts can interactively evaluate, refine, and extend the graph. The changes made to the graph are automatically recorded for tracking the process of collaborative data analysis.

2.2. Collaborative Data Analysis and Visualization

Enabling collaboration was identified as a major challenge for the field of visual analytics by Cook and Thomas [CT05] and Isenberg et al. [IES*11]. A large amount of work provided system design guidelines [WK06, HVW07, VWVH*07, HA08, MT14], software infrastructure [BE14, MBM*12, LCM15], and user behavior studies [ITC08, Rob08, IFM*10] for collaborative visual analysis. Recently, data science and computational notebooks [KRKP*16, RNA*17, RTH18] become a popular medium for collaborative data analysis. While all computational notebooks are limited by the linear document nature, analysis results and findings with hierarchical structures cannot be effectively presented and managed.

For insight management in collaborative data analysis, several researchers have built systems to support sharing and combining findings among a team of analysts. Chung et al. [CYM*10] presented VizCept, a visual analytics system that allows integrating individual findings in a shared node-link diagram. To support synchronous collaboration, VizCept updates the shared node-link diagram immediately when users add new nodes or links. Mahyar et al. [MT14] created CLIP, a tool for sharing findings in collaborative sensemaking. CLIP automatically indicates the common entities in the findings from different analysts to increase the awareness and improve work coordination within a team of analysts. Xu et al. [XBL*18] built Chart Constellations, a system that provides summarization of the visualizations created for collaborative data analysis. The Chart Constellations system organizes and projects all the visualizations into a single view, where visualizations containing related insights are placed closer to each other. As these systems only focused on showing the similar findings in the collaborative analysis results, they do not detect and show the conflicts in the findings. The approach that is more similar to CoVA in managing the process of collaborative visualization is the branch-explore-merge workflow by McGrath et al. [MBM*12], in which the analysts can diverge from the shared analysis results to explore independently and then merge new findings to the shared results. However, this workflow is designed for synchronous and co-located collaboration, where conflicts in the findings can be resolve by the analysts via verbal communications. To our knowledge, the problem of detecting and resolving conflicting insights in collaborative data analysis has not been addressed. Our work is the first step to develop methods for conflict identification and resolution in asynchronous collaborative visual analytics. We leverage declarative visualization grammar to allow visual analytics systems to adopt the revision control mechanism that has been proven to be effective for software engineering. In particular, CoVA uses Git [Spi12], which is a popular revision control system, for managing all the analysis results and insight provenance.

3. Design

Here, we explain our design considerations and describe our system framework and user interface.

3.1. Design Considerations

Through a review of related work and systems, we identify the following set of high-level tasks that collaborative visual analytics systems need to support:

- **HT1: Flexible Data Exploration.** Allowing analysts to use background knowledge for creating and using data visualizations is important in the data exploration process [Kei01, DOL03]. Systems should support flexible exploratory visual analysis, where analysts can expressively create visualizations to use different approaches for gaining insights from the data [KGS09].
- **HT2: Interactive Insight Externalization.** Visual analytic systems for data exploration should allow insights to be interactively externalized from data visualizations and organized in a way that is easy to understand, refine, and expand [BCB09, IES*11]. The task of insight externalization should be tightly coupled with the tasks of visual analysis to support effective data exploration [HA08, IES*11, KS11].
- **HT3: Insight Provenance Tracking.** Understanding how insights were derived from the data and tracking the analysis process are important [CT05, XAJK*15]. The results of data exploration should include insight provenance to support reviewing and tracking the analysis process [CT05].
- **HT4: Effective Result Sharing.** Insights and findings from multiple analysts need to be effectively shared and combined [HA08, IES*11]. A team of analysts should allow to evaluate, refine, and build on the results from each other.

For resolving conflicts in collaborative analysis, an effective mechanism for conflict detection and resolution needs to be incorporated into the workflow. To achieve this, we have identified a set of design considerations.

- **DC1**. The system should able to detect conflicts in the externalized insights and the visualizations and annotations used for insight provenance.
- **DC2**: System functionalities should be provided for assisting analysts to evaluate and resolve conflicts.
- **DC3**: The conflict resolutions should be tracked to allow analysts to re-evaluate, revert, and refine the resolutions.

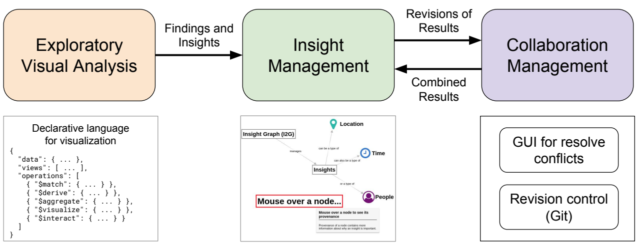

Figure 1: *The system framework of CoVA has three components: Exploratory Visual Analysis, Insight Management, and Collaboration Management.*

3.2. System Framework and User Interface

To support all the tasks with taking the design considerations together, we have developed CoVA's system framework with three major components: 1) Exploratory Visual Analysis (EVA), 2) Insight Management, and 3) Collaboration Management, as illustrated in Figure 1. The EVA component leverages declarative visualization languages for allowing analysts to explore data by creating different visualizations. Declarative languages can provide good insight provenance as data transformations and visual encoding are clearly described, which help a team of analysts to better understand each other's findings. The Insight Management component leverages Insight Graph for externalizing and organizing findings. Node-link diagrams and graphs can provide flexibility for representing insights and allow visualizations and annotations to be attached to nodes and links for tracking insight provenances. Furthermore, results represented as graphs can be easily merged. The Collaboration Management component employs Git for managing the collaboration process and tracking the changes of Insight Graph and the associated insight provenance. By using declarative visualization languages for specifying all the visualizations and Insight Graphs, we can effectively use Git for revision control and track the history of changes in the analysis process.

The primary user interface of CoVA for collaborative analysis of a dataset is shown in Figure 2. The EVA component allows users to perform common data transformations and plot the results using different types of visualizations (A). Declarative specification of visualizations can be entered via the editor (B) for creating interactive visualizations, and the panel on the left of the editor lists all the attributes of the selected dataset. By default, CoVA's EVA component uses P4 [LM18], a GPU-accelerated visualization toolkit, which allows CoVA to handle large datasets with multi-million data items. Users can switch to use Vega [SRHH16] and Vega-Lite [SMWH17], or other declarative visualization libraries.

The use of declarative languages for specifying visualizations allows CoVA to support HT1. To address DC1 and DC2 while adding support for HT2 and HT3, we have developed Insight Graph, an interactive node-link diagram for externalizing and organizing insights from the EVA component. In Insight Graph, insights or analysis artifacts (e.g., data entities and hypotheses) are represented as nodes, and relations between artifacts are represented as links. Different types of insights can be represented by different node icons (i.e. temporal insights can be represented by a clock icon). As shown in Figure 2C, the insights can be externalized from the EVA component and organized in Insight Graph. Throughout the process of collaborative data exploration, the analysts use the Revision Control panel (Figure 2D) to save their results and share with other analysts. Users can review the process of collaborative data exploration by pressing the review button on the top right corner of the user interface. As shown in Figure 3, CoVA provides a simple tree visualization for analysts to understand and review each step of the exploration process. With the latest updates displayed on the top, each tree node is a commit of the exploration results. The node size encodes the number of changes in every commit. The color indicates different analysts or different branches of the exploration. Clicking on a node on the tree visualization shows the results of the associated commit on the right panel. Analysts can review the changes

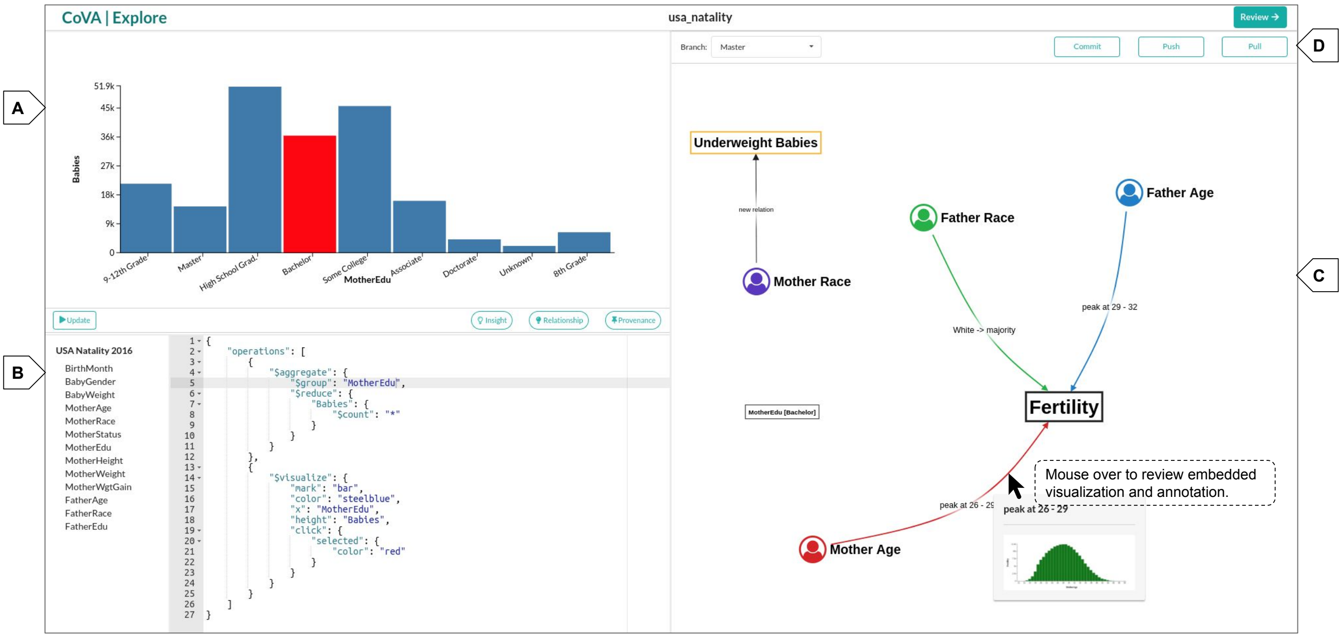

Figure 2: *CoVA's primary user interface for specifying interactive visualizations (A) via declarative language with a editor (B). Insights and findings can be externalized and organized in Insight Graph (C), which can be managed by revision control (D) with the Git workflow (e.g., branch, commit, push, pull).*

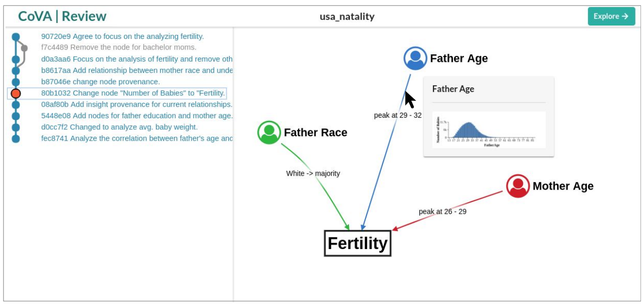

Figure 3: *CoVA shows the history of the data exploration process using a simple tree visualization (left), allowing users to track the changes in Insight Graph (right).*

in Insight Graph by "time-traveling" up and down along the tree, providing an intuitive understanding of the process of collaborative exploration.

To better support insight externalization, Insight Graph provides a set of user interactions for structurally organizing insights and managing insight provenance. Figure 4A illustrates the user interactions and features provided in Insight Graph. Clicking the right mouse button brings up a context menu (A1) for adding nodes Clicking on a node brings up another menu (A2) for creating a links between two nodes, editing node properties, or removing the selected node from Insight Graph. Similarly, clicking on a link brings up a similar menu for editing link properties or removing the selected link. For editing a node or a link, a floating panel (A3) is shown beside the node or link for modifying the properties, such as labels, icons, and colors. Annotations and visualizations can also be attached as insight provenance. When users move the mouse over a node or link in Insight Graph, the attached visualizations and annotations are displayed in a floating widget (A4), so the users can interactively investigate the insight provenance to review all the findings and results. In addition, CoVA provides semi-automatic methods for supporting analysts to easily externalize and organize insights as well as to attach associated visualizations and annotations for insight provenance. After the visualizations are created in CoVA's EVA component, analysts can use the three buttons in CoVA's user interface (Figure 4B) to extract insights and insight provenance from the visualization and structurally organizing them in Insight Graph. The *Provenance* button is for attaching visualizations to a node or link as insight provenance. After the user first clicked the *Provenance* button and selected a node or link, the visualization and its declarative specifications are then attached as insight provenance. The *Insight* and *Relationship* buttons are for generating nodes and links in Insight Graph based on the declarative specification of the current visualization in the EVA component. The *Relationship* button can be used to add a pair of linked nodes to Insight Graph. This mode of extracting insights is enabled when the visualizations depict relationships between two data attributes (i.e. scatterplots and bar charts). The two nodes are based on the x and y axes of the visualization, and the visualization is attached as the insight provenance of the connecting link since it depicts the relationship. By default, the link points from the x-axis attribute to the y-axis attribute. The *Insight* button can also be used to represent a *subset* of the visualized data, instead of the entire visualization, by linking to a selection made by the user

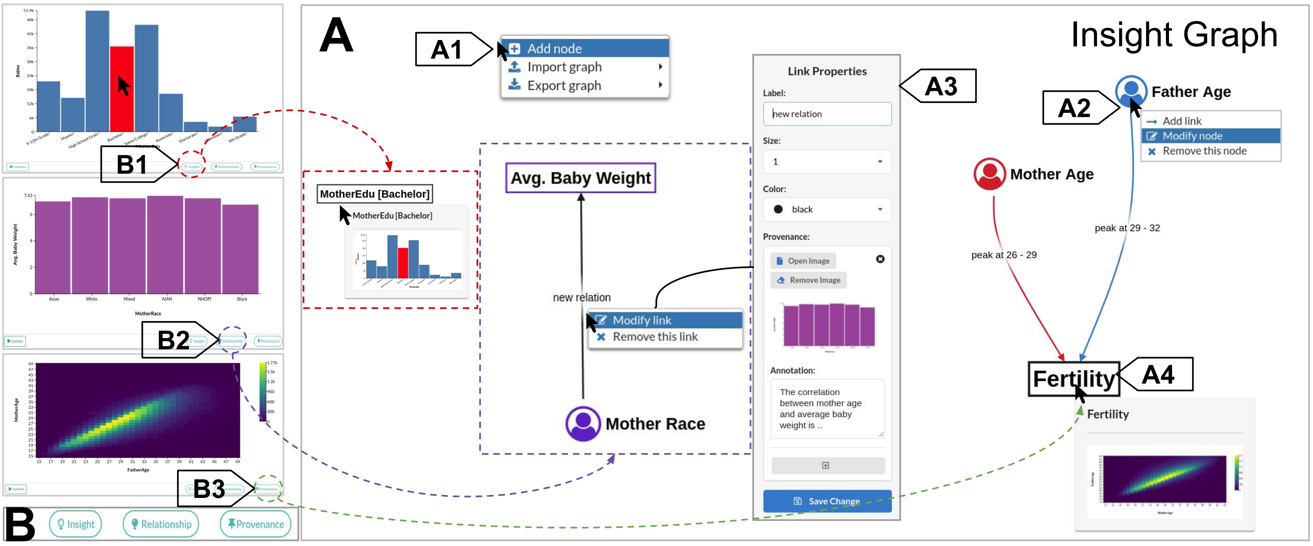

Figure 4: *CoVA provides semi-automatic methods for supporting analysts to easily extract and organize insights as well as to attach associated visualizations and annotations for insight provenance.*

on the visualization (see the first bar chart in Figure 4 on the left). The visualization and selection by the user are also automatically attached as insight provenance to the created node. Users can further add annotations to the nodes and links created using these methods (Figure 4 A3).

3.3. Conflict Detection and Resolution

By using declarative visualization grammars for representing the visualizations and Insight Graph, we can easily adopt the Git revision control workflow, which can support HT4 and address DC2 and DC3. Figure 5 illustrates CoVA's collaborative workflow. Once a new project in CoVA is created with the selected dataset, CoVA initializes a central repository using Git. Users can clone the Git repository to create their own repository, where they can work individually and *commit* their results locally whenever they want to. Users can then share their results by *pushing* their committed results to the central repository, as well as *pulling* the results from the shared repository. To help analysts be aware of the works and findings from others, CoVA's user interface shows the number of new updates from shared repository that can be *pulled* and the number of commits that the user can *push*. As the example shown in the bottom of Figure 5, the user has three commits that can be pushed to and one update that can be pulled from the central repository. When a user pulls new results from the central repository, CoVA leverages Git to detect conflicts and to automatically merge results with no conflicts.

To leverage Git for effectively detecting conflicts and tracking changes in analysis results, we have designed a declarative specification in a JSON format for storing Insight Graphs as files in Git repositories. Each node or link of the Insight Graph is stored in a single line in the declarative specification. The diffing algorithm of Git performs line by line comparisons for text files, and the different

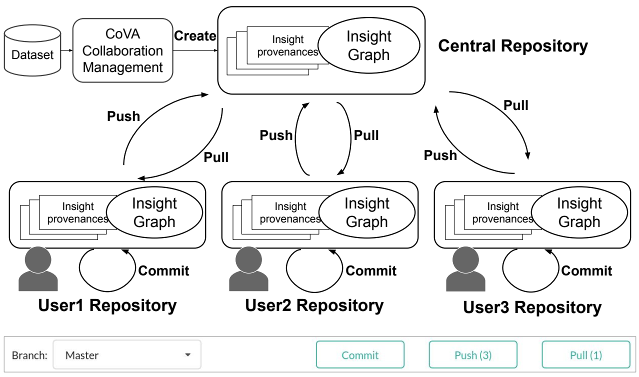

Figure 5: *CoVA's collaborative workflow based on Git for sharing exploration results and managing the process of collaboration.*

lines are either automatically merged or marked for manual conflict resolving. Therefore, CoVA can effectively use Git to identify which nodes and links have conflicts by checking the changes of the lines in the JSON file. Based on the conflicts detected by Git, CoVA parses the associated file contents and identifies the causes of the conflicts. CoVA also uses the insight provenance for comparing and merging the results from different analysts. As declarative visualization specifications are used for insight provenance, Git can handle the conflict detection and merging in the same way. Each declarative visualization specification used as insight provenance for a node or link is saved in a separate file, where the link to the file is saved in the JSON file for the Insight Graph. Using this mechanism, CoVA can effectively detect the following three types of conflicts that commonly occurs during the collaborative analysis process.

- **Property Mismatch**. When two analysts edited the same node or link (i.e., changed the label or another property), merging their results causes this type of conflict.
- **Node Dissonance**. An analyst might edit or add link to a node that has been just removed by another analyst in the latest commit.
- **Provenance Mismatch**. Different analysts might attach different visualizations and annotations to the same node or link, which results in this type of conflict.

If conflicts were detected and the results cannot be merged without users' manual intervention, CoVA provides a graphical interface to help users resolve the conflicts and merge the results. The graphical interface lists each conflict with information about the type of conflicts and the label of the associated node or link. The two different graphs are also displayed for comparison, as shown in Figure 6. From the list of conflicts, the analyst can choose the options for how to resolve the conflicts, which usually either keep the current change or use the previous commit from other analysts. The final merged Insight Graph is shown on the right.

4. Use Cases

Here we provide two use cases to demonstrate how CoVA can be useful for collaboration sensemaking and data exploration.

4.1. Case 1: Global Terrorism

In a collaborative sensemaking scenario, two analysts use the Global Terrorism Dataset [LD07] to explore the terrorist attacks occurred in two different regions: 1) *Europe* and 2) *Middle East & North Africa*. After coordinating among themselves, the two analysts decided explore two different sets of data dimensions. Analyst 1 investigates the temporal patterns of terrorist attacks, while Analyst 2 explores the types of terrorist attacks. By using timeline charts to plot the number of terrorist attacks for each year, Analyst 1 finds out that the number of attacks have significant changes in year 2003, 2004, 2007, and 2014. Using Insight Graph, Analyst 1 externalizes and organizes these findings as shown in the top left of Figure 6, which is *committed* and *pushed* to the system. The three timeline charts used for deriving these insights are attached to the nodes representing the two regions and the terrorist attack. On the other hand, Analyst 2 creates bar charts to visualize the distribution of terrorist attacks by attack types, which show that the top three types of terrorist attacks in *Europe* are bombing/explosion, armed assault, and facility attack. For *Middle East & North Africa*, the top three types of terrorist attacks are bombing/explosion, armed assault, and assassination. These findings are externalized using Insight Graph as shown in the bottom left of Figure 6. When Analyst 2 wants to *commit* and *push* the findings, CoVA's user interface indicates that Analyst 1 has shared some findings. Hence, Analyst 2 *pulls* the other analyst's results from the system to merge their findings. Since two analysts attached different visualizations as insight provenance to the nodes, "Europe", "Middle East & North Africa", and "Terrorism Attack", their results come in conflicts, and Analyst 2 need to resolve them for combining the results. Figure 6 shows how CoVA's visual interface lists and visualizes the findings with conflicts. The three conflicts caused by different insight provenance attached to the three nodes are listed on the left panel, where choices are provided to use either the provenance committed by the Analyst 1 (theirs) or the one committed by her own (ours), or use both. In this case, Analyst 2 choose to include the insight provenance from both analysts. The final graph is then shown on the right side of the user interface, which combines two analysts' findings and shows the important years and the top attack types for the terrorist attacks occurred in the two regions.

4.2. Case 2: Natality

In this case, a team of analysts exploring a dataset with 200K records of newborn babies, which the history of their collaborative analysis is shown in Figure 7. From the tree visualization of the Git histories, we can see that two analysts started the exploration with different paths (teal and gray). Analyst 1 (teal) started by exploring whether parents' ages and age differences have correlation with average birth weight (A). The analyst then organized these results and shared them with the team. By aggregating the data based on parents' ages and age differences and plotting the results in bar charts, Analyst 1 found no strong correlation between these attributes. On the other hand, Analyst 2 (gray) started by analyzing the correlation between parents' ages and fertility (B). By using CoVA to perform data aggregations and visualizations, Analyst 2 found that the highest number of occurrences of having a child is around the age of 28/29 for women and 30/31 for men. After organizing these insights in Insight Graph for sharing with the team, Analyst 2 *pulled* the analytic results generated by Analyst 1 and *pushed* the merged results to the central repository. Because there is no conflict between the results from Analyst 1 and Analyst 2, CoVA automatically merges the results into one (C).

After merging the results, Analyst 1 continued to explore the data by extending the correlation analysis of average baby birth weight to the parents' races. While Analyst 1 was working on this, Analyst 2 reviewed Analyst 1's result and also found no insight related to the average baby birth weight, so Analyst 2 removed the node "Avg. Baby Weight" from the current Insight Graph. Then Analyst 2 *pushed* this change to the central repository before Analyst 1 shared any new results. When Analyst 1 *pushed* the new result containing the "Avg. Baby Weight" node, CoVA detects this conflict in the latest results from Analyst 1 and Analyst 2 and brings up the visual interface for resolving the conflicts and merging the results. Because Analyst 2 removed the node "Avg. Baby Weight" (but not the links connected to it) while Analyst 1 added two links to this node, there are two possible ways (Figure 7 D1 and D2) to automatically merge the two results using the merge method provided by Git. The analyst can also using CoVA's visual interface to decide the best way for merging. In this case, Analyst 1 decided not to continue the exploration with the node "Avg. Baby Weight" and merged the results based on D2, which became the final result committed in D. After merging the results, Analyst 1 continued to explore data related to the "Mother Race" and "Father Race" attributes.

While Analyst 1 was resolving the conflicts using CoVA's visual interface, Analyst 2 continued to explore the data. By realizing the new node "Age Difference" due to merging with Analyst 1's result, Analyst 2 explored the correlation between fertility and age difference, and found that the two attributes have an inverse correlation (large age difference leading to lower chance of having a baby), as

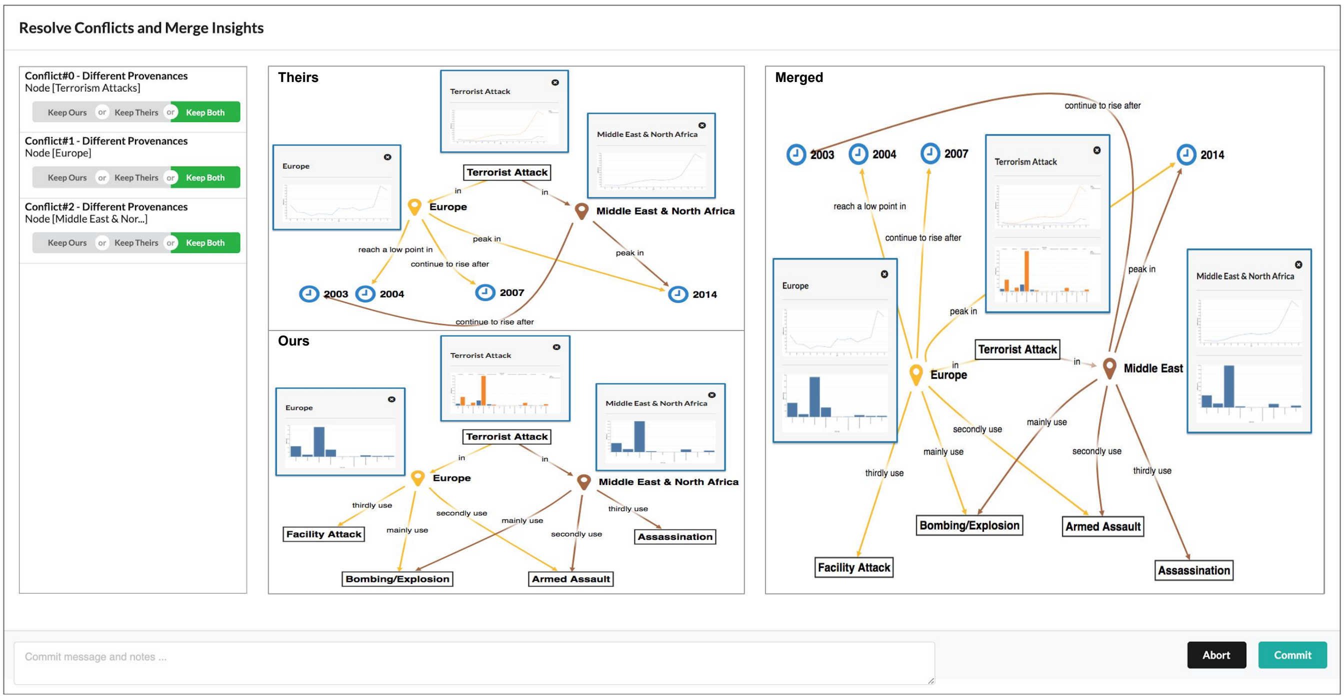

Figure 6: *CoVA's visual interface for resolving conflicts and merge results from different analysts, where the conflicts are listed on the left panel with choices of resolving the conflict based on "Theirs" graph or "Ours" graph, or using both. The choose resolutions are used to merge the results that is shown on the right panel.*

shown in Figure 8D. Analyst 2 then pushed the results to share with the them, but realized Analyst 1 has shared the new result with two new nodes "Mother Race" and "Father Race". So Analyst 2 pulled the new results from the central repository and decided not to use the nodes "Mother Race" and "Father Race" to continue the exploration, when merging with the new results.

At this point, Analyst 3 joined the collaboration and used CoVA's user interface and visualization of the Git tree to review the history and process of the collaborative data exploration. After she realized Analyst 1 has analyzed the correlation related to average baby birth weights but found no insight, Analyst 3 decided to analyze the correlation between parent's ages, as well as the age difference, to the percentage of underweight baby. After performed filtering to get the number of occurrence of underweight babies (weight less than 5.8 pounds) and divide it by the total number of newborn babies to get the percentage for each age and age difference number, Analyst 3 found that the chance for underweight babies starts to increase if the mother is over 43 years old and if the father is over 63. For age difference, no correlation to the percentage of underweight baby is identified. Analyst 3 then organized these findings in Insight Graph and shared them with the team. The final analytic result of the collaborative data exploration is shown in Figure 8.

5. User Study

As incorporating insight detection and resolution for collaborative data analysis has not been explored, very little is known about how such functionalities can impact the results and process. To understand and analyze such impact, we conducted a controlled user study to evaluate the CoVA's revision control functionalities and interface features. Since the focus of our study is reviewing and merging the findings of collaborative data analysis, we simulated an asynchronous collaboration scenario by preparing an initial Insight Graph as the starting point for all the participants. Another set of findings is *pushed* to the participants after they *committed* their first set of findings, and they need to combine the results to continue the data exploration. The findings from the simulated collaborator are the same for all participants. This ensures that each participant can have similar experience with the data exploration process. We compare CoVA to a baseline version of the system without the functionalities of detecting and resolving conflicting insights. We employed a between-subject design in our study. Each participant was first assigned to a group, either CoVA or the baseline system, which kept the same for the entire study session. The results of each group is analyzed to measure the performance and quality of collaborative data analysis.

5.1. Baseline System

The baseline version of our system does not detect conflicts when combining the results of collaborative data analysis. The baseline system uses the same approach in VizCept [CYM*10] for combining analysis results in a node-link diagram, where it merges the nodes with the same labels and treats the nodes with different labels (even with the same visualization attached as insight provenance) as

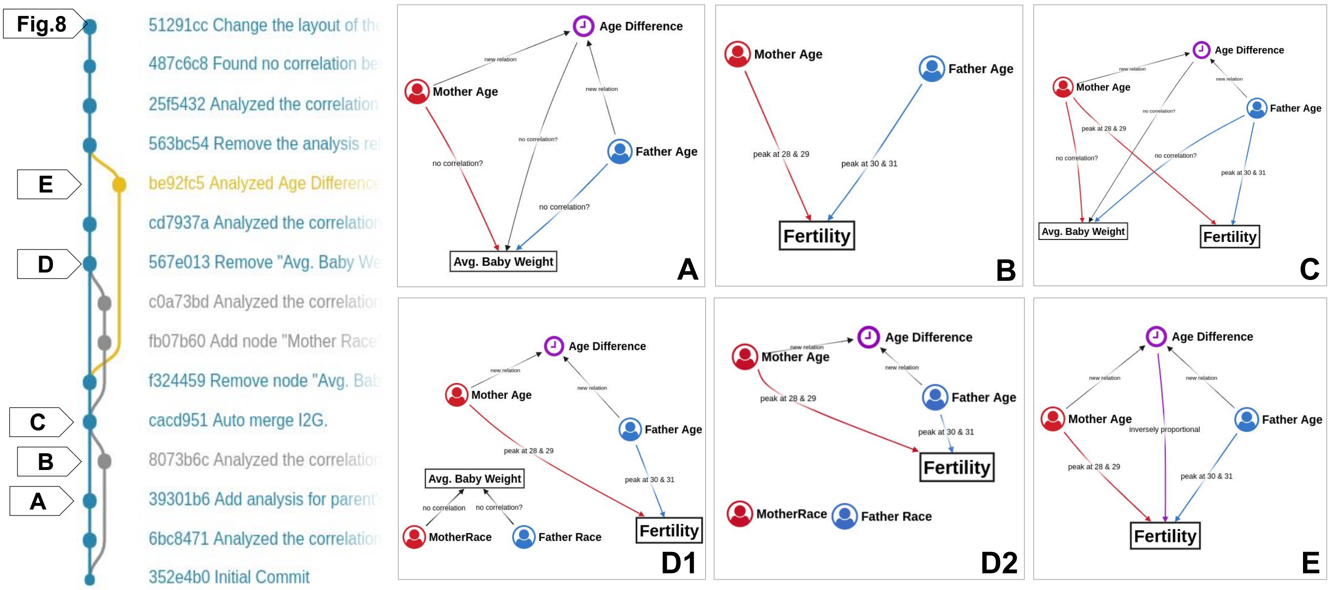

Figure 7: *History for the process of collaborative data exploration showing two analysts started the exploration by committing different findings (A and B). CoVA can automatically merge the findings (C) or suggest different ways (D1 and D2) to merge if detected conflicts. Analysts can choose D1 or D2 as the merged result (D) for continuing the exploration and build on the shared results (E).*

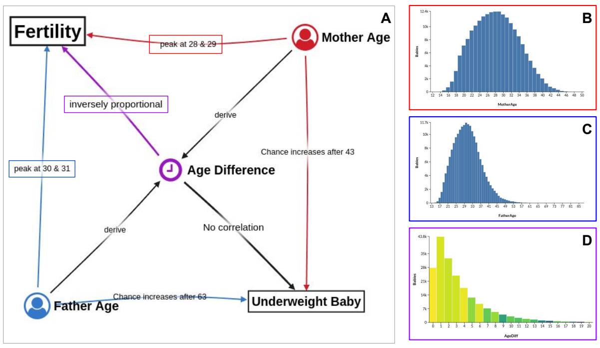

Figure 8: *The final analytic results for the collaborative data exploration. The insights are well organized in Insight Graph (A), with the visualizations in B, C, and D showing the insight provenance stored in the nodes "Mother Age", "Father Age', and "Age Difference", respectively.*

separated nodes. Therefore, the baseline system does not have the visual interface for resolving conflicting insights. For nodes with redundant information in the combined results, the users need to modify the arbitrarily combined graph.

5.2. Design and Procedure

In our user study, each participant needs to go through three stages: 1) training, 2) performing assigned tasks, and 3) exploring the dataset freely to add insights.

Training Stage. Each participant was given a hands-on tutorial for the assigned interface. A short tutorial was given to explain system features and interactions. Participants could then play around with the system and freely ask questions until they felt ready to proceed. During training, participants went through the task with the country datasets from Kaggle [cou18].

Task Stage After training, each participant performed the task with the Global Terrorism Dataset [LD07] from Kaggle [glo18]. To begin this stage, the participant needs to first understand and review the initial Insight Graph which contains three nodes that are associated with the severity of terrorist attacks (number of attacks, number of kills, number of wounds) are connected to five nodes that each represent an active terrorist group. Then the participant is asked to explore different dimensions of the dataset to derive insights about the similarities and differences among the five active terrorist groups. For this task, the participant needs to construct different visualizations to investigate various data dimensions. For convenience, several pre-created charts are created as example visualizations for showing various dimensions of the dataset. The participant can customize these charts or add new visualizations. To complete the task, the participant also needs to use all the interface features, including externalizing insights to Insight Graph and committing their findings using revision control. As we simulated an asynchronous collaboration scenario, another set of findings with conflicts are pushed to the system after the participant *committed* some findings. This set of findings causes a "property mismatch" conflict as described in Section 3.3. In addition, a "provenance mismatch" conflict can occur if the participant adds provenance to one of the nodes or links in the initial result. For the group with the baseline system, all the findings are automatically combined as described in Section 5.1, and the participant can modify the combined results and continue to explore the data. For the group with CoVA, the system automatically checks for conflicting insights and lets the participant to use the visual interface for resolving conflicts and combining the findings. After results are combined, the participant can further adjust or modify the results.

Freeform Analysis Stage. After finishing the analysis task, the participants progressed to the freeform analysis stage to conduct an undirected, freeform analysis and review - there was no explicit "answer the question" task. Participants were given ten minutes to continue their analysis based on the merged results from the task stage. While doing analysis, participants followed think aloud protocol to describe their cognitive processes and actions [FKG93].

After the freeform analysis stage, participants completed a short questionnaire to conclude the study. The questionnaire collected demographic information and queried the perceived usefulness of interface features using a 7-point Likert scale (1 - strongly disagree, 7 - strongly agree). Participants were also encouraged to give any suggestions, and/or criticisms about the system and their experience.

5.3. Participants and Apparatus

We recruited 16 university students (10 male, 6 female) aged between 18 and 34. Because study participants had to play the role as "analysts" in the Freeform Stage, all participants were from computer science who had experience with visualization design and/or data analysis. Figure 12 (P1) lists the familiarity of participants with regards to reading and interpreting visualizations, both interfaces had similarly experienced users. All participants were proficient in English; one was vaguely familiar with the Terrorism dataset (though not at a level that was considered confounding). The hardware apparatus was a 27-inch monitor (Apple Thunderbolt display with 2560×1440 resolution) connected to a MacBook Pro running MacOS Sierra with mouse and keyboard. Both CoVA and Baseline were run using the Chrome browser. Quicktime Player was used to record both audio and screen capture.

6. Results

Here, we analyze and discuss the results of our study.

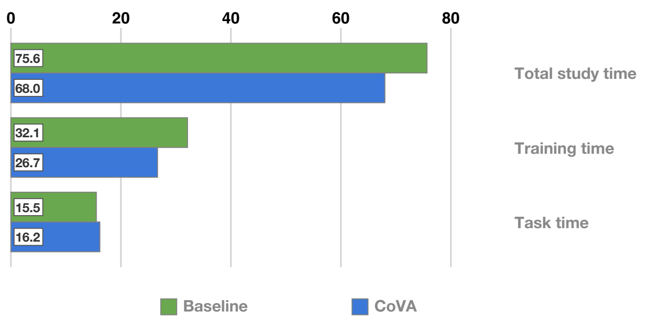

Figure 9: *Average minutes used for the entire study, the training stage, and the task stage. Bars show the mean completion time costs.*

6.1. Time Cost Analysis and Task Stage Performance

Figure 9 shows the time cost for the entire study, training stage, and task stage. Overall, the sessions generally lasted around 60-80 minutes. Since participants need to learn how to use many system functionalities and user interface features, the time cost in the training stage consists around $30-40\%$ of the total time cost. After the training stage, all participants finished the assigned tasks within 20 minutes in the task stage. In the beginning of the task stage, participants in both groups have similar usage patterns of the system. To review the findings in the initial graph, they typically went quickly over each node and link to check the provenance visualizations to understand and review the findings. Then participants created different visualizations to explore the dataset based on the assigned task. After they committed their findings and pulled in the new results from system, participants with the baseline system needed to review an automatically combined graph, in which they reviewed using the same approach for reviewing the initial graph. For the participants with CoVA, the visual interface for showing conflicting insights were used to review and combine the results. As indicated by Figure 9, participants average time for completing the assigned task between both groups is not significantly different. As we observed, the participants with CoVA spent more time with the visual interface to resolve conflicts and combine the results. However, they needed less time to review and understand combined results. On the other hand, the participants with the baseline system spent more time to review and understand the arbitrarily combined graph. As a result, both groups spent about the same amount of time on average in the task stage.

6.2. Freeform Results

In the freeform stage, the participants with CoVA developed more insights than the participants with the baseline system, as shown by Figure 10. Here we use Welch's t-test for statistical analysis of the results, which provides both the p-value and effect size. The number of new nodes created by participants is significantly higher ($p = 0.0490$) using CoVA ($\mu = 2.50, \sigma = 1.511$) than baseline ($\mu = 1.375, \sigma = 0.916$), where the effect size is 0.9 (large). Participants also created significantly more links ($p = 0.00381$) using CoVA ($\mu = 6.250, \sigma = 2.119$) than baseline ($\mu = 2.750, \sigma = 1.488$), where the effect size is 1.72 (large). For provenance, participants added about the same number of visualizations on average in both groups. This result indicates that the visual interface for resolving conflicts and combining results can encourage users to conduct more data exploration and gain more insights. Figure 11 shows the number of derived nodes and links, which connected to the nodes in the findings from the simulated collaborator. On average, participants created more derived nodes using CoVA ($\mu = 1, \sigma = 0.926$) than baseline ($\mu = 0.625, \sigma = 1.488$), but the effect is not significant ($p = 0.368$), where the effect size is 0.18 (small). For derived links, the number is significantly higher ($p = 0.007$) for participants using CoVA ($\mu = 3.0, \sigma = 1.623$) than baseline ($\mu = 1.375, \sigma = 0.916$), where the effect size is 1.44 (large). This result suggests that users are more likely to use and expand the results from collaborators when using CoVA.

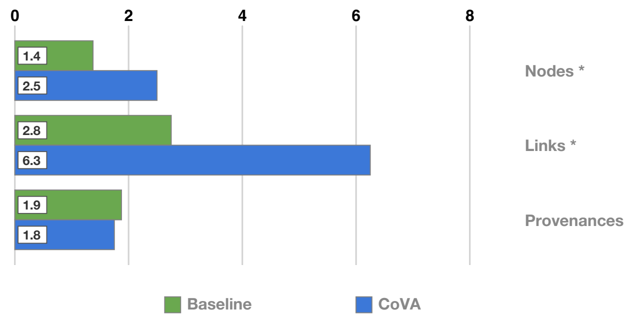

Figure 10: *Number of nodes, links, and provenance created by participants in the freeform analysis stage. Bars show the mean value. Asterisks indicate a statistical difference of $p < 0.05$ between Baseline and CoVA (using a Welch's t-test).*

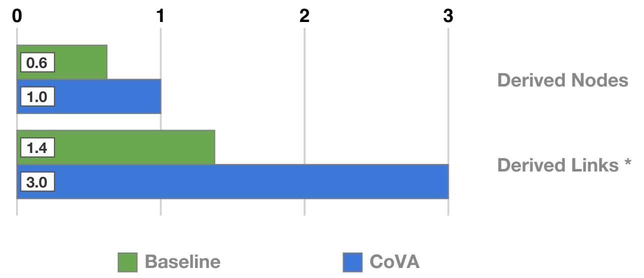

Figure 11: *Number of nodes and links created by participants that are connected to the nodes from the collaborative analysis results **pushed** to them. Bars show the mean value. Asterisks indicate a statistical difference of $p < 0.05$ between Baseline and CoVA (using Welch's t-test).*

6.3. Survey Ratings and Criticisms

Figure 12 lists the responses of the participants to the questionnaire asked at the end of the study. Both systems were rated as easy to learn and use (G1, G2). Based on Mann-Whitney U tests, CoVA rates higher at a statistically significant level ($p < 0.05$) for helping users understand the insights saved by their teammates (S4). For other system functionalities (S1 - S5), both systems were rated positively without significant difference in recording insights (S1), organizing insights (S2), saving insights (S3), and allowing to use teammates' findings (S5). For the interface features shared by both systems (F1 - F6), most ratings are positive. The interface feature of using the "insight" button to add nodes to the Insight Graph received the lowest rating. For the two extra interface features that are specific to CoVA (F7, F8), the ratings were mostly positive. At the end of the questionnaire, we asked participants to select the most and least useful system functionalities and interface features as well as state the reasons. There are two notable observations. First, four out of eight, 50% of participants with the baseline system chose the system functionality for reviewing and understanding the insights saved by teammates (S4) as the least useful feature. This indicates that combining collaborative analysis results using the conventional method in the baseline version is not useful for helping users to understand the results from other analysts. Second, among all the features supported by the system, 13 out of 16, over 80% of participants chose adding and reviewing the provenance stored in nodes/links (F3) as the most useful feature. This result suggests that supporting insight provenance is useful and important for collaborative visual analytics.

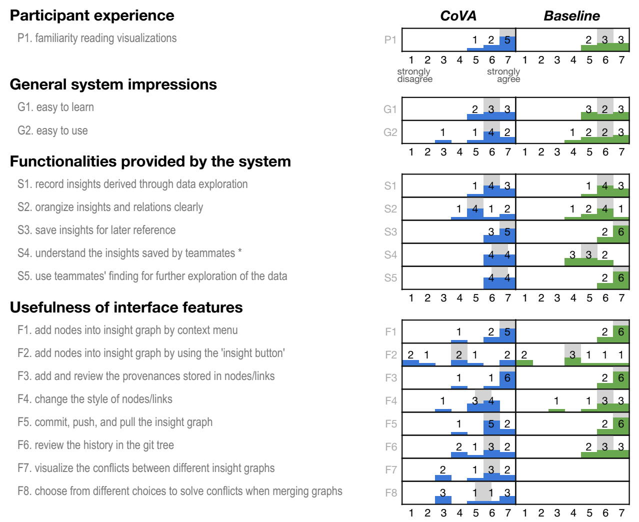

Figure 12: *Participants' ratings about various system aspects during the Review Stage. Median ratings are indicated by gray. Asterisks indicate a statistical difference of $p < 0.05$ between Baseline and CoVA (using Mann-Whitney U tests) for that system aspect.*

7. Discussion

While we focused on evaluating CoVA's system and interface features for supporting collaborative data analysis, the collaboration scenario in our study only has two analysts. For collaboration involving more analysts, further investigation is needed to evaluate the applicability and usefulness of the visual interface for resolving conflicts and combining results. Regarding system usability, all participants are familiar with visualization and programming, as well as the mechanism for revision control of source codes. People without such background might find that the system is more difficult to learn and use. Nonetheless, the user study allows us to better identify the limitations in our system design and provide insights for improving the system as well as adding new interface features.

7.1. Declarative Visualization Language

Representing visualizations and insights using declarative visualization languages allows CoVA to effectively use Git for detecting and resolving conflicts. However, relying on the use of declarative languages for exploratory data analysis is insufficient. In our user study, five participants commented that using the declarative visualization language for creating simple plots was tedious. To improve usability and effectiveness, we can provide both GUI and a declarative language for creating visualizations. Declarative specifications can be automatically generated for the visualizations created using the GUI. We can also employ the method from the visual analytics system by Li et al. [LMR*17], which can allow analysts to use GUI for common visualization tasks and switch to the declarative language for specifying advanced analyses. This can greatly improve the flexibility and usability of the EVA component in our system.

For insight provenance, declarative visualization languages cannot be used to capture the interactive analysis process. While researchers are extending declarative languages to provide better support for interactive visualizations [SWH14], it is possible to save all the interactions made by the analysts. However, not all the interactions of the analysts are relevant to the derived insights. Further research is required to develop methods for effectively saving the interaction history by only logging the relevant interactions and ignoring the irrelevant ones.

7.2. Collaborative Analysis Process

All participants found Git useful for managing the results and process of collaborative data analysis. Three participants expressed that when the system notified them about new results *pushed* by the collaborating analyst, they were unsure whether they should pull the results immediately or continue to work on their own exploration. In general, we should pull the results from collaborating analysts immediately if conflicts can occur. As the system can automatically detect conflicts, we can also inform users whether there are conflicts in the new results to be pulled. In addition, users should also know if any collaborating analysts have committed the same insights. Therefore, combining our method with the techniques used in CLIP [MT14] to inform users about both the conflicting and common insights from collaborating analysts could be a useful new feature. Participants also found CoVA's visual interface for combining insights and resolving conflicts very useful, but two participants wanted to see the temporal history for the nodes or links related to the conflicts. Adding such a feature to the visual interface is worth considering, as it can help user to better decide how to resolve the conflicts and combine the findings. In addition, if the users cannot decide the best way to combine the findings, the *branch* option in Git can be leveraged to combine the results in different ways and save to different branches. To support this new feature, our system can adopt a similar workflow used in [MBM*12] to allow users to continue exploring the data using different branches and decide which branch to use after more insights were confirmed.

7.3. Management of Insight and Provenance

Most participants rated the functionalities provided by the Insight Management module useful. However, there is a need for better support of externalizing insights and recording provenance. Our semi-automatic method described in Section 3.2 for externalizing insights from visualizations is not helpful to users, which is also indicated by the participants' responses to F2 in Figure 12. Participants also complained that manually externalizing and organizing the insights as nodes and links in a graph is tedious. To improve the efficiency for insight externalization, it is worth considering more advanced semi-automatic and automatic methods (e.g., Annotation-Graph [ZGB*16]) for generating an initial graph of insights as the starting point for data exploration. For collaborative data exploration where the analysts generate a large amount of insights, one possible approach to this scalability issue is to provide more interactions, such as zooming and filtering the nodes and links based on user selected parameters. Another approach is to enable grouping of insights in Insight Graph. The grouped insights can be shrunk into a meta node to conserve screen estate. The insight provenance of the meta node should include all the grouped insights. A shrunk node can be expanded in place when needed to reveal the original graph for further detailed exploration.

For insight provenance, manually adding visualizations as provenance to Insight Graph lacks efficiency, as indicated by the result of the freeform analysis stage in our user study. Each participant only added about two visualizations for insight provenance (Figure 10). To provide better support for tracking where the insights came from, effective methods are needed for automatically logging the provenance of insights. Alternatively, other insight management and result reporting methods can be used. For instance, reporting collaborative analysis results via data-driven storytelling [MHK*19] might better encourage analysts to add insight provenance to the results.

For detecting conflicts, our current design can only detect low-level conflicts in the insights that are structurally arranged in Insight Graph. High-level conflicts, such as two insights that suggest two different directions for decision making, cannot be detected in CoVA, in which the analysts need to manually modify the results after understanding the conflicts. To provide better coverage for conflict detection, a thorough analysis of the collaborative sensemaking process to develop a taxonomy of insight conflicts is needed. Further research to develop methods for detecting and resolving different types of conflicts can be based on such a taxonomy. As shown by the user study results, detecting and resolving conflicting insights can better support the collaborative analysis process and lead to more findings. These research directions are promising for advancing collaborative visual analytics.

8. Conclusion and Future Work

We have presented CoVA, a visual analytics system that leverages revision control workflow to facilitate asynchronous collaborative data analysis. While the support for detecting and resolving conflicts is neglected by current collaborative analytics systems and research, our study shows that awareness and understanding of conflicting insights are critical to the findings and overall process of collaborative data analysis. Results of our study suggest that providing a visual interface for resolving conflicts and combining insights can better support collaborative data analysis. In the future, we plan to conduct more user studies with more participants to further evaluate CoVA. We also aim to enhance and extend CoVA based on what we learned from our study, including making insight externalization easier and providing better awareness for both common and conflicting insights.

9. Acknowledgements

This research was supported in part by the U.S. National Science Foundation through grants IIS-320229 and IIS-1741536.

References

[BB96] BUZAN T., BUZAN B.: *The Mind Map Book: How to Use Radiant Thinking to Maximize Your Brain's Untapped Potential*. Plume, Mar. 1996. 2

[BCB09] BIER E. A., CARD S. K., BODNAR J. W.: Principles and tools for collaborative entity-based intelligence analysis. *IEEE transactions on visualization and computer graphics 16*, 2 (2009), 178–191. 3

[BE14] BADAM S. K., ELMQVIST N.: Polychrome: A cross-device framework for collaborative web visualization. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces* (2014), ACM, pp. 109–118. 2

[BPW*93] BRODLIE K., POON A., WRIGHT H., BRANKIN L., BANECKI G., GAY A.: Grasparc-a problem solving environment integrating computation and visualization. In *Proceedings Visualization'93* (1993), IEEE, pp. 102–109. 2

[CAB*11] CHEN Y., ALSAKRAN J., BARLOWE S., YANG J., ZHAO Y.: Supporting effective common ground construction in asynchronous collaborative visual analytics. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on* (2011), IEEE, pp. 101–110. 2

[CBY10] CHEN Y., BARLOWE S., YANG J.: Click2Annotate: Automated insight externalization with rich semantics. In *2010 IEEE Symposium on Visual Analytics Science and Technology* (Oct. 2010), pp. 155–162. 2

[CCH*05] CAÑAS A. J., CARFF R., HILL G., CARVALHO M., ARGUEDAS M., ESKRIDGE T. C., LOTT J., CARVAJAL R.: Concept maps: Integrating knowledge and information visualization. In *Knowledge and Information Visualization: Searching for Synergies*, Tergan S.-O., Keller T., (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 205–219. 2

[CFS*06] CALLAHAN S. P., FREIRE J., SANTOS E., SCHEIDEGGER C. E., SILVA C. T., VO H. T.: Vistrails: visualization meets data management. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data* (2006), pp. 745–747. 2

[CHC*04] CAÑAS A. J., HILL G., CARFF R., SURI N., LOTT J., GÓMEZ G., ESKRIDGE T. C., ARROYO M., CARVAJAL R.: CmapTools: A knowledge modeling and sharing environment. In *Concept Maps: Theory, Methodology, Technology. Proc. of the First Int. Conference on Concept Mapping*, Cañas A. J., Novak J., González F., (Eds.), vol. 1. Universidad Pública de Navarra, Pamplona, Spain, Sept. 2004, pp. 125–135. 2

[cou18] Countries of the world | kaggle, 2018. URL: `https://www.kaggle.com/fernandol/countries-of-the-world`. 8

[CT05] COOK K. A., THOMAS J. J.: Illuminating the path: The research and development agenda for visual analytics. 2, 3

[CY13] CHEN Y., YANG J.: ManyInsights: A visual analytics approach to supporting effective insight management. *Tsinghua Sci. Technol. 18*, 2 (Apr. 2013), 171–181. 1, 2

[CYM*10] CHUNG H., YANG S., MASSJOUNI N., ANDREWS C., KANNA R., NORTH C.: VizCept: Supporting synchronous collaboration for constructing visualizations in intelligence analysis. In *2010 IEEE Symposium on Visual Analytics Science and Technology* (Oct. 2010), pp. 107–114. 1, 2, 7

[CYR09] CHEN Y., YANG J., RIBARSKY W.: Toward effective insight management in visual analytics systems. In *2009 IEEE Pacific Visualization Symposium* (Apr. 2009), pp. 49–56. 2

[DHRL*12] DUNNE C., HENRY RICHE N., LEE B., METOYER R., ROBERTSON G.: Graphtrail: Analyzing large multivariate, heterogeneous networks while supporting exploration history. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2012), pp. 1663–1672. 2

[DOL03] DE OLIVEIRA M. F., LEVKOWITZ H.: From visual data exploration to visual data mining: A survey. *IEEE Transactions on Visualization and Computer Graphics 9*, 3 (2003), 378–394. 3

[DR01] DERTHICK M., ROTH S. F.: Enhancing data exploration with a branching history of user operations. *Knowledge-Based Systems 14*, 1-2 (2001), 65–74. 2

[Epp06] EPPLER M. J.: A comparison between concept maps, mind maps, conceptual diagrams, and visual metaphors as complementary tools for knowledge construction and sharing. *Inf. Vis. 5*, 3 (June 2006), 202–210. 2

[FKG93] FONTEYN M. E., KUIPERS B., GROBE S. J.: A description of think aloud method and protocol analysis. *Qualitative health research 3*, 4 (1993), 430–441. 9

[glo18] Global terrorism dataset | kaggle, 2018. URL: `https://www.kaggle.com/START-UMD/gtd`. 8

[GZ08] GOTZ D., ZHOU M. X.: Characterizing users #x2019; visual analytic activity for insight provenance. In *2008 IEEE Symposium on Visual Analytics Science and Technology* (Oct. 2008), pp. 123–130. 2

[HA08] HEER J., AGRAWALA M.: Design considerations for collaborative visual analytics. *Information visualization 7*, 1 (2008), 49–62. 1, 2, 3

[HVW07] HEER J., VIÉGAS F. B., WATTENBERG M.: Voyagers and voyeurs: supporting asynchronous collaborative information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2007), ACM, pp. 1029–1038. 1, 2

[IES*11] ISENBERG P., ELMQVIST N., SCHOLTZ J., CERNEA D., MA K.-L., HAGEN H.: Collaborative visualization: definition, challenges, and research agenda. *Information Visualization 10*, 4 (2011), 310–326. 1, 2, 3

[IFM*10] ISENBERG P., FISHER D., MORRIS M. R., INKPEN K., CZERWINSKI M.: An exploratory study of co-located collaborative visual analytics around a tabletop display. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on* (2010), IEEE, pp. 179–186. 2

[ITC08] ISENBERG P., TANG A., CARPENDALE S.: An exploratory study of visual information analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2008), ACM, pp. 1217–1226. 2

[JKMG07] JANKUN-KELLY T., MA K.-L., GERTZ M.: A model and framework for visualization exploration. *IEEE Transactions on Visualization and Computer Graphics 13*, 2 (2007). 1

[Kei01] KEIM D. A.: Visual exploration of large data sets. *Communications of the ACM 44*, 8 (2001), 38–44. 1, 3

[KGS09] KANG Y.-A., GORG C., STASKO J.: Evaluating visual analytics systems for investigative analysis: Deriving design principles from a case study. In *2009 IEEE Symposium on Visual Analytics Science and Technology* (2009), IEEE, pp. 139–146. 3

[KRKP*16] KLUYVER T., RAGAN-KELLEY B., PÉREZ F., GRANGER B. E., BUSSONNIER M., FREDERIC J., KELLEY K., HAMRICK J. B., GROUT J., CORLAY S., ET AL.: Jupyter notebooks-a publishing format for reproducible computational workflows. In *ELPUB* (2016), pp. 87–90. 2

[KS11] KANG Y.-A., STASKO J.: Characterizing the intelligence analysis process: Informing visual analytics design through a longitudinal field study. In *2011 IEEE conference on visual analytics science and technology (VAST)* (2011), IEEE, pp. 21–30. 3

[LCM15] LI J., CHOU J.-K., MA K.-L.: High performance heterogeneous computing for collaborative visual analysis. In *SIGGRAPH Asia 2015 Visualization in High Performance Computing* (2015), ACM, p. 12. 2

[LD07] LAFREE G., DUGAN L.: Introducing the global terrorism database. *Terrorism and political violence 19*, 2 (2007), 181–204. 6, 8

[LM18] LI J. K., MA K.-L.: P4: Portable parallel processing pipelines for interactive information visualization. *IEEE transactions on visualization and computer graphics* (accepted in 2018). (preprint: https://ieeexplore.ieee.org/document/8468065). 3

[LMR*17] LI J. K., MUBARAK M., ROSS R. B., CAROTHERS C. D., MA K.-L.: Visual analytics techniques for exploring the design space of large-scale high-radix networks. In *IEEE International Conference on Cluster Computing* (2017), pp. 193–203. 11

[MBM*12] McGrath W., Bowman B., McCallum D., Hincapié-Ramos J. D., Elmqvist N., Irani P.: Branch-explore-merge: facilitating real-time revision control in collaborative visual exploration. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces* (2012), ACM, pp. 235–244. 1, 2, 11

[MHK*19] Mathisen A., Horak T., Klokmose C. N., Grønbæk K., Elmqvist N.: Insideinsights: Integrating data-driven reporting in collaborative visual analytics. In *Computer Graphics Forum* (2019), vol. 38, Wiley Online Library, pp. 649–661. 1, 11

[MT14] Mahyar N., Tory M.: Supporting communication and coordination in collaborative sensemaking. *IEEE transactions on visualization and computer graphics 20*, 12 (2014), 1633–1642. 1, 2, 11

[NCE*11] North C., Chang R., Endert A., Dou W., May R., Pike B., Fink G.: Analytic provenance: process+ interaction+ insight. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (2011), ACM, pp. 33–36. 2

[Nov91] Novak J.: Clarify with concept maps. *The science teacher* (1991). 2

[RESC16] Ragan E. D., Endert A., Sanyal J., Chen J.: Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes. *IEEE transactions on visualization and computer graphics 22*, 1 (2016), 31–40. 2

[RNA*17] Rädle R., Nouwens M., Antonsen K., Eagan J. R., Klokmose C. N.: Codestrates: Literate computing with webstrates. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (2017), pp. 715–725. 2

[Rob08] Robinson A. C.: Collaborative synthesis of visual analytic results. In *Visual Analytics Science and Technology, 2008. VAST'08. IEEE Symposium on* (2008), IEEE, pp. 67–74. 2

[RTH18] Rule A., Tabard A., Hollan J. D.: Exploration and explanation in computational notebooks. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–12. 2

[SGL08] Stasko J., Görg C., Liu Z.: Jigsaw: supporting investigative analysis through interactive visualization. *Information visualization 7*, 2 (2008), 118–132. 2

[SMWH17] Satyanarayan A., Moritz D., Wongsuphasawat K., Heer J.: Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2017), 341–350. 3

[Spi12] Spinellis D.: Git. *IEEE Softw. 29*, 3 (May 2012), 100–101. 2

[SRHH16] Satyanarayan A., Russell R., Hoffswell J., Heer J.: Reactive vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE transactions on visualization and computer graphics 22*, 1 (2016), 659–668. 3

[ST14] Sarvghad A., Tory M.: Exploiting history to reduce interaction costs in collaborative analysis. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)* (Oct. 2014), pp. 273–274. 1, 2

[SvW08] Shrinivasan Y. B., van Wijk J. J.: Supporting the analytical reasoning process in information visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2008), CHI '08, ACM, pp. 1237–1246. 2

[SWH14] Satyanarayan A., Wongsuphasawat K., Heer J.: Declarative interaction design for data visualization. In *Proceedings of the 27th annual ACM symposium on User interface software and technology* (2014), pp. 669–678. 11

[VWVH*07] Viegas F. B., Wattenberg M., Van Ham F., Kriss J., McKeon M.: Manyeyes: a site for visualization at internet scale. *IEEE transactions on visualization and computer graphics 13*, 6 (2007). 1, 2

[WK06] Wattenberg M., Kriss J.: Designing for social data analysis. *IEEE transactions on visualization and computer graphics 12*, 4 (2006), 549–557. 2

[XAJK*15] Xu K., Attfield S., Jankun-Kelly T., Wheat A., Nguyen P. H., Selvaraj N.: Analytic provenance for sensemaking: A research agenda. *IEEE computer graphics and applications 35*, 3 (2015), 56–64. 3

[XBL*18] Xu S., Bryan C., Li J. K., Zhao J., Ma K.-L.: Chart constellations: Effective chart summarization for collaborative and multi-user analyses. In *Computer Graphics Forum* (2018), vol. 37, Wiley Online Library, pp. 75–86. 2

[YXRo07] Yang D., Xie Z., Rundensteiner E. A., others: Managing discoveries in the visual analytics process. *ACM SIGKDD Explorations* (2007). 2

[ZGB*16] Zhao J., Glueck M., Breslav S., Chevalier F., Khan A.: Annotation graphs: A graph-based visualization for meta-analysis of data based on user-authored annotations. *IEEE transactions on visualization and computer graphics 23*, 1 (2016), 261–270. 11

[ZGI*17] Zhao J., Glueck M., Isenberg P., Chevalier F., Khan A.: Supporting handoff in asynchronous collaborative sensemaking using knowledge-transfer graphs. *IEEE Transactions on Visualization and Computer Graphics* (2017). 1, 2