Integrating Annotations into Multidimensional Visual Dashboards

Sriram Karthik Badam,[1] Senthil Chandrasegaran,[2] and Niklas Elmqvist[3]

Abstract

Multidimensional data is often visualized using coordinated multiple views in an interactive dashboard. However, unlike in infographics where text is often a central part of the presentation, there is currently little knowledge of how to best integrate text and annotations in a visualization dashboard. In this paper, we explore a technique called FacetNotes for presenting these textual annotations on top of any visualization within a dashboard irrespective of the scale of data shown or the design of visual representation itself. FacetNotes does so by grouping and ordering the textual annotations based on properties of (1) the individual data points associated with the annotations, and (2) the target visual representation on which they should be shown. We present this technique along with a set of user interface features and guidelines to apply it to visualization interfaces. We also demonstrate FacetNotes in a custom visual dashboard interface. Finally, results from a user study of FacetNotes show that the technique improves the scope and complexity of insights developed during visual exploration.

Introduction

Annotations to structured data help expand the analyst understanding during exploratory visual analysis.[1,2] These annotations can be different types: (1) notes added by analysts to target data points to explain a data behavior or insights they developed (as in Sense.us[3]), (2) textual explanations of the context from real world for data items (cf. Contextifier[4]), or simply (3) the textual components within an enhanced text dataset. For example, descriptive text and labels are commonly part of infographics for data-driven storytelling.[5] However, for interactive visualization dashboards,[6,7] annotations have a much less well-defined role; where should they be placed, how should they be used, and what is the most effective way to show their connection to the data visualizations they reference?

For instance, it is common to use histograms to capture frequencies of values rather than individual data points. In such visual interfaces, challenges with viewing annotations include (1) providing mechanisms to access annotations at any scale of data presented (i.e., a data point vs. a group), (2) connecting annotations between views in the dashboard (i.e., understand where the annotation appears on a different view), and (3) exploring annotations through any visual representation (i.e., irrespective of the data operations such as aggregation and sampling, or the visual design).

In this paper, we present FACETNOTES, a technique for storage and presentation of annotations on top of visualization dashboards to aid visual analysis tasks. FacetNotes uses a double-linked storage model that connects the annotations to data points and visual properties (and back). It uses binning techniques to group annotations based on the target visualization and attaches them to corresponding elements in the visualization. Then it further groups the annotations for each visual element based on their textual content. While doing so, it also connects each annotation back to other dimensions within the multivariate dataset represented in the visual dashboard through a *connector plot*.

We introduce the conceptual space for FacetNotes in Section 3 and illustrate our approach. Then we instantiate the technique in a visualization dashboard for multidimensional data (a video demo of the dashboard for FacetNotes can be found at `https://goo.gl/8wGZTy`). As an evaluation, we recruited 9 participants with visualization experience to work with our VA tool with a dataset of flight delays containing text annotations describing the cause of delay. The FacetNotes technique made it possible to effectively explore the annotations along with delay data to isolate the causes of delays and also predict their effects on other data variables (e.g., number of destinations and the geographical region affected). An alternative interface with a conventional annotation model lacking the visual adaptation and grouping features of FacetNotes led to fewer and simpler observations from visual analysis in comparison. Overall, our contributions in this paper include,

---

[1]Apple Inc., Cupertino, CA, USA. Work done during doctoral studies at University of Maryland, College Park; [2]TU Delft, Delft, Netherlands
[3]University of Maryland, College Park, MD, USA

Corresponding author:
Niklas Elmqvist, College of Information Studies, University of Maryland, College Park, MD 20742, USA.
Email: elm@umd.edu

1. An exploration of the design space through FacetNotes for managing and representing annotations on interactive visual dashboards for multidimensional datasets.

2. A visualization dashboard that instantiates FacetNotes along with the UI features to explore the annotations.

3. Results from an evaluation revealing the affordances of our technique as well as its effects on the sensemaking process of analysts during visual analysis.

Background

Annotations are important for insight management,[8] provenance in sensemaking[9], storytelling,[10] and collaboration[3,11–15]. Here, we discuss the role of annotations in visual analysis and previous work in representing them.

The Role of Annotation in Visual Analytics

As described by Heer and Shneiderman,[12] annotations have multiple roles in the iterative process of visual analysis: they can act as a means to (1) convey trends to the system to *derive* new visualizations, (2) explain specific data behaviors within visualizations to the users,[16] (3) capture the real world context for data behaviors (i.e., events behind stock price changes[4]) (4) highlight regions in views and guide the user[17], (5) indicate how selected data items correspond between views, (6) facilitate conversations between users in different settings.[3,15] Furthermore, externalizing the thought process of the analyst through note taking and annotations has been shown to improve the outcomes of the sensemaking process.[18,19] Finally, meta-analysis and identification of connections between annotations to organize them for exploration has also been found to be effective for documenting annotations and consolidating ideas in visual analysis.[20]

Showing Annotations during Visual Analysis

Visualization interfaces for multidimensional data tend to take the form of a dashboard display with multiple, coordinated views (cf. Keshif[21] and InsightsDrive[22]). Being a critical component of the visual analysis process, annotations need an effective model for capture, management, and retrieval. Recognizing this importance, Elias and Bezerianos[23] proposed a set of requirements for supporting annotations in business intelligence (BI) dashboards, covering transparency of annotations, validity and lifetime of annotations, and their visibility and ease of sharing.

Shrinivasan and van Wijk[24] propose an alternate approach consisting of three views: data view, knowledge view, and navigation view. The knowledge view supports the creation of annotations, while the data and the navigation view offers visualizations of the data and the analysis process respectively. They look at annotations as "view states" that can be navigated to and from other view states. Heer and Agrawala[13] also view annotations as a means of communication between collaborating analysts. PanoramicData[25] supported creation of annotations through pen-and-touch interactions across visualizations on a large interactive display.

Visualizing Annotations

Publicly created and shared contextual annotations became popular in online visualization platforms such as Sense.us[26] and Wikimapia.[27] Visual analysis tools such as ManyEyes (now defunct)[15] and Tableau also support annotation and social analysis. The representations of visualizations have evolved from diagrammatic abstractions to more sophisticated interpretations that attach annotations to data, views, and system states. For instance, large volumes of annotations can be visualized together with two-dimensional data visualizations using "Voronoi growth rings" scaled to annotation size, and colored based on authorship.[28]

Annotations are now also used to edit and reorganize underlying data representations in visual analysis. For instance, TreeDyn,[29] a tool for phylogenetic analysis of biomolecules, allows the user to query annotations and use them to select and manipulate data representations. While the focus of this paper is not on automated annotation generation, Kandogan's *Just-in-Time* descriptive analytics[30] focuses on generating annotations based on automated data analysis. Their annotations are interactive, connected to groups of data points, and modifiable by the user.

Visualizing and organizing annotations is a research area in VA that is beginning to receive attention. Kucher et al.[31] propose a technique called *CatCombos* to visualize annotations created through a visual analytic system. Graphical cues are used to represent individual and aggregate annotations. Zhao et al.[20] developed *annotation graphs*, a visualization technique for capturing, grouping, and analyzing annotations to aid data meta-analysis. Closer to our approach is that of Chen and Yang[8], who organize annotations by using predefined categories and tags to mark observations. Their system, ManyInsights, clusters annotations using these tags, and overlays them on one or more relevant views. Our approach differs from ManyInsights in the following ways: (1) we focus on ways to group and order annotations based on the content of both the annotations and the associated data, and (2) we allow the user to view patterns in the datasets associated with grouped annotations to form further insights.

The FacetNotes Technique

Annotations are effective and aid the user's understanding when they can be easily accessed in relevant visualizations. FacetNotes focuses on the *management* and *representation* of annotations rather than their creation on the visualization interface. Our goal is to develop and evaluate techniques for managing, ordering, grouping, and displaying annotations that have already been created. In general, there can be two kinds of annotations: (1) those created by the system or the person collecting data, which makes them part of a *dataset*, or (2) those created by the person analyzing the data through VA interfaces, which typically makes them part of a *view*. These two types arise from distinct stages of the analytical process. However, in terms of content, these two types can capture any information that enhances the structured data points that feed into the visualizations. For instance, they can add an additional context from the real world (e.g., Contextifier[4]) or a common pattern observed from the data (e.g., Sense.us[3]). In both cases, annotations are closely

Annotations captured in specific data granularities

Challenge: Managing and representing individual & aggregate annotations on dashboards with multiple coordinated views
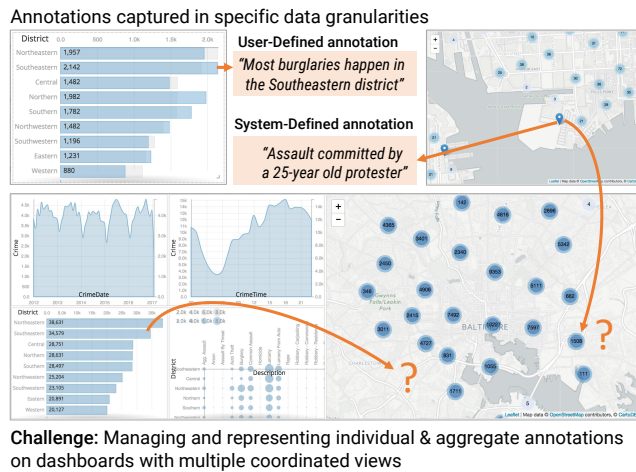
Figure 1. Showing annotations in dashboards. Dashboards show both overview and detail representations (top panels) of data points with directly attached annotations. Connecting these annotations to other views is challenging, as the annotations associated with specific data point may not appear in a single visual element. Popular past approaches have also relied on showing annotations in a separate view away from the actual visual representations (e.g., a comment panel in ManyEyes[15]).

associated with data points and not just particular views of data[23]. Such annotations have in fact been referred to as data-aware annotations in the past.[12] Our approach considers this association between the data and the annotations to have an holistic approach to work with these different types of annotations. Note that we focus on textual annotations (or notes) in this paper.

Here we outline the design rationale for FacetNotes and explain the functions and features that result.
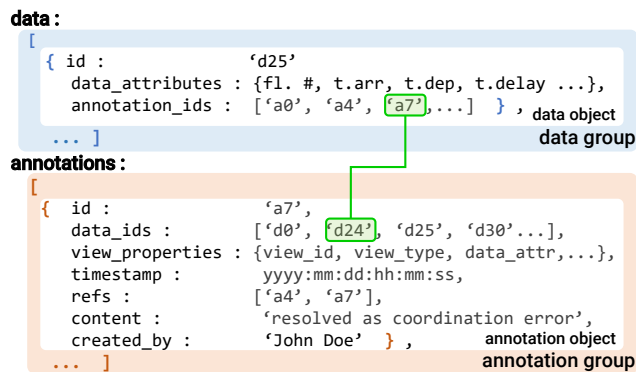
```
data :
[
  { id :              'd25'
    data_attributes : {fl. #, t.arr, t.dep, t.delay ...},
    annotation_ids : ['a0', 'a4', 'a7',...]  } ,  data object
  ... ]                                          data group
annotations :
[
  { id :           'a7',
    data_ids :     ['d0', 'd24', 'd25', 'd30'...],
    view_properties : {view_id, view_type, data_attr,...},
    timestamp :    yyyy:mm:dd:hh:mm:ss,
    refs :         ['a4', 'a7'],
    content :      'resolved as coordination error',
    created_by :   'John Doe'  } ,     annotation object
  ... ]                                annotation group
```

Figure 2. Storing annotations with data. The proposed model for storing annotations along with data points shows a *data* group, containing all data objects and an *annotation* group of all annotation objects. The properties of each data object include references to annotation objects where applicable, and vice versa (green link).

Design Rationale

Challenge: Figure 1 shows a visualization dashboard containing multiple visualizations of the attributes within a crimes dataset from Baltimore in the United States. The dashboard features overview visualizations that aggregate data (e.g., bar charts, heatmaps), and detail visualizations that show individual data points (e.g. a location on a

geographical map). The textual notes with the dataset explaining the charge from each crime can act as annotations that enhance our view of the data. Furthermore, analysts looking at this dataset start to find common patterns across the distributions of the crimes. Surfacing these annotations on any visualization view generated from the dataset is not straightforward as the granularity of the data and the visual representation will differ in each visualization (Figure 1). In fact, this challenge is further complicated when multiple annotations are attached to each data point.

Solution: FacetNotes presents a modular operational approach for adapting annotations to any given view(s). To do this, it assumes that annotations are maintained in a double-linked structure (see Figure 2) and applies grouping, ordering, and aggregation methods based on the target view and content following a series of operations presented in Figure 3. In the following paragraphs, we discuss the rationale that grounds the design of FacetNotes.
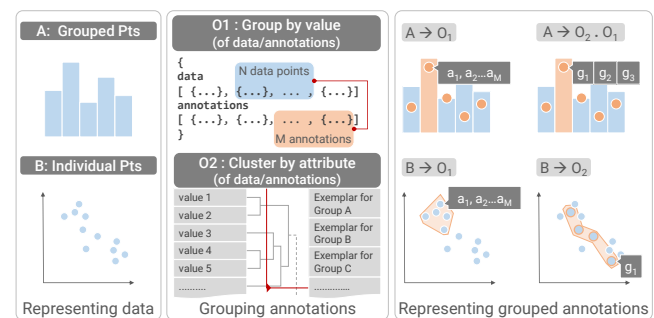

Figure 3. Grouping and representing annotations. These operations for grouping and representing annotations are based on the target view and of associated data objects. Operation O_1 groups annotations by value of selected data or annotation attributes, while O_2 clusters annotations based on the *similarity* between the attribute values.

R1: Associating annotations to any view. To make annotations adaptive and transferable across views, they should be treated as notes attached to both views and the underlying data elements within the views. In FacetNotes, we achieve this through a two-way connection when storing the annotations along with the structured multidimensional data (Figure 2): annotations bear information of the data points to which they are associated as well as metadata, visual properties, and links to other annotations. In turn, data points bear information on their associated annotations. For any visualization, the annotations corresponding to data items can be quickly accessed with such a storage model.

R2: Contextual, view-dependent grouping of annotations. Aggregate views such as histograms, clusters, and treemaps are used to help the user make sense of large datasets. Just as data points are grouped to form these views, annotations should also be grouped in complementary ways. For instance, a bar chart of annual sales can have annotations grouped by the year making it easy to know why sales dropped in, say, 2008. FacetNotes performs this by binning annotations based on the data attributes shown in the visualization. This gives a set of annotations for each visual mark in the view (e.g., each bar in the above bar chart example). At the same time, it also allows further grouping within each bin based on the annotation content and
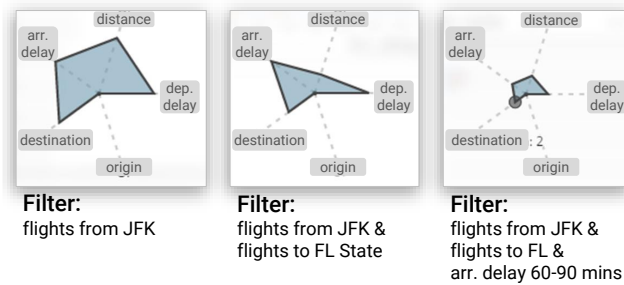
Figure 4. Connector plot. This is a starplot where each "arm" indicates the value *range* on the corresponding data attribute. The data shown is drawn from a flight delays dataset. With progressive filtering of data, the plot shrinks to reflect the reduction in the number of annotations in the group, as the range associated with attributes decreases.

other data attributes. Finally, the grouped annotations can be ordered based on their frequency.

R3: Connecting grouped annotations with other views. Depending on the criteria for grouping annotations, there may be a need to look for patterns in data associated with the grouped annotations. For instance, extending the previous sales example, connecting a particular annotation (reason) for a bump in sales with the investment data attribute. For multidimensional data, FacetNotes achieves this through a *connector plot*, an overview representation attached to the annotation group that shows the range of each attribute of the associated data points (Figure 4).

R4: Adapting to data filtering operations. Annotations should respond to filters applied by the user. For instance, if an annotation is associated with a set $S1$ of data points, and the user applies a filter to reveal a set $S2 : S2 \subset S1$, the annotations should also be filtered to show those associated with $S2$. FacetNotes filters annotations based on the user selections and updates annotation groups (per **R2**) to the dataset in focus. These filters apply to the *connector plot*, too: applying more filters on the dashboard views restricts the range of data displayed, which changes the connector plot.

R5: Maintaining transparency and control. It is important that the user is made aware of criteria and metrics used for grouping and ordering annotations, and has the option to control them. To support user-driven exploration of such metrics, FacetNotes provides options to manipulate the distance metrics and attributes of interest for ordering and grouping procedures in the techniques. For example, for the crimes dataset, multiple attributes such as District, City, and Post Code are related to each other, and the user can specify which of these data attributes can be used for grouping and ordering annotations.

The requirements are by no means exhaustive to support exploration of annotations along with the data in all application contexts. In fact, they are focused mainly on the association between the annotations and the structured data in the datasets themselves, so that they can be shown together on a visualization dashboard. We believe this is a fundamental design question when dealing with annotations on a multidimensional visualization dashboard. It was also the focus of past research in this area.[23] Therefore, the next section digs further deep into how the above requirements are satisfied by FacetNotes in a visualization dashboard.

Having said that, annotations can be rich with metadata that convey their origins. For example, who created them and why and when they were created. This provenance information can be essential to further define how they can be explored along with the data. We imagine more requirements need to be considered to enhance the FacetNotes technique to capture provenance of the annotations. The Discussion section covers these limitations along with future work.

Visualization Dashboard

To instantiate the FacetNotes technique, we developed a dashboard interface for multidimensional data (Figure 5). In this section, we briefly discuss design decisions based on the rationale outlined in the previous section.

Visualizations and interactions. The dashboard shows multiple coordinated views of the data including (1) aggregated views (linecharts and barcharts), and (2) granular representations (scatterplots). For granular representations, data points are automatically grouped in two conditions, (1) based on category if one or both axes are categorical, and (2) based on spatial proximity if the points are clustered too densely (as seen in the geomap). These choices are reflected in Figure 5. All the visualizations are connected by interaction: selecting points or ranges in each view updates other views to the corresponding data.

Representing annotations. For both aggregated and granular views, annotations are represented as tags on top of the individual visualization elements (*per rationale R2*). For example, as seen in the Figure 5, they appear as dots on the visualizations. Our technique also gives an overview of the number of annotations associated with each aggregated dataset (*per R4 & R5*) in the visualization (e.g., each bar in a bar chart), using encodings such as position or size for the tags. Alternative representations for annotations on the view can include visualization-dependent mappings such as small bars along the x-axis in the barchart, piled circles for the scatterplots, or more structured views such as graphs based on annotation similarity[20].

Connecting grouped annotations to attributes. When an annotation is shown for a visualization element in a view, it cannot directly convey its connection to the data attributes in other views. The FacetNotes technique uses a connector plot to reveal this connection (*see R3*). In our dashboard, we used a starplot[32] to present the range of each data attribute connected to a grouped set of annotations. The length of the star's arm represents the size of the range of the corresponding attribute values associated. Interacting with each arm of the star shows the corresponding ranges. As shown in Figure 4, filtering operations change the range of attribute values, which are reflected by a change in the shape and size of the starplot (*per R4*). Alternative representations, such as parallel coordinate plots or span charts[33] can also be used (if space permits) to capture the footprint of annotations.

Interacting with annotations. FacetNotes provides controls for grouping and ordering the annotations. Furthermore, the connector plot is configurable to add, remove, or reorder data attributes.
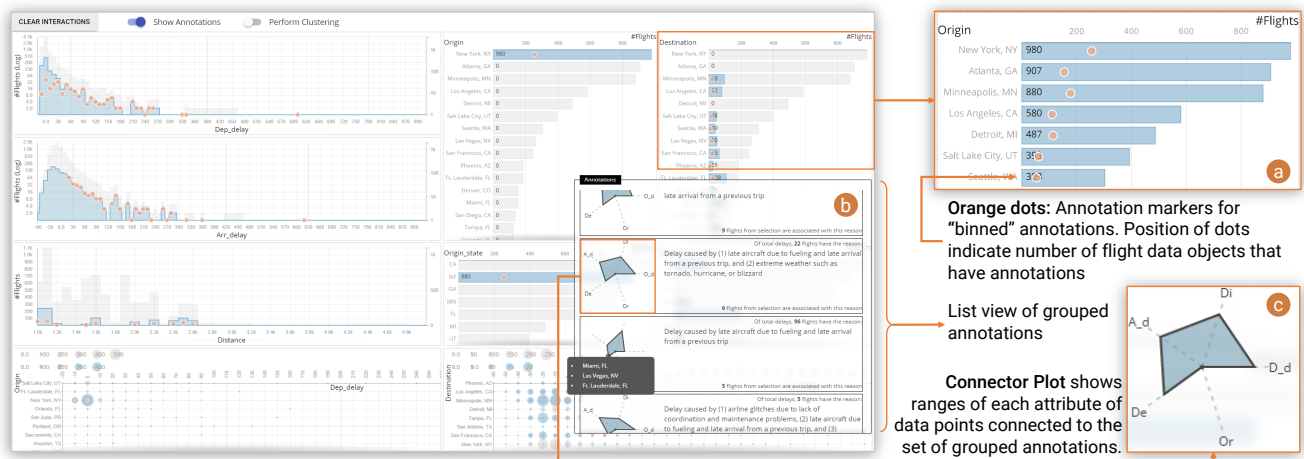
Figure 5. FacetNotes applied to a flight delay dashboard. Annotations adapt to existing views of bar, area, and scatterplots. Orange dots on each view are annotation markers (a), showing the presence of annotations associated with that set of data. Selecting an annotation marker pulls up a list view (b) within which annotations are grouped. Each annotation group within this list represents the range of associated data attributes in the form of a *Connector Plot* (c). A video demo of this dashboard can be accessed here: https://goo.gl/8wGZTy.

Operations in FacetNotes

FacetNotes attaches annotations to the visual representations following a series of operations to adapt the annotations to the visual design and data granularity. Figure 3 captures these operations and applies them to two different visual representations. These operations can be coupled with one another in a modular fashion. Overall, they can be tagged under three operation types: (1) **grouping** operations to bin annotations based on data attributes in the visualizations, (2) **clustering** based on similarities between one or more data/annotation attribute values, and extracting exemplars for each cluster, and (3) **ordering** operations to prioritize the annotations within the groups generated from the above operations. Here, we explain the design choices made for our dashboard (Figure 5) and the alternatives that exist for different operations.

Grouping based on target view. Annotations can be filtered to reflect the user's focus, and grouped based on the data attributes in the view (*per R1*). For doing so, we use standard binning techniques, grouping based on the values of the dimensions for aggregated views and based on the categorical combinations for granular views. In Figure 5, you can see the grouping based on a categorical dimension in part (a), while the binning based on numerical dimensions can be in the seen in action on the line charts. The grouping technique helps us represent the annotations on the views (as orange dots in Figure 5a).

The grouping itself provides a list of annotations for each data point in the chart. Within these lists, we use the annotation content itself to further consolidate the annotations as seen in Figure 5b. We use a naïve approach here based on keyword extraction and matching as a prototype. This leads to a compact list of annotations associated with each data point in any view in the dashboards. In a scenario with hundreds of annotations, this annotation grouping provides a tangible way to observe the annotations while interacting with any visualization in the dashboard. Advanced models incorporating semantic similarities and topic models can also be used to more intelligently group annotations together.

Ordering annotations within groups: The grouping technique provides a collection of annotations attached to each data point in every view in the dashboard. Ordering these lists of annotations will help us convey them in a meaningful way to the user. Note that the grouped annotations already have a data footprint attached to them as discussed earlier. For example, in Flights delays (Figure 5), an annotation about aircraft delayed due to fueling corresponds to a particular range of arrival delays and links to specific origin cities in the dataset. As you may imagine, this data footprint can form the basis for the ordering mechanisms.

In our dashboard, we explored two data-centered approaches for ordering, based on (1) the frequency (number of data points) of each annotation within the current group and frequency in the entire dataset, and (2) the ranges of attribute values of the data points attached to the annotation group and the overall ranges in the entire dataset. In the former approach, the annotation groups linked to most flights are surfaced on top. For example, for New York, annotations about "delays caused by airline glitches" appear on top since they cover the most flights coming from New York. In the latter approach, annotations are ordered by their spread in the data attributes. Expanding on the New York example, annotations about "delays caused by power outage" have the highest normalized range across the attributes and they appear on top. These ordering techniques convey two distinct information to the user with one conveying annotation frequency in terms of the data points and the other highlighting the scope of the annotation in terms of the underlying data distributions. For an analyst looking at this data, these mechanisms provide additional affordances to explore the annotations along with the structured data in the visualization dashboards. Beyond the above, alternate ordering strategies can also be

considered including distance-based and annotation-content-based ordering (similar to the grouping strategies) to further enhance the user's capabilities.

Clustering to extract data points and annotations: After grouping and ordering, we may still be left with a lot of annotations to convey in the visualization dashboard. In fact, in the Flight delays prototype in Figure 5, we observed annotations for each rectangle in the bar charts and each bin in the line chart. It can be overwhelming to the user. To overcome this challenge, highlighting "important" points (based on user definition and target scenario) can direct the users' attention to significant groups, and alternately, optimize the use of real estate on a display. While not currently implemented in our prototype, clustering methods including hierarchical clustering[34] or flat clustering (cf. k-means and DBSCAN[35]) can be used to combine annotations (*per R5*). Hierarchical clustering can also allow traversal through a cluster tree to pull up more clusters based on user interaction. Visually, this approach reduces the number of annotation tags as well as the annotation content attached to each tag (orange dots in Figure 5), making it easier for the user to go through them.

Implementation

Our dashboard implementation of FacetNotes uses web technologies—HTML5, CSS, JS—for the interface components, D3[36] for visualizations, and Leaflet.js for the geographical maps. It uses a Python server built with Flask for handling the operations within the technique include grouping, ordering, and clustering. These methods are developed using scikit-learn[37] and NLTK libraries[38]. The data is maintained in a MongoDB database accessed through the server. The server provides query APIs using MongoDB's query engine and helps us support fast, interactive visualizations on the web interface.

The visualization dashboard uses a simple grid layout for representing the data attributes and expands the space for specific attributes based on a pre-defined priority. The connection to MongoDB, the server-side components, and the use of aggregate representations scale very well to larger datasets. The entire codebase is available as open source for the interested reader.*

Application Example

Consider a data journalist trying to understand the reaction on Twitter following a presidential debate during an election to isolate the issues that the people care about. She plans to make sense of the tweet streams from Twitter along with information about the authors, shares, and likes. The first step is to extract structural data for visualization based on the goals of her analysis. In this case, she is interested in understanding the topics discussed, as well as their public sentiment and popularity among the Twitter users. For this reason, she extracts the keywords and topics (cf. tf-idf[39] and LDA[40]), sentiment values (cf. VADER[41]), number of likes, and usernames for tweets. This generates a tabular dataset with keywords, sentiments, likes, and user popularity as the columns, which can be used to construct a multidimensional visualization dashboard.

To take advantage of the FacetNotes technique, the dashboard system can treat the original tweets as the annotations from the data source. After all, these tweets are providing the complete context surrounding each structured data point shown in the dashboard. Note that FacetNotes is agnostic to the source as long as the annotations are connected to the data. FacetNotes technique now enhances the dashboard by (1) adding annotation markers on top of the dashboard views to give an overview of distribution of the annotations across the graphical elements in the views (per *R1*, similar to Figure 5), (2) showing an annotation view when one of the markers is selected to present annotations grouped based on their content similarity (keywords) and ordered based on their frequency (per *R2*), (3) allowing the user to open up multiple annotation views in the dashboard and compare the annotation content from different attribute perspectives (also per *R2*), and (4) showing the connector plot beside each annotation group to capture its associated range for the data attributes, i.e., the size of range as height of the arm in the plot (per *R3* & *R4*). These specific features are illustrated in Figure 5 and described in the previous section.

The journalist starts her exploration by checking each annotation marker on a popular keywords chart to find the Twitter users talking about "healthcare." She confirms this by going through a chart of the popular users and checking if their ordered annotation groups contain good number of tweets with keyword "healthcare." She then chooses to observe the policies these users are talking about and how they are being received by other Twitter users. She interacts with the dashboard to filter the charts to a particular user and observes that most of the tweets made by the user have a negative sentiment but a lot of likes. She opens up the annotations corresponding to this data point in the sentiment chart and observes the ordering of the annotation groups constructed using a bag-of-words similarity model. Looking at the content in the annotation groups, she notices that the negative tweets were about personal experiences arising from when the healthcare plan aided struggling individuals. She realizes the advantages of the specific healthcare policies from these annotation groups (per *R5*) and adds it to her story. She performs a similar analysis for the other topics such as "climate change" and "LGBT rights" to understand which candidate policies were better received by the Twitter users and their followers.

To expand on her story, she selects other popular keywords in the dataset and notices that they mostly have a positive sentiment. She selects them to observe which user groups always write positively about the selected keywords and learn about other topics that these users focus on, by exploring their tweets (present in the annotation groups). Overall, this exploration of structured data in the Twitter dataset and the textual tweet content in tandem is powerful as it allows our journalist to identify important patterns from the data while gaining insightful context from the textual annotations (in this case, the tweets themselves). She gains multiple perspectives into the data by this approach and builds an insightful story.

---

* https://github.com/karthikbadam/facetnotes

While this particular example uses the raw tweets as the annotations on the structured data points, annotations can come from other sources. The journalist could add her insights as annotations on the data points in the dashboard to help other journalists build on her story in the future. Alternatively, she could add hints about navigating the dataset as annotations to the visualization dashboard. Either way, FacetNotes can present these diverse annotations on the dashboard by considering their data footprint through the grouping and ordering mechanisms.

Evaluation

FacetNotes integrates annotations into interactive dashboards by grouping and ordering based on visual elements and the data attributes within target views. To understand the advantages of such an approach, We conducted a within-subjects study comparing our technique to a baseline interface to observe the affordances of the grouping/ordering operations and the connector plots during the sensemaking process. We used a flight delays dataset in both conditions along with annotations corresponding to the delay causes. For user tasks, we focused analyzing the cause of delays and effect on flight times for this study.

Participants

We recruited 9 paid participants (5 female, 4 male, aged 18–45) from the student population within our university campus. All participants self-reported as proficient computer users with 6+ years of experience. All participants had previously used visualization as a means of data analysis using MATLAB, R, SPSS, and Tableau platforms, and were thus considered visualization experts.

Dataset

We used a flight delays dataset (8,000 flight delays)[†] containing information about the departure and arrival delays of flights in the United States along with the origin, destination, and distance. Along with this information, this dataset provides delay causes such as air carrier delay, weather delay, or security delay. Using this dimension, we synthetically annotated the dataset with "reason for delay," adding detail to what was originally an attribute, for the purpose of the study. For example, if the cause was a "systems issue" then the annotation mentioned the delay was due to a "systems issue caused by a power outage at the operations center of the airport." The presence of annotations of the cause of delays along with the quantitative delays (effects) encourages a cause and effect analysis.

Apparatus and Interfaces

The study was conducted on a standard 22-inch monitor with 1920×1080 display resolution. We counterbalanced the order of two interfaces between participants:

- **FacetNotes Dashboard (FD):** This dashboard interface implements the FacetNotes technique to explore the annotations alongside the data within the Flights dataset (seen in Figure 5). It allows participants to observe the annotations across dimensional values,

open up the ordered annotations in place within the views, and observe their connections to the other dimensions through the connector plots. For instance, a user can click on an annotation marker corresponding to delay of 30 min within the departure delays chart. She can then see the causes of those delays (in annotations) grouped based on type (e.g., delay caused by weather). On top of that, the connector plot will show the value ranges with respect to other dimensions for the annotation group; say, arrival delays for the flights that departed late by 30 min due to bad weather.

- **Conventional Dashboard (CD):** To serve as a baseline, we developed a standard annotation dashboard placing annotations in a separate view connected to the data (similar to previous work[3]). This baseline interface *lacks* (1) the overviews created by the grouping operations and (2) the connector plot for the data context of the annotations. The baseline interface ensured that all other factors in the dashboard visualizations remain the same, isolating the influence of FacetNotes technique.

Tasks and Protocol

In the dataset, the annotations explained why specific flights were delayed. While the cause of delays are captured in the annotations, the effects of delays are represented in the visualizations in the dashboard. We there formulated investigative tasks that required understanding the cause and effect of delays since these are the main questions raised by the dataset. In particular, the participants performed two types of tasks:

- **Controlled task**: In this task, the participant is given questions to answer by identifying target visualizations and performing few interactions. For creating the controlled tasks, we sought help from two visualization experts. They used the interface to explore the data by filtering views to find interesting observations and formulated questions about the dataset based on the observations (as done by Sarvghad et al.[42]). We picked questions that require one or two filters—applied through selections in views. Participants answered four questions on each interface. Example questions include,

  - If arrival to New York city is delayed by at least 60 mins, what could be the most probable cause?

  - What cause of delay affects the most destinations if you are traveling more than 2000 miles from Los Angeles?

  - How long can be the arrival delay to Detroit from 1500-2000 miles away if there are airline system issues?

  - For flights traveling more than 3000 miles, what can be the worst arrival delay at San Francisco due to airline glitches?

[†]Dataset: https://www.kaggle.com/usdot/flight-delays

- **Open-ended task:** The participant was given responsibility over a region in the United States and asked to develop as many observations as possible to isolate causes of delays from the annotations and predict their effects. This task required them to develop goals, generate hypotheses, make observations, and create insights from the dataset, resembling a sensemaking process.

Participants performed both tasks on one interface and answered related questions before moving to the next interface. Their answers and observations were audio recorded during the session along with a screen recording of the interface, and analyzed posthoc to extract interaction and user observation patterns.

Procedure

Participants completed a demographics survey and were then introduced to the assigned interface. They first went through a presentation (<5 slides) explaining the interface features, and answered sample questions to ensure that they understood how to use the interface. The training presentation was repeated and demonstrations were given if needed. Following this, the participants went to answering the controlled tasks first and then spent up to 15 minutes on the open-ended task. We employed a think-aloud protocol for both tasks. They then repeated the entire process with the next interface. They were audio-recorded and their interface was screen-recorded during the session. At the end of session, participants provided feedback on the ease of use, efficiency, and enjoyability of the interfaces. Each session lasted less than one hour.

Results

We separated the interaction patterns of our participants based on the controlled tasks and the open-ended exploration tasks. In this section, we report the results from these tasks.

Controlled Tasks

The questions given to the participants were designed to identify possible causes and/or effects of delay(s). Participants explored the data and the annotations to extract the main cause or effect from a group of candidates. When using FacetNotes, participants followed three strategies to develop their answers:

S1: Focusing on top annotation groups. Three participants (P1, P6, P7) answered all their questions by just focusing on the overall top causes and effects. Given a question, say, "find probable cause(s) for departure delays between 30 and 60 mins," they would apply the 30–60 min delay filter on the interface, open the connector plot on a view (say, a bar chart of Flight origins), and pick the cause corresponding to the largest/most frequent annotation group (e.g. "delays caused by system issues"). While this identifies a probable cause, they do not check for other probable causes. For instance, there could be other causes that specifically only result in 30–60 min delays, or even 30–40 mins, which lie within the given range.

S2: Combining multiple annotation groups. Four participants (P2, P3, P4, P8) came up with answers that were "safe": they would isolate all possible causes for a particular range of delay and then combine them to arrive at an answer that is less precise. This required them to explore the annotation groups on the target view and verify the effects of each annotation using the connector plot and list out all the causes that could possibly lead to the given delay (not just the most likely cause). They followed a similar strategy for questions centered on the effects. In either case, they viewed the multiple annotations and combined them together visually based on the connector plot.

S3: Exploring annotations across multiple views. Some participants (P2, P5, P9) took full advantage of the FacetNotes technique to integrate annotations. They would first identify a list of possible causes (or effects based on the question) on a target view and then extract the important ones by viewing the annotations on other views. For example, to isolate causes of departure delays of one hour at New York, they would go through different annotations near 60-min mark on the delay chart to develop potential answers. They would then go through the origin chart to check the distribution of each particular cause by clicking the annotation mark on top of New York. By exploring annotations from multiple views, these participants were able to develop precise answers to the questions. They attributed their strategy to the availability of the annotations markers directly on top of the views.

These strategies were hard to perform on the conventional dashboard (baseline) since it lacked the connector plot and required the participants to interact with the dashboard to manually connect multiple annotations to the data views. As this was tedious, participants neglected few evidences and only focused either on most frequent annotations (e.g., weather issues and late aircraft issues) or on most frequent effects (typically, < 10-min). Answers given by all participants using the conventional dashboard were thus less precise. They were similar to answers from S1 where the FacetNotes technique was least used.

Open-ended Task

The open-ended task used a scenario where participants were responsible for the operations from a place (e.g., New York or Georgia in United States). They were asked develop insights about the causes and effects on flight times by freely exploring the data through interactive filtering. We noticed three levels of visual exploration:

Minimal exploration. Three participants (P6, P7, P9) performed minimal selections on the dashboard when using FacetNotes. They focused on the "flight origin" view, examined all the annotations and identified the causes for different levels of delays from the connector plots. This focus on just one chart limited their perspective.

Moderate exploration. Two participants (P1, P8) explored more than two views in the dashboard during their exploration. They identified delay causes (from the annotations) for different destinations in the East and West coasts. They further selected specific ranges of delays to observe the association between the causes/effects and the flight origins/destinations. For example, which airports are

affected by shorter delays or which airports are affected by specific causes of delays.

Extended exploration. Four participants (P2, P3, P4, P5) fully explored all the views in the FacetNotes dashboard to go from overview to detailed observations. They started by analyzing all the annotations attached to their target location on the origin chart. Based on the major causes and effects (e.g., large delays, most airports, and distances affected), they tried to verify if these trends repeat in specific regions in the US and distance of flights. Finally, they explored the delay charts (departure and arrival delays) by selecting specific delay ranges (e.g., < 30min, > 120min) to check for causes responsible for specific effects across locations. Here, too, the observations from this strategy are facilitated mainly by the grouping and ordering in FacetNotes.

In the conventional dashboard, six participants (P1–P5, P8) made fewer observations. While they explored similar number of views as in the FacetNotes condition, their observations were less detailed. They only covered the top causes and did not make nuanced observations about how each cause affected the flights across the United States. Their answers resembled the first strategy from the controlled task. Participants commented that it was hard to compare the distributions of annotations across the views in the conventional dashboard.

Subjective Ratings

Participants rated each interface on three metrics— efficiency, ease of use, and enjoyability—on a 5-point Likert scale ranging from 1 (strongly disagree) to 5 (strongly agree). These questions were presented as, "Is this interface efficient for answering the questions?" Their responses show that the FacetNotes dashboard was received more positively on all the scales (Fig. 6). FacetNotes was also preferred over the baseline by 8/9 participants. Participants associated these ratings to various reasons linked the features with the FacetNotes technique. P3 said, "*The annotation part [for FacetNotes] was more conclusive. It was easier to analyze data and answer the questions. It made the textual data kind of interactive which was good.*" Referring to the connector plots, P4 said, "*This one gives multidimensional information from a single point of the data.*" P1 who preferred the conventional dashboard said, "*If I had more experience of using this interface, I will go with FacetNotes because I believe I can learn much diverse insights from it. But since I don't have much experience with this interface, I felt easier to interact with [the conventional dashboard].*"
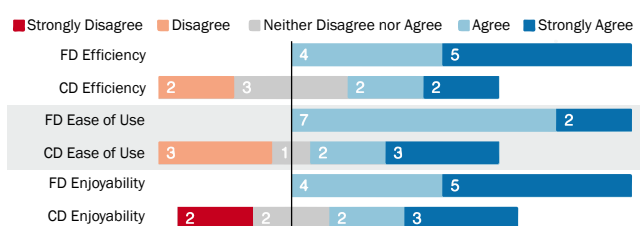

Figure 6. Participant responses on a 5-point Likert scale shows that the FacetNotes dashboard (FD) performed consistently better than the conventional dashboard (CD).

Discussion

Our study revealed the strategies used to answer specific questions for the controlled tasks and general observations for the open tasks. We observed some commonalities in the strategies participants used in both kinds of tasks. One pattern of exploration among the participants was the parallel between the extended exploration from open-ended tasks and exploration across multiple views (S3) from the controlled tasks. In both cases, the participants explored the dashboard with multiple filters through the following steps: (1) viewing overall dashboard and annotations, (2) observing the data connected to each annotation through the connector plot, (3) developing high-level observations based on the connector plots, and (4) filtering data on charts and identifying specific trends. Following this the users went back to the original interface without filters (to explore new threads of annotations). These steps allowed the users to explore the visualized data in the dashboard and the annotations in tandem with one guiding the other. This approach was also exemplified in the application example. This workflow is facilitated by FacetNotes through the grouping and ordering operations and the connector plots, and was not possible with the conventional dashboard.

The above observation from the study hints at the advantage of FacetNotes compared to past approaches. In Sense.us,[3] the annotations appear in a separate panel similar to the conventional dashboard. This allows them to capture many annotations in one place. In Contextifier,[4] the annotations are attached to the data points in the visualization, but the representation does not scale up when many annotations are present or even when multiple views of the dataset are available. In contrast to both approaches, FacetNotes brought together the best of both worlds by using the grouping and ordering mechanisms to attach the annotations to the data. Beyond the conceptual advantages, it had an impact on the cognitive process of the user and led to more complex insights from extended exploration. In our study, the technique unlocked the sensemaking process and enabled our participants make complex observations from the flights dataset by exploring the annotations across multiple views of the data.

There were a few participants who only performed minimal exploration in open-ended tasks, or focused only on the top causes/effects in the controlled tasks in both interfaces. These participants showed a common pattern as well: they either arrived at very broad answers or did not explore more nuanced details across various views. This was bound to happen: our goal was to capture the participants' interest in exploring the data given a subject (in this case, flight delays). Not all participants can be expected to express equal interest in the subject.

Limitations and Future Work

Annotation presentation: The focus of this work has been on annotation presentation rather than creation. As noted earlier, it is challenging to show annotations across multidimensional dashboards. The FacetNotes technique supports users to explore annotations along with the data in a blended manner during sensemaking, which is promising

as seen in the user study. Future work should investigate how creation of annotations can integrate into this workflow.

Annotation content: We alluded to the fact that annotations can come multiple sources. They can be added by an analyst looking at the structured data in the visualization dashboard or they could come from observing the real world processes surrounding the dataset. The commonality across all annotations is their connection to the data. Irrespective of the source, this association can help us guide how we show the annotations. This association has been the focus of FacetNotes. While having this concrete focus, we overlooked some of the other common aspects of annotations; particularly, their metadata—who created them and when they were created. A more general treatment of annotations should consider those aspects as well, but it is unfortunately outside the scope of this paper. For future work, we guide readers to consider such additional information to further enhance FacetNotes.

Connector plot design: The connector plot was widely used by the participants of our study to go through annotations and data values, since it connects the structured and unstructured parts of the dataset. However, it only shows range of values (as area) on each dimension within its starplot design. It therefore loses the distribution information. For this reason, a better design for the connector plot is needed, which not only gives the range of connected attributes values but also the distributions. This may require a re-imagining of the connector plot to one that is more expressive while remaining compact. This will be a part of our future work.

Ordering of annotations: The ordering mechanism within the annotations view for our user study was one dimensional, based on the frequency of the annotations within the current data filters. This required the participants to explore each group of annotations in the annotation view to identify causes/effects of interest. In future work, we need to consider alternate ordering mechanisms that use semantic distances between the annotations texts and the spatial distances within the data connected to them, to order annotations from multiple perspectives. These alternative orderings can highlight anomalies, common patterns, or statistical significance of their association.

Study coverage: Our user study has also not explored clustering mechanisms that could control the number of shown annotations. By showing only the annotations representing the clusters, this approach can prevent the overloading of views with annotations. Future studies should focus on how such a feature would be useful to the explore the annotations. Finally, the grouping of annotations based on its text content is by itself a challenge, since complex strategies (beyond bag-of-words similarity models) might need to be explored to figure out the best groupings.

Study design: Finally, our current evaluation showcases an initial step towards understanding the effects of tightly integrating annotations within multidimensional visual dashboards. Our qualitative results from 9 participants present interaction and visual exploration patterns that provide evidence of the effectiveness of FacetNotes. However, more evaluations need to be performed with other visual analysis tasks, not just cause and effect analysis from our user study, to understand the efficacy of our technique.

Multi-user analytics: Annotations are often found in collaborative analytic processes.[14] We have not explicitly considered such processes in our evaluation, but it is a natural extension to evaluate the advantages of FacetNotes in future.

Implications

We found that the main implication of the FacetNotes technique was the blend between the exploration driven by the data visualized in the dashboard, and the new questions and insights created by the shown annotations. While we focused on textual annotations on top of multidimensional visualizations, such a technique can represent other information. For example, instead of annotations, we could present the exploration history on top of the visualizations using FacetNotes. Imagine when you pick New York city in the Flights dashboard, you could see who filtered New York in the past, when, and what else they chose next. The connector plot and the additional representations are well suited to present this provenance[11] for self reflection or remote collaboration. To convey such implications, we discuss below when you should use FacetNotes along with more promising applications.

When to use FacetNotes. Our new technique helped the user switch between data-centric and annotation-centric exploration modes, where they would interact with the data to pull up the annotations, and interact with the annotations on other views to understand their connected data. The presence of annotations on each view in the dashboard provided them quick access to verify the data trends and anomalies across all the charts. In enabling such a workflow, we inherently assumed that the data and the annotations provide meaningful information to the user to explore both at the same time. In fact, our application example and the user study capture situations where knowledge is hidden in structured data as well as the textual annotations. Exploring both together helps the users unlock more than looking at them separately. However, this may not always be the case. There are scenarios where there can be an imbalance between the two sources of information (i.e., structured data and textual annotation). For example, due to lack of enough annotations to help explore the data, or presence of very few dimensions in the multidimensional data. In such scenarios, it is not essential to have the complete version of the FacetNotes technique. We recommend visualization designers to pick and choose parts of FacetNotes—grouping, ordering, or clustering of annotations—best suited for the target scenario.

For remote collaboration. Beyond our envisioned application example, we foresee applications for FacetNotes in remote collaborations where annotations are created by many users to capture their process, knowledge, and insights. While we made conceptual comparisons to ManyEyes[15] and Sense.us[3] in earlier sections, we have not considered the annotation creation process. Therefore we cannot provide concrete suggestions for how FacetNotes technique should be integrated into such tools. However, our findings are promising and they convey the advantages of FacetNotes. We can infer that grouping and ordering annotations can be helpful in such scenarios to build on the knowledge of other analysts. In collaborative analytics,[14,19,22] it is also important to represent user activity and provenance in the

visualization interface along with the annotations. Recent work in asynchronous collaborative visual analytics includes techniques to incorporate provenance of insights in the form of visualizations and annotations.[43] Prior work has also represented user activity in synchronous collaborations to convey the presence and activity of the users through additional representations on top of the dashboards.[22,42] FacetNotes will integrate well with such techniques by presenting the annotations based on the user activity (e.g., user is focused on flights from New York).

For intelligence analysis. The field of visual analytics[2] has built great applications for intelligence analysis, where heterogeneous data in the form of textual, quantitative, and pictorial formats needs to analyzed to make decisions in social contexts. More broadly, FacetNotes can shine in such scenarios since it primarily combines structured and unstructured data into one visualization dashboard. It can enable analysts take advantage of their perceptual abilities to find relevant annotations and cognitive abilities to connect the annotations back to the visualizations. We look forward to developing applications in this space using FacetNotes.

Conclusion

In this paper, we have introduced the FacetNotes technique for presenting textual annotations on visual dashboards that contain multiple coordinated visualizations. FacetNotes uses grouping methods based on the elements of each visualization as well as the textual content of the annotation. This enables exploration of many annotations connected to the graphical elements of the views. It also uses ordering strategies to sort the annotation groups based on their frequency and other metrics. We applied this technique to a flight delays dashboard and conducted a user study to understand the strategies used by the participants when working with the technique to answer specific questions and also perform an open-ended visual exploration. Results revealed that the technique helped the participants easily blend data exploration with annotation exploration, and make precise observations across multiple visualizations driven by the annotations in the dashboard.

References

1. John Wilder Tukey. *Exploratory Data Analysis*. Addison-Wesley Series in Behavioral Science: Quantitative Methods. Addison-Wesley, Reading, MA, USA, 1977.

2. Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Mélancon. Visual analytics: Definition, process and challenges. In *Information Visualization – Human-Centered Issues and Perspectives*, number 4950 in LNCS, pages 154–175. Springer, Aug 2008. doi: 10.1007/978-3-540-70956-5_7.

3. Jeffrey Heer, Fernanda B. Viégas, and Martin Wattenberg. Voyagers and voyeurs: supporting asynchronous collaborative information visualization. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 1029–1038, New York, NY, USA, 2007. ACM. doi: 10.1145/1435417.1435439.

4. Jessica Hullman, Nicholas Diakopoulos, and Eytan Adar. Contextifier: automatic generation of annotated stock visualizations. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 2707–2716, New York, NY, USA, 2013. ACM. doi: 10.1145/2470654.2481374.

5. Edward Segel and Jeffrey Heer. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1139–1148, 2010. doi: 10.1109/TVCG.2010.179.

6. Jonathan C Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Proceedings of the IEEE International Conference on Coordinated and Multiple Views in Exploratory Visualization*, pages 61–71, Piscataway, NJ, USA, 2007. IEEE. doi: 10.1109/CMV.2007.20.

7. Alper Sarikaya, Michael Correll, Lyn Bartram, Melanie Tory, and Danyel Fisher. What do we talk about when we talk about dashboards? *IEEE Transactions on Visualization and Computer Graphics*, 25(1):682–692, 2019. doi: 10.1109/TVCG.2018.2864903.

8. Yang Chen and Jing Yang. ManyInsights: A visual analytics approach to supporting effective insight management. *Tsinghua Science and Technology*, 18(2):171–181, 2013. doi: 10.1109/TST.2013.6509100.

9. Peter Pirolli and Stuart Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of international conference on intelligence analysis*, volume 5, pages 2–4, 2005.

10. Robert Kosara and Jock Mackinlay. Storytelling: The next step for visualization. *Computer*, 46(5):44–50, 2013. doi: 10.1109/MC.2013.36.

11. Dennis P. Groth and Kristy Streefkerk. Provenance and annotation for visual exploration systems. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1500–1510, 2006. doi: 10.1109/TVCG.2006.101.

12. Jeffrey Heer and Ben Shneiderman. Interactive dynamics for visual analysis. *Queue*, 10(2):30, 2012.

13. Jeffrey Heer and Maneesh Agrawala. Design considerations for collaborative visual analytics. *Information Visualization*, 7(1):49–62, 2008. doi: 10.1109/VAST.2007.4389011.

14. Petra Isenberg, Niklas Elmqvist, Jean Scholtz, Daniel Cernea, Kwan-Liu Ma, and Hans Hagen. Collaborative visualization: definition, challenges, and research agenda. *Information Visualization*, 10(4):310–326, 2011. doi: 10.1177/1473871611412817.

15. Fernanda B Viégas, Martin Wattenberg, Frank Van Ham, Jesse Kriss, and Matt McKeon. ManyEyes: A site for visualization at internet scale. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1121–1128, 2007. doi: 10.1109/TVCG.2007.70577.

16. Donghao Ren, Matthew Brehmer, Bongshin Lee, Tobias Hollerer, and Eun Kyoung Choe. ChartAccent: Annotation for data-driven storytelling. In *Proceedings of the IEEE Pacific Visualization Symposium*, April 2017. doi: 10.1109/PACIFICVIS.2017.8031599.

17. Yuet Ling Wong, Jieqiong Zhao, and Niklas Elmqvist. Evaluating social navigation visualization in online geographic maps. *International Journal of Human-Computer Interaction*, 31(2):118–127, 2015. doi: 10.1080/10447318.2014.959106.

18. Youn-ah Kang and John Stasko. Characterizing the intelligence analysis process through a longitudinal field study: Implications for visual analytics. *Information Visualization*, 13 (2):134–158, 2014. doi: 10.1109/VAST.2011.6102438.

19. Narges Mahyar, Ali Sarvghad, and Melanie Tory. A closer look at note taking in the co-located collaborative visual analytics process. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*, pages 171–178, Piscataway, NJ, USA, 2010. IEEE. doi: 10.1109/VAST.2010.5652879.

20. Jian Zhao, Michael Glueck, Simon Breslav, Fanny Chevalier, and Azam Khan. 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):261–270, 2017. doi: 10.1109/ TVCG.2016.2598543.

21. Mehmet Adil Yalcin, Niklas Elmqvist, and Benjamin B Bederson. Keshif: Rapid and expressive tabular data exploration for novices. *IEEE Transactions on Visualization and Computer Graphics*, 2017. doi: 10.1109/TVCG.2017. 2723393.

22. Sriram Karthik Badam, Zehua Zeng, Emily Wall, Alex Endert, and Niklas Elmqvist. Supporting team-first visual analytics through group activity representations. In *Proceedings of Graphics Interface*, pages 208–213, 2017.

23. Micheline Elias and Anastasia Bezerianos. Annotating BI visualization dashboards: Needs & challenges. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 1641–1650, New York, NY, USA, 2012. ACM. doi: 10.1145/2207676.2208288.

24. Yedendra Babu Shrinivasan and Jarke J van Wijk. Supporting the analytical reasoning process in information visualization. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 1237–1246, New York, NY, USA, 2008. ACM. doi: 10.1145/1357054.1357247.

25. Emanuel Zgraggen, Robert Zeleznik, and Steven M Drucker. Panoramicdata: Data analysis through pen & touch. *IEEE Transactions on Visualization and Computer Graphics*, 20(12): 2112–2121, 2014. doi: 10.1109/TVCG.2014.2346293.

26. Jeffrey Heer. The design of Sense.us. In Toby Segaran and Jeff Hammerbacher, editors, *Beautiful Data: The Stories Behind Elegant Data Solutions*, pages 183–204. O'Reilly Media, Inc., Sebastopol, CA, USA, 2009.

27. Wikimapia. https://wikimapia.org/. Accessed: 2017-12-07.

28. Yue Wang, James Shearer, and Kwan-Liu Ma. VICA: a Voronoi interface for visualizing collaborative annotations. *Cooperative Design, Visualization, and Engineering*, pages 21–32, 2007.

29. François Chevenet, Christine Brun, Anne-Laure Bañuls, Bernard Jacq, and Richard Christen. TreeDyn: towards dynamic graphics and annotations for analyses of trees.

BMC Bioinformatics, 7(1):439, 2006. doi: 10.1186/ 1471-2105-7-439.

30. Eser Kandogan. Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*, pages 73–82, Piscataway, NJ, USA, 2012. IEEE. doi: 10.1109/VAST.2012.6400487.

31. Kostiantyn Kucher, Andreas Kerren, Carita Paradis, and Magnus Sahlgren. Visual analysis of text annotations for stance classification with ALVA. In *Poster Abstracts of the EuroGraphics/VGTC Conference on Visualization*, pages 49–51, 2016.

32. Michael Friendly. Statistical graphics for multivariate data. *SAS SUGI*, 16:1157–1162, 1991.

33. Span chart. https://datavizproject.com/ data-type/span-chart/. Accessed: 2021-12-25.

34. Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010. doi: 10. 1016/j.patrec.2009.09.011.

35. Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*, pages 226–231, 1996.

36. Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011. doi: 10. 1109/TVCG.2011.185.

37. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

38. Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly Media, Inc., Sebastopol, CA, USA, 1st edition, 2009.

39. Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972. doi: 10.1108/eb026526.

40. David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

41. Clayton J. Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the AAAI Conference on Weblogs and Social Media*, pages 216–225, 2014.

42. Ali Sarvghad and Melanie Tory. Exploiting analysis history to support collaborative data analysis. In *Proceedings of the Graphics Interface Conference*, pages 123–130, 2015.

43. Jianping Kelvin Li, Shenyu Xu, Yecong Ye, and Kwan-Liu Ma. Resolving conflicting insights in asynchronous collaborative visual analysis. *Computer Graphics Forum*, 39(3):497–509, 2020. doi: 10.1111/cgf.13997.