ChartAccent: Annotation for Data-Driven Storytelling

Donghao Ren*
University of California,
Santa Barbara

Matthew Brehmer†
Microsoft Research

Bongshin Lee‡
Microsoft Research

Tobias Höllerer§
University of California,
Santa Barbara

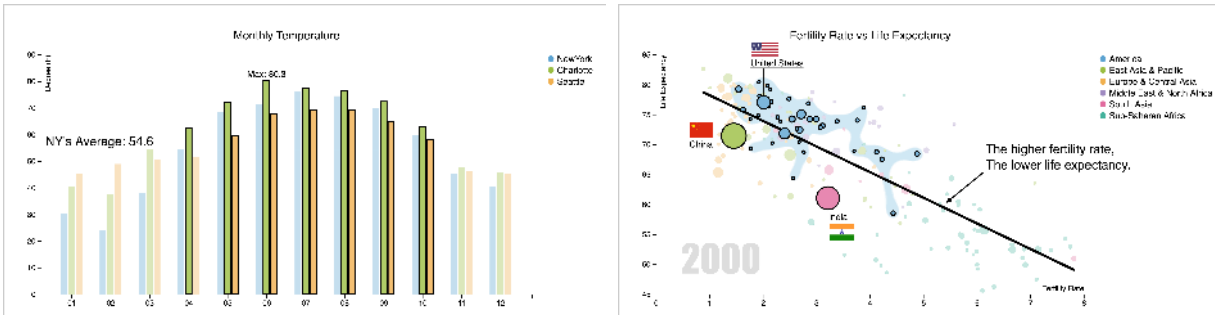Eun Kyoung Choe¶
Pennsylvania
State University

Figure 1: Examples of annotated charts created with ChartAccent: (left) emphasizing the months when Charlotte and Seattle's temperatures are higher than New York's average; (right) the relationship between fertility rate and life expectancy, with text and image annotations for the United States, China, and India; countries from North and South America are highlighted in blue.

ABSTRACT

Annotation plays an important role in conveying key points in visual data-driven storytelling; it helps presenters explain and emphasize core messages and specific data. However, the visualization research community has a limited understanding of annotation and its role in data-driven storytelling, and existing charting software provides limited support for creating annotations. In this paper, we characterize a design space of chart annotations, one informed by a survey of 106 annotated charts published by six prominent news graphics desks. Using this design space, we designed and developed ChartAccent, a tool that allows people to quickly and easily augment charts via a palette of annotation interactions that generate manual and data-driven annotations. We also report on a study in which participants reproduced a series of annotated charts using ChartAccent, beginning with unadorned versions of the same charts. Finally, we discuss the lessons learned during the process of designing and evaluating ChartAccent, and suggest directions for future research.

Index Terms: H.5.2 [Information Interfaces and Presentation (e.g., HCI)]: User Interfaces—Graphical user interfaces (GUI).

1 INTRODUCTION

Annotation is an essential part of visual data-driven storytelling [29]. *The New York Times* graphics editor Amanda Cox once stated that *"the annotation layer is the most important thing we do...otherwise it's a case of here it is, you go figure it out"* [28]. For example, annotations in an interactive slideshow convey a narrative, providing explanations that viewers would be unlikely to identify on their own [40]. In addition to helping presenters explain core messages or specific data, annotations enable them to

*e-mail: donghaoren@cs.ucsb.edu
†e-mail: mabrehme@microsoft.com
‡e-mail: bongshin@microsoft.com
§e-mail: holl@cs.ucsb.edu
¶e-mail: echoe@ist.psu.edu

emphasize and draw viewers' attention to specific parts of the chart [31]. Furthermore, appropriate annotations can help presenters provide additional context, potentially facilitate the memorability of a chart [7], and increase the aesthetic appeal of a chart.

However, commercial charting software such as Excel or Tableau provides limited support for creating annotations; similarly, current business intelligence tools that incorporate visualization provide little annotation support beyond simple text annotations [16]. Thus, to create an envisioned set of annotations, people often export charts to presentation authoring tools or graphic illustration tools [4, 5], where they can add individual annotations to a chart, albeit at the cost of being disconnected from the underlying data. For example, personal data presentations by "quantified selfers" often contain annotated charts to make key points more salient, interpretable, and enjoyable [14], but these annotations are often added manually. Considering its high importance and utility in storytelling, we argue that the visualization community requires a better understanding of the annotation design space: the forms of annotation, what purposes these forms serve, and how these forms of annotation can be applied to elements in a chart via user-driven and data-driven approaches.

In this work, we introduce ways to help presenters quickly and easily augment their charts with various annotations. We characterize a design space of chart annotations, drawing from a survey of 106 annotated charts published by six prominent news graphics desks including *The Economist* and *The New York Times*. We also discuss the reasons and goals of annotating charts, and what does and does not constitute annotation in this context. Informed by the survey and design space, we designed and developed ChartAccent, a proof-of-concept tool that provides a palette of annotation interactions that generate manual and data-driven annotations. We report on a reproduction study, in which we aimed to evaluate the experience of annotating charts with ChartAccent. After a short tutorial and practice session, most participants could easily annotate line graphs, bar charts, and scatterplots with ChartAccent, reproducing a series of previously annotated charts. Furthermore, they enjoyed creating annotation with ChartAccent, and expressed a strong desire to use ChartAccent for future presentations. We conclude with a discussion on the lessons learned from the design and evaluation of ChartAccent, and we indicate future research directions.

The contribution of this paper is threefold: (i) a reflection on chart annotation in the context of data-driven storytelling; (ii) design dimensions for chart annotation based on a survey of 106 annotated charts published by prominent news graphics desks; and (iii) the design, development, and evaluation of ChartAccent, which is now available for use at `https://chartaccent.github.io`.

2 BACKGROUND & RELATED WORK

We review previous work that discusses the role of annotation in visual data-driven storytelling and visualization, as well as techniques and tools to support annotation design.

2.1 Annotation in Data-Driven Storytelling

Visual data-driven storytelling often involves the use of visualization techniques to support or complement a written or spoken narrative. Segel and Heer [40] refer to this form of storytelling as *narrative visualization*, and in their proposed design space of narrative visualization, they indicate the importance of textual and graphical annotation as well as visual highlighting. Kosara and Mackinlay [29] have also emphasized the importance of annotation and highlighting in visual data-driven storytelling, particularly in the context of live presentations. In an effort to scope visual data stories, Lee et al. [30] emphasize the need of the intended message in stories and the role of written explanations or annotations that help the viewer capture the message. They also identify that, to make it easier to tell data-driven stories, it is useful to support the easy creation of custom annotations through direct manipulation and reuse of existing story elements. Stolper et al. [41] identify seven common annotation techniques used to communicate narrative and explain data in popular data-driven stories. In addition, Choe et al. [14] provide empirical evidence of the important role of annotation through an analysis of the design choices used in live presentations by self-tracking enthusiasts or "quantified selfers." Hullman and Diakopoulos [26] consider annotation to be one of the four editorial layers in this genre of storytelling, alongside the layers of data, visual representation, and interactivity; and while there are certainly a number of tools and techniques that address these other layers, support for the annotation layer remains underdeveloped. Our aim is to enumerate and realize the design choices in the annotation layer of visual data-driven storytelling.

2.2 Current Approaches to Annotation

Many contemporary visualization tools provide limited annotation support, particularly in the domain of business intelligence [16]. Annotation functionality also features prominently in tools intended for collaborative visual analysis, such as ManyEyes [44], sense.us [25], and CommentSpace [46], as the ability to annotate a chart allows people to share their insights with others. In the following, we give further consideration to tools that support data-driven or data-aware annotation, and to annotation support in interactive visualization authoring tools.

Data-aware annotation: In the context of visualization, annotation can mean more than the mere addition of textual and graphical elements to an existing chart. Annotation functionality can be implemented in such a way that annotations are aware of the data being visualized. Heer and Shneiderman's refer to "data-aware annotations" [24], or graphical and textual elements that appear in response to the interactive selection and brushing of visual marks that correspond to data [23]. This concept is realized in Click2Annotate [12] and Touch2Annotate [13], in which the semi-automated annotation of data items is facilitated via simple direct manipulation interactions. In a prototype tool by Heer et al. [23], a set of annotations for data items can appear in response to query criteria as well as selection and brushing interactions, and their tool decides which annotations to show based upon a relevance ranking.

Kandogan [27] introduced the concept of *just-in-time annotation*, in which a chart is automatically annotated as a person interacts with it. Similarly, Bryan et al. [10] introduced an approach for automatically annotating charts depicting time-oriented data during interactive exploration based on visual salience and significant features in the data. Altogether, these data-driven approaches certainly eliminate the tedium of manually annotating data items in a chart, though it could also be argued that these approaches take too much control away from the person using the system: they neither provide full control over which data items are annotated, nor do they provide control over the form or content of these annotations. Moreover, these approaches are predominantly geared toward data analysis tasks, and while many annotations in the context of visualization do correspond with specific data-bound graphical elements, every annotation in a chart should not be required to directly reference the underlying data. Our work, on the other hand, focuses more on the task of augmenting a chart for communication, combining both data-driven and manual annotation. With ChartAccent, we provide more freedom in terms of deciding which items get annotated and in terms of the form and content of these annotations, leveraging a data-driven approach when needed. Finally, while data-aware annotations that are triggered by interactive selection certainly fall under our purview, we do not wish to downplay the importance of static (i.e., non-interactive) forms of annotation when presenting a chart to an audience. This view is reflected by *The New York Times'* deputy graphics director Archie Tse, who recently argued that the best form of visual storytelling is often *static* [34].

Interactive visualization authoring tools: Many recent interactive visualization authoring environments or visualization construction interfaces [20, 37] attempt to balance simplicity with an expressive range of visual encoding choices. However, they generally lack a support for annotation—especially data-driven annotation. With general-purpose chart authoring tools such as those offered in Microsoft Excel, chart authors are limited to selecting individual items or data series and modifying their visual encoding; data-driven operations such as selecting and annotating data items within a particular value range, for example, are not supported. As a result, an author will typically create a chart with a tool such as Excel and subsequently export it to a graphical editing tool such as Adobe Creative Suite to manually add annotations [4, 5], and in doing so awareness of the underlying data is lost.

Recent interactive visualization authoring tools such as Lyra [39] and iVisDesigner [36] are more expressive than general-purpose tools like Excel in terms of visual encoding design choices. Although authors can create data-driven annotations with these tools, their learning curve is significant, and the annotation process is particularly tedious. We also draw inspiration from Mr. Chartmaker [1], the interactive charting tool used internally at *The New York Times*, which provides some data-driven annotation support via direct manipulation and selection of individual data items; we build upon these capabilities in ChartAccent by introducing data-driven selections in addition to annotation via manual selection.

Tableau Desktop [42] also provides several options for annotating charts, some of which overlap with ChartAccent. For example, Tableau allows a person to add trend lines to a chart, and text annotations of marks in Tableau can be data-driven, referencing marks' underlying data attributes. However, ChartAccent goes beyond Tableau in various aspects. In Tableau, three annotation targets are explicitly denoted when interacting with a chart: a x,y *point*, a rectangular *area*, or any selected *mark(s)*. With ChartAccent, we consider a wider set of annotation targets informed by our survey of 106 annotated charts, including both one- and two-dimensional ranges and other targets listed in Section 3.2. In addition, highlighting marks in ChartAccent is simpler and more flexible than Tableau, without requiring explicit set creation; selected marks can be highlighted directly without affecting unselected marks.

Annotation via programming: It is certainly possible to author a chart with precise control over annotation using visualization programming libraries and packages such as ggplot2 [45] or D3.js [9], or using libraries specific to data-driven annotation and labelling such as swoopyDrag.js [35] or labella.js [47], or by using the Hanpuku D3-Illustrator bridge tool [5]. With ggplot2 in particular, chart authors can add sophisticated data-driven annotations such as trend lines or annotations relying upon more complex statistical models. However, although these approaches offer precise control, they require programming skills and can only provide asynchronous feedback to the author; with ChartAccent, we opt for an interactive visualization authoring environment that requires no specialized programming knowledge and provides instant feedback to the author.

3 ANNOTATION DESIGN SPACE

Existing definitions of "annotation" and the act of "annotating" relate to the altering of an existing object by adding a note or comment to it. Unfortunately, there is currently limited treatment of annotation in the visualization research literature, either in the context of design spaces [11] or in task or interaction taxonomies [19, 24, 38]. Thus we set out to survey a corpus of annotated charts and identify a design space for annotation.

With regards to the scope of our design space, there are several chart elements that we do not consider to be forms of annotation. First, we exclude graphical marks that correspond directly to the underlying data, such as bars in a bar chart or the points in a scatterplot. Unlike Kirk's characterization of annotation [28], we also exclude nameable graphical and textual elements associated with a variety of charts, maps, and plots; these include chart titles, legends, axes, axis labels and tick marks, as well as grid lines or graticules. Though we do not consider these elements to be annotations, they can certainly *be annotated*, as we explain below in Section 3.2. Finally, we constrain our scope to visual channels of communication.

3.1 Survey of Annotated Charts

There is an abundance of annotated visualization artefacts that could help us identify a design space for annotation in the context of visual data-driven storytelling. Annotated charts can be readily found in scientific publications, journalistic media, information graphics, as well as in government and organizational memorandum [6]. These charts also appear in the context of live presentations [29], such as those delivered in educational or conference settings, where the speaker can elaborate further upon the charts and their annotations.

Our survey was motivated by the analysis of a corpus of such presentations from the "Quantified Self" or personal data tracking community; Choe et al. documented how quantified selfers visually presented their insights, which included a discussion of annotation [14]. The quantified selfers commonly used annotations to effectively communicate their personal insights. The most common annotations included text, shapes, trend lines, ranges highlighted via color or texture segmentation, and lines indicating meaningful values. To further inform our design space, we surveyed annotated charts found in journalistic media. We selected this domain for three main reasons: such charts are intended for a large audience, they are numerous and readily accessible, and most importantly, they are often intended to support or tell a story, such as by accompanying a news article.

We began by collecting a small pilot corpus of 30 annotated charts from sources including *The New York Times* and *FiveThirtyEight*. By inspecting this corpus and by considering the findings of Choe et al. [14], we formulated an initial design space of annotation targets and annotation forms.

Like the machine learning approach of validating whether a classification based on an initial training set generalizes appropriately, we tested against a larger representative corpus of annotated charts to validate our classification of annotation targets and forms. We began with the collection of 705 news charts from the Massvis 2k dataset [6], which were predominantly variants of bar charts, line graphs, and scatterplots published by *The Economist* and *The Wall Street Journal (WSJ)*. We identified 257 of these charts as having some form of graphical or textual element meeting our criteria for annotation. To be clear, our aim was not to quantify the various forms of annotation or make claims about their prevalence, but to identify unique annotations. For instance, consider two bar charts published by the same news desk that use the same style guidelines; if both charts included text annotation adjacent to bars, we only retained one of these two charts in our corpus. However, if one of these two charts contained an additional unique form of annotation, such as a line perpendicular to the bars indicating the average value of all bars, both charts were retained in our corpus. After discarding charts with duplicate forms of annotation, we were left with 23 *Economist* charts and 39 *WSJ* charts. Acknowledging the stylistic differences and the different forms of annotation employed by these two organizations, we then decided to gather additional annotated charts from four other news graphics desks: *The New York Times* (10), *FiveThirtyEight* (11), *The Washington Post* (11), and *Bloomberg Visual Data* (12). To ensure comparability with the *Economist* and *WSJ* charts from the Massvis dataset, we limited ourselves to variants of bar charts, line graphs, and scatterplots containing a unique form of annotation.

Altogether, we arrived at a corpus of 106 annotated charts from six sources; this corpus, labeled with our design dimensions, along with links to original sources, is available at `https://chartaccent.github.io/#section-survey`. For each chart in this corpus, we characterize its type (e.g., bar chart, line graph), the annotated chart elements, annotation forms, the visual properties of annotations such as font size, color, and stroke width, along with additional comments.

3.2 Analysis

Informed by our survey of annotated charts, we characterize two design dimensions for annotation in the context of visual data-driven storytelling: annotation form and annotation target.

The question of why a person would annotate a chart cross-cuts these two dimensions: a particular annotation form applied to a particular target will ideally achieve a specific effect, to add interpretive value [33]. In some cases, the act of annotation is self-serving; consider a student who annotates a textbook as a means to study, or a person who annotates a calendar to serve as a personal reminder. In other cases, the act of annotation is intended to enable communication: to attract and orient the audience, to explain and facilitate interpretation [28], to draw their attention and emphasize one or more elements in a document, to separate and distinguish elements [43], and to provide context or editorial commentary with reference to these elements. Prior discussion of annotation in the visualization literature refers to both its personal and communicative purposes, and the act of annotation has often been associated with the abstract notion of "insights": manipulating them [19], externalizing them [38], as well as recording, organizing, and communicating them [24]. As the motivation for this paper pertains to visual data-driven storytelling, we are most interested in the communicative purposes of annotation.

Dimension 1: Annotation Form

Many definitions of annotation tend to imply that an annotation is merely text commentary that has been added to a document; we find such definitions to be too narrow. According to Marshall [33], an annotation can be any kind of superstructural element added to a document, and her study of students' marginalia in textbooks suggests that annotations can take many forms beyond mere text. In

the visualization literature, annotation is a "layer of user assistance and user insight" [28], a single annotation is regarded abstractly as meta-information related to some data, and the annotation is represented with a visual object [19]. Other definitions are less abstract [24, 38], referring to graphical or textual elements added to a visualization artefact (e.g., a chart, a map, a plot, a graph).

Based on our survey of annotated charts, we distinguish four forms of visual annotation: text, shapes, highlights, and images.

Text: Data-driven text annotations indicate values corresponding to data-bound chart elements, such as the attribute value pair of a point in a scatterplot, or the upper and lower bounds of a range along one attribute; examples include the items in a set (Figure 2-c) and the average temperature for a series (Figure 2-h). When only a subset of data-bound chart elements is annotated, the intent is to draw the viewer's attention to them before they examine elements lacking any annotation. Other text annotations that are not data-driven can provide additional context, orientation, or editorial comment, such as in Figure 2-g. Text annotations also have a number of visual properties including those pertaining to font, justification, padding, wrapping, and position relative to the annotation target.

Shapes: These annotations can be distinguished by their type (e.g., line, arrow, curve, rectangle, ellipse, bracket, star, speech bubble), by their stroke and fill values, and by their position relative to the annotation target. Rectangles or ellipses can be placed to contain chart elements, prompting the viewer to acknowledge the importance of these elements or to compare them to elements outside of the set. Arrows, stars, and symbols can be used to direct the viewers' attention to a single element including an annotation added to emphasize a group of elements. Like text annotations, shape annotations can also be data-driven; for instance, trend lines in a scatterplot require calculations to be performed on the underlying data, such as in Figure 1-right.

Highlights: This form of annotation involves altering or embellishing the target to emphasize or diminish its importance. A highlight can be distinguished by the visual properties of the target that it alters, such as its size or its stroke and fill values. For instance, items in Figure 2-c have an orange fill and stroke, distinguishing them from other temperatures in the Chicago or Phoenix series.

Images: Images and icons added to targets can be distinguished by their size, their opacity or saturation, their position relative to the target, including whether they are in the foreground or background. Examples of image annotations include the flags in Figure 1-right. Highly salient image annotations may be used to promote the memorability of a chart [7].

Combining annotation forms: Note that any single target can be annotated with several annotations. For instance, a point in a scatterplot can be highlighted via a different fill color and stroke, it may have a text annotation displaying its value, and a dropline or arrow may originate from the text annotation to the point.

Dimension 2: Annotation Target Type

We distinguish four types of targets, where a target is the object or objects being annotated: data items, structural chart elements, coordinate spaces, and prior annotations. These target types were represented throughout our corpus of annotated charts, which consisted of variants of bar charts, line graphs, and scatterplots; as a result, our current set of targets may not account for all possible targets in other chart types.

Data item, set, & series targets: These targets correspond with data: (a) a single item in the data, such as Phoenix's May temperature in Figure 2-a; (b) a series of items reflecting relations in the underlying data, such as Phoenix's series of average temperature values in Figure 2-b; or (c) a set of items, such as Chicago and Phoenix average temperatures on November and December in Figure 2-c. When a chart contains multiple series, a set could include
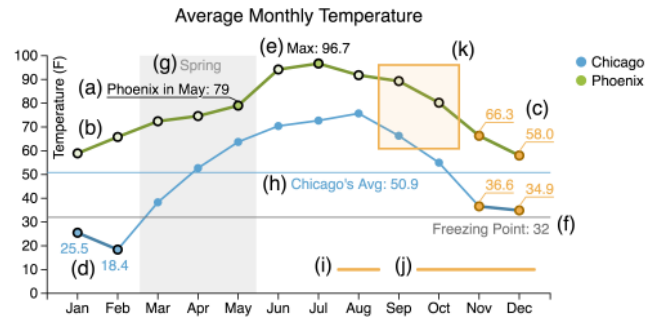

Figure 2: An example line chart of average monthly temperatures indicating the types of data item, set, & series targets (a–e) and coordinate space targets (f–k); the parenthetical labels (a–k) are annotations used to reference prior annotations.

items from more than one series. Additional data item targets include: (d) items that satisfy inequalities, such as temperatures below the freezing point in Figure 2-d; or (e) items with extreme values, such as the maximum overall temperature in Figure 2-e. The purpose of annotating these targets is to indicate their importance, to serve as exemplars for other nearby data items, or to orient the viewer to the chart's coordinate system.

Coordinate space targets: These targets are specific to the coordinate system of the chart; we assume a Cartesian coordinate system with one or two quantitative scales, which is reflected in our survey of annotated charts. Coordinate space targets include: (f) those that refer to a value such as the freezing point of $32°$ in Figure 2-f; (g) a span along an attribute, such as between March and May in Figure 2-g; or (h) a minimum, mean, median, or maximum value for a particular series, such as the average Chicago temperature in Figure 2-h.

Coordinate space targets also include those that refer to value pairs or spans along two attributes: (i) a point, such as $10°$ in August (Figure 2-i); (j) a partial span at a specific attribute value, such as $10°$ in a range between October and December (Figure 2-j); or (k) a span along two dimensions, such as the region in Figure 2-k. Shape annotations for coordinate space primarily serve to provide context and orientation to the viewer, to emphasize important positions or ranges, including those that contain no data items.

Chart element targets: These targets include the chart title, the axes, the axes labels and tick marks, the legend, the plot area of the chart, and finally the chart itself. It is important to distinguish the final two target types in this list: the plot area refers only to the area encompassed by the coordinate system indicated by the axes, whereas the entire chart encompasses all of the preceding elements in this list. For instance, a text annotation within the plot area may provide additional context about the data or help orient the viewer by explaining the choice of scale or range. In contrast, an annotation on the entire chart typically appears on the periphery of the chart, encompassing captions, credits, and footnotes, providing additional context information or text that would be too verbose to include within the plot area.

Prior annotations: Annotations are by definition additive, and thus a previous annotation can be the target of additional annotations. For example, the parenthetical labels (a–k) in Figure 2 are annotations used to reference prior annotations.

4 CHARTACCENT

We used our design space to design and implement ChartAccent, a tool that provides a palette of interactions for the data-driven annotation of a chart. This section contains our design rationale and ChartAccent's treatment of annotation forms and targets.
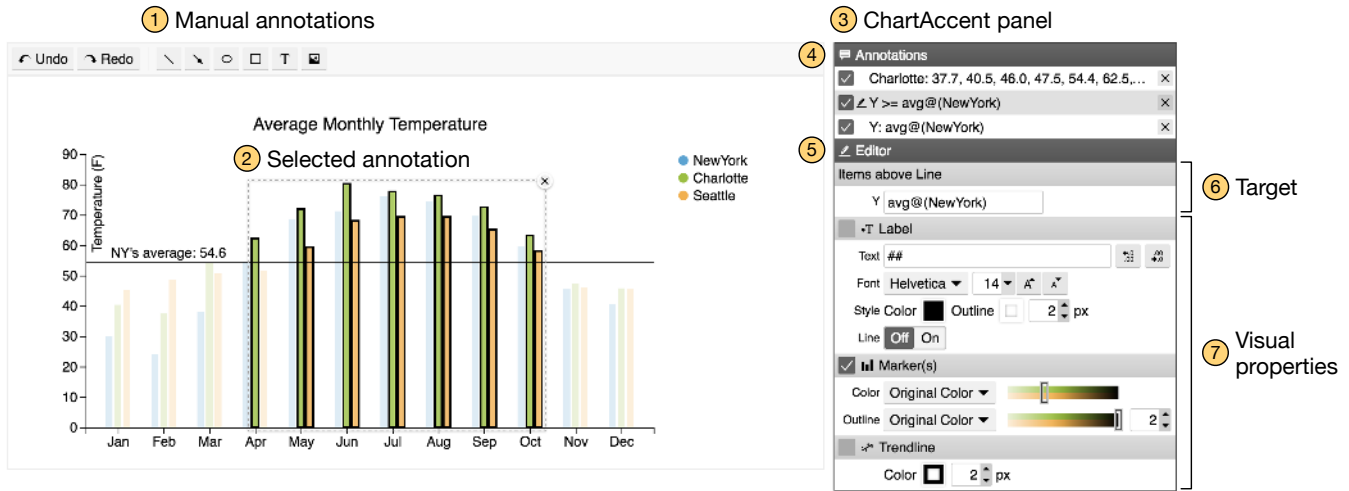
Figure 3: ChartAccent's user interface; (1) manual annotations; (2) the selected annotation represented by a gray dotted rectangle; (3) the ChartAccent control panel; (4) the list of annotations currently applied to the chart; (5) the annotation editor, which allows one to modify (6) the annotation target; and (7) the visual properties of the selected annotation.

4.1 Interacting with ChartAccent

We decided to separate interactions for annotation from those for chart creation. This has two benefits: (1) as a research prototype, it allows us to evaluate annotation interactions independently; (2) the interactions we designed for ChartAccent are transferable and can be ported to other systems used to create charts. There are two classes of annotation interactions: those that specify the annotation target(s), and those that specify or modify the annotation form.

Annotation Target Selection

Following a typical user interface selection mechanism, mouseover highlights the target to be annotated, and mouse click selects and annotates the target; a mouse drag, lasso, or click while holding a modifier key results in a selection for either a coordinate space target or data item target.

ChartAccent includes an implementation of bubble cursor selection [21]: when approaching a data item target (e.g., a series in a line chart), the cursor passes an initial distance threshold and the entire series becomes a selection target; after a second threshold closer to the line, either a data item (an inflection point in the line) or a line segment and the two data items that it joins becomes the selection target, depending on which is closer to the cursor. Meanwhile, a coordinate space target can be selected by clicking a position along an axis or by dragging over a span along an axis.

Data-driven selection: ChartAccent adopts aspects of Heer et al.'s selection design [23], in which a person can click on a legend item to trigger annotations for a series of data items that share the selected categorical value.

Once the default annotation for a selected target appears, the position or span can be adjusted using ChartAccent's target editor (Figure 3-6); this novel counterpart to direct target selection allows a person to compose basic formulas that can reference the attributes of the dataset. The editor features auto-complete support and includes a set of basic statistical functions including minimum, maximum, mean, and median. For instance, the horizontal line annotation at $y = 54.6$ in Figure 3 representing New York's average temperature was positioned using such a formula.

Upon selection of a coordinate space target, the target editor provides a novel mechanism to select additional targets relative to these positions or spans: the ability to select data items above or below a position or within or outside a span without having to manually click or drag over them; the Charlotte and Seattle monthly temperature values higher than the New York average in Figure 3 were selected in this manner. This selection is tightly coupled with the originally selected range target; if the line or rectangle annotation corresponding to the original coordinate space target is adjusted, ChartAccent automatically updates any associated selections of data items.

Annotation Form Specification & Modification

As we encountered in our survey of annotated charts, annotation can take many forms and the visual properties of each form can vary tremendously. With ChartAccent, we established a default annotation form for each data item and coordinate space target type following its selection. The default annotation for an individually-selected data item is a black stroke border and a text annotation indicating the item's value, which appears adjacent to the item. The default annotation forms for a coordinate space targets are lines or rectangles drawn perpendicular to the axis from the selection position or span, respectively, and a text annotation indicating the values (Figure 2-g–h).

Annotation forms for set and series targets: When a selection involves more than one data item target, a trend line can be added to the selected set or series, as shown in Figure 1-right and Figure 6-4. Additionally, when a set or series are selected in a scatterplot, a Bubble Set [15] annotation can be added to the contour surrounding these targets, as shown in Figure 1-right and Figure 6-5.

Manual annotation forms: Chart element targets can be manually annotated by selecting a form (Figure 3-1) and positioning the resulting annotation anywhere on the chart via dragging; lines, arrows, ellipses, rectangles, text, and images can be used to annotate chart element targets. Coordinate space targets that do not intersect an axis (such as Figure 2-i,j,k) can also be annotated in this manner.

Annotation form modification: Once added to a chart, the visual properties of the annotation can be interactively modified. The position and size of existing annotations can be adjusted via dragging. Meanwhile, ChartAccent's control panel for modifying the visual properties of annotations mimics popular graphical tools (Figure 3-7). Annotations for sets and series of data items share the same visual properties, and thus each item does not need to be modified individually. Annotation properties include font properties such as type, size, and color, the visibility of a dropline connecting an annotation to its target, as well as stroke and fill properties.

4.2 Usage Scenarios

We illustrate the process of annotating charts with ChartAccent through two usage scenarios, which are also showcased in the supplementary video.

Scenario 1: Monthly Temperature Line Chart

In our first scenario, we consider average monthly temperatures for several American cities from July 2014 to June 2015 (data from FiveThirtyEight [17]). Let us imagine a situation in which we want to convince others that the weather in New York is generally not as favourable as the weather in Charlotte or Seattle. To support this argument, we will highlight the months when the temperature in Charlotte and Seattle is higher than New York's average temperature with a grouped bar chart.

We begin by diminishing the highly salient bars to provide a greater contrast for the bars that we want to highlight. To do so, we select all the bars by dragging over all of them. We then diminish the salience of the bars by adjusting their brightness in the visual properties editor (Figure 3-7).

Next, we indicate the average temperature in New York. To do so, we select a range target by clicking on a point along the Y axis, which results in a horizontal line annotation. Then, in the target editor (Figure 3-6), we replace the target value with `avg@(NewYork)`. We then prepend "NY's Average: " to the default label and drag it to the left side of the chart, as in Figure 3.

To highlight the months when Seattle and Charlotte's temperatures are above New York's average, we open the "Select Items Using this Line" dropdown menu, select "Above," "Charlotte," and "Seattle," leaving "New York" unselected. Once selected, black strokes and text annotations appear for each bar meeting this criterion; once again, we toggle off the visibility of the text annotations, but we keep the black strokes for the selected bars.

Finally, we export the resulting annotated chart (Figure 1-left) as a PNG or SVG image.

Scenario 2: Fertility Rate vs. Life Expectancy Scatterplot

In our second scenario, we consider fertility rate and life expectancy in several countries as of the year 2000 (data from Gapminder [18]). In particular, we want to (i) convey the overall trend that higher fertility rate is correlated with lower life expectancy, (ii) highlight North and South American countries, and (iii) highlight the United States, China, and India.

We begin by creating a scatterplot with these two attributes where countries are color-coded by world region and sized by population. Much in the same way that we diminished the salient bars in our first scenario, we diminish the fill value of the points. We then add a trend line by toggling trend line visibility (Figure 3-7). We manually add a text annotation to explain the trend line, as well as an arrow shape annotation to establish a visual connection between this text and the trend line.

To highlight countries in North and South America, we click on the "America" legend item to indirectly select all of the corresponding data items. We toggle off the visibility of the text annotations and we turn on the bubble set annotation for this selection.

To highlight United States, China, and India, we repeat the following steps for each country: selecting each country and moving the default text annotation (i.e., country name) to a nearby location where it does not occlude other points, then manually adding and positioning each country's flag as an image annotation. Since the text annotation for the United States is far from its corresponding point, we trigger the visibility of a dropline from it to the point.

Finally, after adding a text annotation indicating the year "2000," we can export the resulting chart (Figure 1-right).
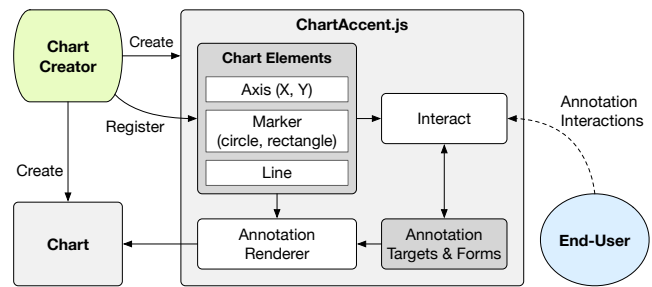

Figure 4: The ChartAccent architecture; the chart creator program that loads the data and renders the chart creates a ChartAccent object and registers chart elements; ChartAccent.js includes facilities for managing annotation layers and rendering an internal annotation representation to the chart in response to an end-user's interaction.
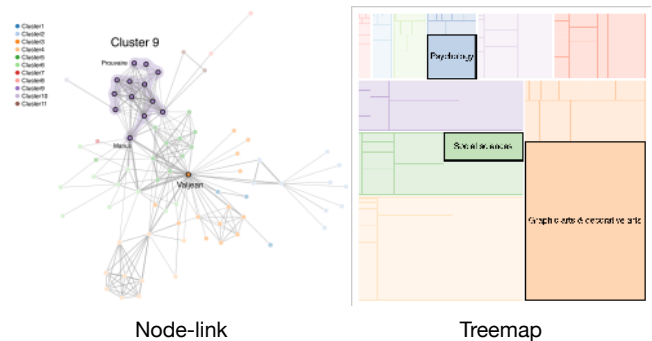

Node-link   Treemap

Figure 5: Data item annotations in a node-link graph and a treemap. Left: character co-occurrence graph in *Les Misérables*, with a bubbleset-highlighted set ("Cluster 9") and 3 other nodes selected, with other nodes diminished in salience (data from [8]). Right: checkout count by Dewey category from Seattle Public Library, highlighting three items and diminishing the salience of others (data used with kind permission from George Legrady).

4.3 Implementation Details

We initially implemented the ChartAccent.js library for annotating SVG-based charts (such as those generated using D3.js [9]). Figure 4 shows the relationship between ChartAccent.js, a chart creator, a chart, and an end-user. To make a chart "annotatable," the chart creator loads the data and renders the chart. The chart creator must also create a "ChartAccent object," register chart elements such as axes, marks, and legends as well as their metadata, and map marks with their corresponding data items as well as their default annotation form. ChartAccent.js also has functions for creating and managing annotation layers and their visual properties, as well as interaction handlers for the various forms of annotations.

ChartAccent.js supports charts comprised of the following elements: Cartesian coordinate systems of linear or ordinal scales, circle or rectangle marks, polylines connecting marks within series, and bullet-style series legends. ChartAccent.js thus can be used to annotate many variants of bar charts, line graphs, and scatterplots. For other types of charts, ChartAccent.js currently provides limited support for item-based annotations (such as in the annotated node-link graph and treemap featured in Figure 5).

Finally, ChartAccent itself is a standalone tool that encapsulates a basic chart creator interface and the ChartAccent.js library, which includes an interactive panel for modifying annotations and their visual properties (Figure 3-3).

5 EVALUATION: REPRODUCTION STUDY

To evaluate the experience of interacting with ChartAccent to annotate different types of charts encompassing a variety of annota-
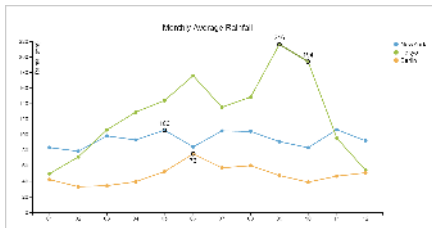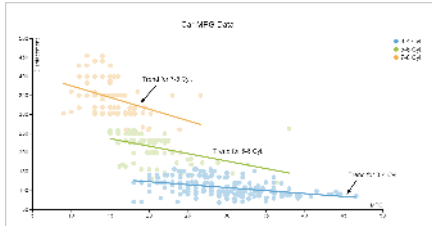
Task 1: Select and annotate four individual data items; drag text annotations to correct positions.
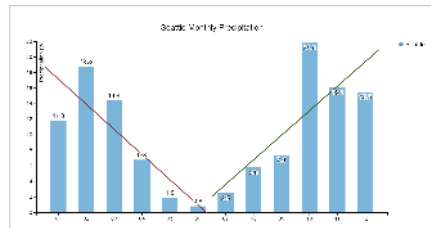
Task 2: Select and annotate two sets of data items; remove stroke highlights; add trend lines for both sets; modify trend line colors.
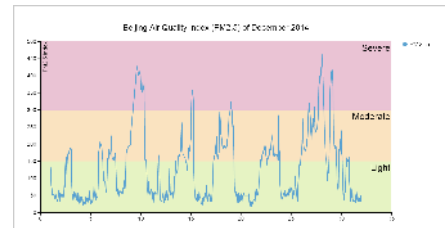
Task 3: Annotate three coordinate space targets; modify fill color of range annotations; edit text labels for these coordinate space annotations.

Task 4: Select each series; diminish their salience; add trend line for each series; add and position text and arrow annotations.
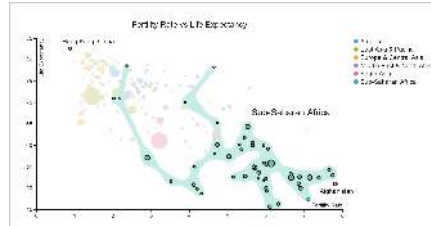
Task 5: Select all items and diminish their salience; annotate a series and four individual data items; modify and drag text annotations to correct positions.
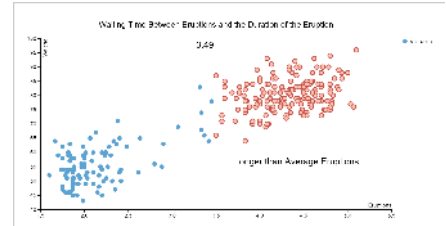
Task 6: Add a coordinate space annotation at the average x value; select and highlight all data items with a greater x value; add and position a text annotation.

Figure 6: Reference annotated charts used in the study for Tasks 1–6 along with the required steps to complete the task. Task 7 (Figure 1-left and Figure 3, Scenario 1): Select all series and diminish their salience; add line annotation to a coordinate space target corresponding to average New York temperature; select and annotate values above this value; select and annotate a single data item; edit and position text annotation.

tion targets and forms, we asked participants to reproduce a set of previously annotated charts, beginning with unadorned (i.e., unannotated) versions of the same charts.

The major question that motivated this evaluation was whether the set of interactions to support annotation in ChartAccent were learnable, usable, and efficient. We were particularly curious about data-driven annotation via direct and indirect selection, as the combinations of these features set ChartAccent apart from previous approaches for annotating charts.

5.1 Participants and Setup

We recruited 11 (3 women, 8 men) participants from the Greater Seattle area. All of our participants had created basic charts (e.g., bar charts, line graphs, pie charts, etc.) using commercial tools within the past three months; they also stated that they had previously used charts for communication purposes, such as in a live presentation. These participants had normal or corrected-to-normal vision, however one participant was color-deficient but not color-blind. Various occupations were represented among our participants, including a receptionist, a real estate broker, a business analyst, a software engineer, and a user experience researcher. The average age of our participants was 35, ranging from 23 to 50 years of age. Participants were compensated with a software gratuity.

We used a 3.6 GHz Windows 8 desktop machine with 32 GB RAM, using two side-by-side 24-inch Dell LCD displays running at 1920×1080 resolution; the left monitor was in a portrait orientation. For all tasks, we logged start, reset, and end times, and we saved the result annotated chart as an image. We also captured screen recordings along with concurrent video and audio recordings of the participants as they interacted with ChartAccent.

5.2 Tasks and Datasets

We prepared seven annotated charts using three chart types: Tasks 1 and 3 featured a line graph, Tasks 2 and 7 featured a bar chart, and Tasks 4–6 featured a scatterplot, as shown in Figure 6; step-by-step animations of these tasks are available on the supplemental website. These tasks and the corresponding annotated charts

were selected to encompass ChartAccent's capabilities and featured a variety of annotation forms and targets. Each subsequent task/chart increased in terms of complexity: from individual data items to set and series targets, the annotation of coordinate space targets, the combination of data-driven and manually-added annotation, and an increasing number of visual property modifications. Figure 6 summarizes the steps required to complete each task, and a step-by-step animated GIF for each task is available at `https://chartaccent.github.io/#section-examples`. We also prepared an additional twelve annotated charts for practice tasks.

The datasets to create these charts included those pertaining to: monthly average rainfall in various cities [48] (Task 1); monthly precipitation in Seattle [48] (Task 2); the Beijing air quality index [3] (Task 3); mileage statistics for various cars [32] (Task 4); fertility rate and life expectancy in various countries [18] (Task 5); Old Faithful Geyser eruptions [2] (Task 6); and monthly temperatures for major American cities [17] (Task 7).

5.3 Procedure

We began with a brief explanation of the study goals and overall procedure. We then asked the participants to complete a pre-study background questionnaire, and led them through a tutorial on the core concepts and features of ChartAccent, with an emphasis on coordinate space targets as well as the combination of data-driven selection via direct manipulation and indirect selection via the target editor; participants were encouraged to interact with the tool during this tutorial. On average, the tutorial lasted 36 minutes.

Following the tutorial, the participants performed twelve practice tasks to familiarize themselves both with the task procedure and with ChartAccent. For each task, we showed a static annotated chart as a reference on the left monitor, and asked the participants to reproduce the same chart on the right monitor with ChartAccent, which was embedded within a study management application. We also showed the original chart without any annotations as an image below the annotated chart as a reference.

Before starting each task, we asked the participants to verbalize the required annotations based on what they saw in the annotated

reference chart; we did so to ensure that the participants did not miss any annotations from inattention, and to provide them with an opportunity to seek clarification regarding the properties of each annotation, as properties such as line thickness and fill color were not always clearly distinguishable, especially for small targets. After we confirmed that the participants understood all of the required annotations, we asked them to press a "Start" button to load the editable version of the chart and begin the task. We also asked them to press a "Submit" button after completing the task, which would save the resulting chart as an image and advance to the next task. We allowed the participants to press a "Reset" button, which would remove all annotations from the chart and restart the task (but not the timer). We encouraged the participants to think aloud, especially when any aspect of the task or ChartAccent was confusing or unclear. On average, the time to complete the twelve practice tasks was about 22 minutes.

After completing the 12 practice tasks, the participants repeated the same procedure with Tasks 1–7. We provided hints to the participants either when their progress stalled or when they tried to submit the results with incorrect or missing annotations; for the latter, we pointed out the errors and asked the participants to fix them. We noted the cause of the stall or error in both cases. We encouraged the participants to complete the task as quickly as possible, and explained that they did not have to match the exact x,y position or color of annotations in the reference chart, which would have been especially tedious for text annotations. On average, the time to complete all seven tasks was about 15 minutes.

At the end of the study session, participants filled out a questionnaire regarding their experience with ChartAccent. On average, the study session lasted about 1.5 hours.

5.4 Results

Nine out of 11 participants successfully reproduced the annotated charts for all seven timed tasks; the two remaining participants arrived late and ran out of time, and thus could only complete the first five tasks. Because we asked the participants to fix any errors without imposing a time limit, all of the resulting annotated charts they created were correct copies of the reference charts.

Hints and task completion time: As mentioned in our description of the procedure, we provided two types of hints to the participants. First, we explained what went wrong (e.g., the z-order of annotations prevented the addition of another annotation) or reminded the participants of ChartAccent's capabilities (e.g., indirect selection and annotation of data items relative to a coordinate space target via the target editor). Second, we pointed out the difference between the reference chart and the participants' outcome (e.g., missing text annotations or the addition of unnecessary annotations), which was usually caused by oversight. Table 1 shows the number of hints we provided for each task. Overall, participants successfully completed the tasks with very few hints, and they needed more hints for minor mistakes. The most hints were given during Task 5, which was similar to Scenario 2 and required a combination of data-driven annotations, manual text annotations, and form modifications.

With regards to task completion time, the overall average task time across all tasks and participants was 102.7 seconds (see Figure 7), indicating that charts of varying complexity requiring several steps can be completed quickly. The two longest task completion times (P4 and P7 in Task 5; see Figure 7) were due primarily to a z-order issue: *"The steps to hiding/obscuring and lines, if done out of sequence, it can be a little frustrating"* (P4). Some participants were unaware of the consequences of deleting an annotation, as sometimes they deleted an annotation with the intention of merely hiding it. This issue caused some delay in completion time when the form of the deleted annotation had been modified.

Participant feedback: Participants rated ChartAccent on four satisfaction criteria. All ratings were on a 1–7 Likert scale from

	# Hints: features & concepts		# Hints: missing, incorrect annotations	
Task	Total	Average	Total	Average
T1	0	.00	4	.36
T2	6	.55	10	.91
T3	3	.27	1	.09
T4	5	.45	7	.64
T5	13	1.18	16	1.45
T6	3	.33	7	.78
T7	8	.89	3	.33
Total	38		48	

Table 1: Total and average number of hints per task. (P1 and P5 completed only the first five tasks.)
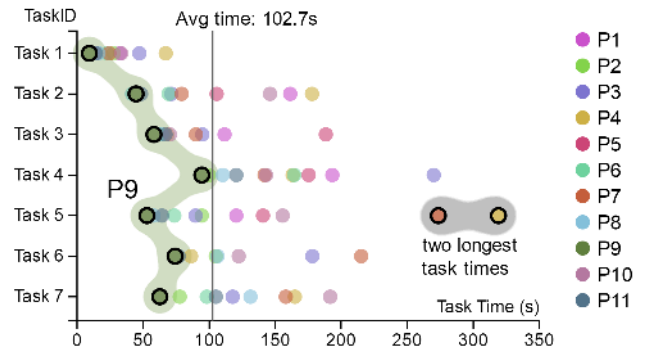

Figure 7: Task completion time by participant and by task; P9 (a software engineer and prolific maker of charts) had the shortest completion time for most tasks. (This figure was generated with ChartAccent.)

"Strongly Disagree" to "Strongly Agree." Overall, ChartAccent was rated fairly highly; participants deemed it easy to learn (Avg = 5.8) and use (6.2), they found the annotation creation process to be enjoyable (6.7), and they indicated a desire to use ChartAccent to annotate charts in the future (6.9).

In a post-study interview, participants elaborated on the usefulness of ChartAccent: *"Very cool idea, I can see it being very useful for presenting data. [I] definitely can think of times where I wish I had this kind of tool to make my presentations easier to make, and better overall."* (P8); *"I liked [that] it was semantic based. ... Overall I liked it a lot actually. Because it's dynamic, for the most part, it means that I will be able to modify values and the annotation might be able to follow the similar rules right out of the box. That's cool."* (P11 referring to ChartAccent's data-driven annotation and the target editor in particular); *"No more photo editing! This will save me time."* (P4). Participants also mentioned how ChartAccent was easy to learn: *"[there is] good indication about what will be selected, straightforward interface (very similar to existing products) making it easy to learn"* (P2); *"It had [a] familiar interaction language that I'm used to, so learning was relatively easy and using it was intuitive."* (P9, a software engineer who is familiar with Adobe Photoshop). On the other hand, several participants noted learnability issues: *"There was a lot to remember."* (P5, who was unfamiliar with marquee/lasso and keyboard shortcut conventions).

6 DISCUSSION AND FUTURE WORK

We now discuss the lessons learned during the design and evaluation of ChartAccent, along with directions for future work.

6.1 Reflection on the Survey of Annotated Charts

Our work was initially inspired by the analysis of annotated charts used personal data presentations by "quantified selfers" [14], which investigated *what* people annotate in terms of personal insights and *how* they annotate those insights in terms of annotation forms. To validate and build upon this analysis, we surveyed annotated charts found in journalistic media; these charts are targeted for a broader audience and created by professionals for asynchronous reading. We observed a similar annotation targets and annotation forms in charts emanating from these two domains.

While our analysis of annotated charts from these domains may not fully encompass all possible existing forms of annotation, they helped us distill a design space, which in turn informed the design of ChartAccent. In the future, it would be helpful to compare against annotated charts from other domains, such as those used in public policy, corporate governance, and education.

6.2 Improving and Extending ChartAccent

In the months following the reproduction study, we addressed usability issues that we identified during the study, including the ability to adjust the z-order of annotations by reordering the annotation list (Figure 3-4), and by adding basic undo and redo functionality.

On a larger scale, the question of how to enable people to effectively construct charts and other visualization artefacts is an ongoing research problem [20]. Instead of building a comprehensive chart authoring system, we decided to focus on developing and testing a set of interactions to support chart annotation. In addition, we ensured that our ChartAccent.js library would be compatible with existing charts generated with D3.js [9] and would not be limited to charts created with the current standalone ChartAccent tool. As indicated in Figure 4, we envision cases in which chart creators use function calls to register chart elements created with D3.js. One way to achieve this goal could involve incorporating D3 Deconstructor [22] for automatically determining the underlying data and the visual mappings of a chart, thus enabling ChartAccent to import existing charts without having to specify the chart elements.

Currently, ChartAccent supports variants of bar charts, line graphs, and scatterplots. Although ChartAccent can be used to add annotations to data item targets in other chart types, such as node-link graphs or treemaps (see Figure 5), current support for charts that do not have a Cartesian coordinate system is limited. As such, future work remains to extend ChartAccent to other coordinate systems and chart types.

6.3 Desirable Annotation Form Defaults and Templates

In our reproduction study, we learned that the default annotation forms applied by ChartAccent were often not ideal for the tasks that we had selected, which were intended to assess a broad range of features and required several steps to complete. We note that, in general, desirable default annotation forms depend upon context and personal preference. For instance, each newsroom has a style guide for charts, and the chart author may have annotated similar charts of the same type in the past in accordance with this style guide; in such cases, selecting the form of annotations and modifying their visual properties might become very tedious. We plan to allow people to configure default annotation forms instead of redesigning ChartAccent to best support the particular tasks that we examined in our reproduction study.

Furthermore, as P11 acknowledged, annotation interactions can be reused, as in the case of Photoshop's "Action" feature: people can record interactions and replay the steps to complete the tasks automatically. These annotations could also be saved as a template and applied to an updated dataset having the same schema. For example, in Scenario 1, we could highlight the months with temperatures higher than New York's average for every year by applying the same annotation template. For additional flexibility, data-driven annotations can be applied to a different dataset with a similar schema by devising an annotation specification language. As the selection and the visual properties of annotations can become complex with many annotations, it would be helpful (if not necessary) to have an editor for annotation specification templates. We view the development of such a language and corresponding template editor as important future work.

6.4 Study Limitations and Web Deployment

As a preliminary evaluation, the main goal of the reproduction study was to assess the usability and learnability of ChartAccent and to assess if people could reproduce examples efficiently without excessive prompting or aid from the researchers.

We therefore specified the set of tasks used in our usability study such that they would encompass the broad range of features and interactions introduced in ChartAccent, such as those related to data-driven selection via direct manipulation and indirect selection using the target editor. However, since the charts and the data they depict were not personally meaningful to the study participants, we cannot assume that they were highly engaged and motivated to annotate these charts. As a result, we have yet to truly assess the expressiveness of ChartAccent; as a next step, we plan to study the annotation of charts containing data provided by participants.

Furthermore, though we emphasized to our participants that we were evaluating ChartAccent and not their performance, any study protocol where a researcher is observing over their shoulder and pointing out mistakes may have imposed a high level of stress. We suspect that this pressure may have caused some of the errors or obsessive detail-checking, inflating our task completion times. With respect to their subjective feedback, we acknowledge the possibility of a social desirability bias, particularly because we did not perform a comparative evaluation between alternative approaches.

After refining ChartAccent, we made ChartAccent available on the web: `https://chartaccent.github.io`. People can use ChartAccent with their own data, add personally meaningful annotations, and share the resulting annotated charts. ChartAccent allows people to create an annotated chart via three steps: 1) import the data, 2) select a chart type and configure chart axes, and 3) annotate the chart. In addition, it allows them to export the annotated chart either as a static image (in PNG or SVG) or as an animated GIF, which illustrates the step-by-step annotations. To assess post-deployment adoption, we plan to monitor usage as well as collect and analyze the annotated charts people created with ChartAccent as a way to understand *how* people use ChartAccent, *what* they annotate, and how we can further improve the annotation experience.

7 CONCLUSION

In this paper, we reflected upon annotation in the context of visual data-driven storytelling. To provide empirical knowledge on the spectrum of annotations used in data-driven storytelling, we characterized a design space of annotation informed by a survey of 106 annotated charts. Using this design space, we designed and implemented ChartAccent, a tool that provides support for manual and data-driven annotation. ChartAccent provides a novel way of defining and using data-driven annotations, and we envision that our annotation interactions can be integrated into other charting environments. To evaluate the experience of creating annotations with ChartAccent, we found that most participants could easily reproduce a series of previously annotated charts using ChartAccent after a short tutorial and practice session. We also found that ChartAccent's design and interaction provides an enjoyable experience, and all of our study participants expressed a strong desire to use ChartAccent to annotate charts in the future. Our deployment of ChartAccent will help us better understand how people annotate charts as a means of telling stories with data.

REFERENCES

[1] G. Aisch. Seven features you'll want in your next charting tool. Presented at NICAR, 2015. `http://goo.gl/cywIqO`. 2.2

[2] A. Azzalini and A. W. Bowman. A look at some data on the Old Faithful Geyser. *J. Royal Statistical Society. Series C (Applied Statistics)*, 39(3):357–365, 1990. 5.2

[3] @BeijingAir. `https://twitter.com/BeijingAir`. 5.2

[4] A. Bigelow, S. Drucker, D. Fisher, and M. Meyer. Reflections on how designers design with data. In *Proc. Intl. Working Conf. Advanced Visual Interfaces (AVI)*, pages 17–24, 2014. 1, 2.2

[5] A. Bigelow, S. Drucker, D. Fisher, and M. Meyer. Iterating between tools to create and edit visualizations. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 23(1):481–490, 2017. 1, 2.2

[6] M. A. Borkin, Z. Bylinskii, N. W. Kim, C. M. Bainbridge, C. S. Yeh, D. Borkin, H. Pfister, and A. Oliva. Beyond memorability: Visualization recognition and recall. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 22(1):519–528, 2016. 3.1

[7] M. A. Borkin, A. A. Vo, Z. Bylinskii, P. Isola, S. Sunkavalli, A. Oliva, and H. Pfister. What makes a visualization memorable? *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 19(12):2306–2315, 2013. 1, 3.2

[8] M. Bostock. Force-Directed Graph, 2016. `https://bl.ocks.org/mbostock/4062045`. 5

[9] M. Bostock, V. Ogievetsky, and J. Heer. D3: Data-driven documents. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 17(12):2301–2309, 2011. 2.2, 4.3, 6.2

[10] C. Bryan, K.-L. Ma, and J. Woodring. Temporal summary images: An approach to narrative visualization via interactive annotation generation and placement. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 23(1):511–520, 2017. 2.2

[11] S. K. Card and J. Mackinlay. The structure of the information visualization design space. In *Proc. IEEE Symp. Information Visualization (InfoVis)*, pages 92–99, 1997. 3

[12] Y. Chen, S. Barlowe, and J. Yang. Click2annotate: Automated insight externalization with rich semantics. In *Proc. IEEE Symp. Visual Analytics Science and Technology (VAST)*, pages 155–162, 2010. 2.2

[13] Y. Chen, J. Yang, S. Barlowe, and D. H. Jeong. Touch2Annotate: Generating better annotations with less human effort on multi-touch interfaces. In *Extended Abstracts ACM Conf. Human Factors in Computing Systems (CHI)*, pages 3703–3708, 2010. 2.2

[14] E. K. Choe, B. Lee, and m. c. schraefel. Characterizing visualization insights from quantified selfers' personal data presentations. *IEEE Computer Graphics and Applications*, 35(4):28–37, 2015. 1, 2.1, 3.1, 6.1

[15] C. Collins, G. Penn, and S. Carpendale. Bubble sets: Revealing set relations with isocontours over existing visualizations. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 15(6):1009–1016, 2009. 4.1

[16] M. Elias. *Enhancing User Interaction with Business Intelligence Dashboards*. PhD thesis, École Centrale Paris, 2012. 1, 2.2

[17] FiveThirtyEight. Data and code behind the stories and interactives at FiveThirtyEight. GitHub repository. `https://github.com/fivethirtyeight/data`. 4.2, 5.2

[18] Gapminder Foundation. `http://gapminder.org`. 4.2, 5.2

[19] D. Gotz and M. X. Zhou. Characterizing users' visual analytic activity for insight provenance. In *Proc. IEEE Symp. Visual Analytics Science and Technology (VAST)*, pages 123–130, 2008. 3, 3.2, 3.2

[20] L. Grammel, C. Bennett, M. Tory, and M.-A. Storey. A survey of visualization construction user interfaces. In *Proc. EuroVis Short Papers*, pages 19–23, 2013. 2.2, 6.2

[21] T. Grossman and R. Balakrishnan. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proc. ACM Conf. Human Factors in Computing Systems (CHI)*, pages 281–290, 2005. 4.1

[22] J. Harper and M. Agrawala. Deconstructing and restyling D3 visualizations. In *Proc. ACM Symp. User Interface Software and Technology (UIST)*, pages 253–262, 2014. 6.2

[23] J. Heer, M. Agrawala, and W. Willett. Generalized selection via interactive query relaxation. In *Proc. ACM Conf. Human Factors in Computing Systems (CHI)*, pages 959–968, 2008. 2.2, 4.1

[24] J. Heer and B. Shneiderman. Interactive dynamics for visual analysis: A taxonomy of tools that support the fluent and flexible use of visualizations. *Comm. ACM*, pages 1–26, 2012. 2.2, 3, 3.2, 3.2

[25] J. Heer, F. B. Viégas, and M. Wattenberg. Voyagers and voyeurs: Supporting asynchronous collaborative information visualization. In *Proc. ACM Conf. Human Factors in Computing Systems (CHI)*, pages 1029–1038, 2007. 2.2

[26] J. Hullman and N. Diakopoulos. Visualization rhetoric: Framing effects in narrative visualization. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 17(12):2231–2240, 2011. 2.1

[27] E. Kandogan. Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations. In *Proc. IEEE Conf. Visual Analytics Science and Technology (VAST)*, pages 73–82, 2012. 2.2

[28] A. Kirk. *Data Visualization: A Successful Design Process*. Packt, 2012. 1, 3, 3.2, 3.2

[29] R. Kosara and J. Mackinlay. Storytelling: The next step for visualization. *IEEE Computer*, 46(5):44–50, 2013. 1, 2.1, 3.1

[30] B. Lee, N. Henry Riche, P. Isenberg, and S. Carpendale. More than telling a story: Transforming data into visually shared stories. *IEEE Computer Graphics and Applications*, 35(5):84–90, 2015. 2.1

[31] B. Lee, R. H. Kazi, and G. Smith. Sketchstory: Telling more engaging stories with data through freeform sketching. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 19(12):2416–2425, 2013. 1

[32] M. Lichman. UCI Machine Learning Repository, 2013. `http://archive.ics.uci.edu/ml`. 5.2

[33] C. C. Marshall. Annotation: From paper books to the digital library. In *Proc. ACM Intl. Conf. Digital Libraries*, pages 131–140, 1997. 3.2, 3.2

[34] L. H. Owen. At the Malofiej Infographics World Summit, "the best form of storytelling is often static". *NiemenLab*, 2016. `http://goo.gl/zMfHk5`. 2.2

[35] A. Pearce. swoopyDrag.js: Artisinal label placement for d3 graphics, 2016. `http://1wheel.github.io/swoopy-drag/`. 2.2

[36] D. Ren, T. Höllerer, and X. Yuan. iVisDesigner: Expressive interactive design of information visualizations. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 20(12):2092–2101, 2014. 2.2

[37] L. C. Rost. What I learned recreating one chart using 24 tools. *Source*, 2016. `https://goo.gl/uGE5dc`. 2.2

[38] R. E. Roth. An empirically-derived taxonomy of interaction primitives for interactive cartography and geovisualization. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 19(12):2356–2365, 2013. 3, 3.2, 3.2

[39] A. Satyanarayan and J. Heer. Lyra: An interactive visualization design environment. *Computer Graphics Forum (Proc. EuroVis)*, 33(3), 2014. 2.2

[40] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 16(6):1139–1148, 2010. 1, 2.1

[41] C. D. Stolper, B. Lee, N. Henry Riche, and J. Stasko. Emerging and recurring data-driven storytelling techniques: Analysis of a curated collection of recent stories. Technical Report MSR-TR-2016-14, Microsoft Research, 2016. `https://goo.gl/jlGpyf`. 2.1

[42] Tableau Software. `http://tableau.com`. 2.2

[43] E. Tufte. Layering and separation. In *Envisioning Information*, pages 52–65. Graphics Press, 1990. 3.2

[44] F. B. Viégas, M. Wattenberg, F. van Ham, J. Kriss, and M. McKeon. ManyEyes: A site for visualization at internet scale. *IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis)*, 13(6):1121–1128, 2007. 2.2

[45] H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer, 2009. 2.2

[46] W. Willett, J. Heer, J. Hellerstein, and M. Agrawala. Commentspace: Structured support for collaborative visual analysis. In *Proc. ACM Conf. Human Factors in Computing Systems (CHI)*, pages 3131–3140, 2011. 2.2

[47] K. Wongsuphasawat. labella.js, 2015. `http://twitter.github.io/labella.js/`. 2.2

[48] World climate. `http://worldclimate.com`. 5.2