

Kori: Interactive Synthesis of Text and Charts in Data Documents

Shahid Latif, Zheng Zhou, Yoon Kim, Fabian Beck, and Nam Wook Kim

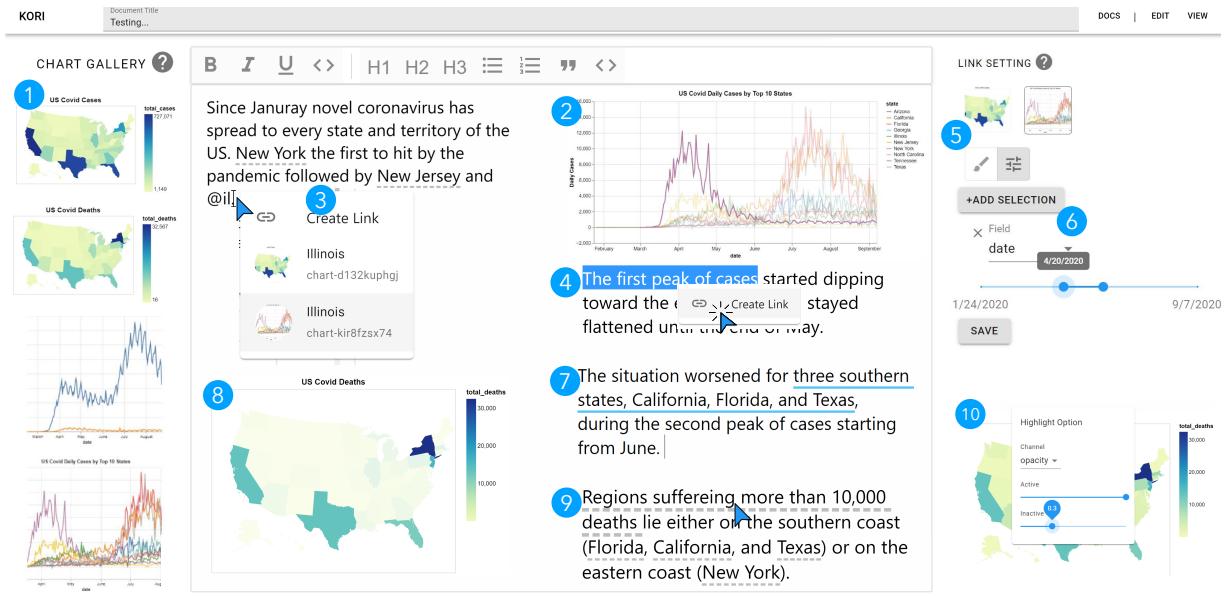


Fig. 1. The Kori system consists of a chart gallery (left), edit area (middle), and a link setting panel (right). Kori automatically suggests potential references (dotted gray underline) as a user types. Besides, it supports manual creation of links through simple interactions (4–6). The steps 1–10 describe a usage scenario to create an interactive story (see Sect. 5).

Abstract—Charts go hand in hand with text to communicate complex data and are widely adopted in news articles, online blogs, and academic papers. They provide graphical summaries of the data, while text explains the message and context. However, synthesizing information across text and charts is difficult; it requires readers to frequently shift their attention. We investigated ways to support the tight coupling of text and charts in data documents. To understand their interplay, we analyzed the design space of chart–text references through news articles and scientific papers. Informed by the analysis, we developed a mixed-initiative interface enabling users to construct interactive references between text and charts. It leverages natural language processing to automatically suggest references as well as allows users to manually construct other references effortlessly. A user study complemented with algorithmic evaluation of the system suggests that the interface provides an effective way to compose interactive data documents.

Index Terms—Data-driven storytelling, interaction design, authoring, visualization–text linking, mixed-initiative interface, interactive documents.

1 INTRODUCTION

As our society becomes more data-driven, the need for communicating data to a broader audience is inevitable. This has led to the emergence of narrative visualization and data-driven storytelling [21], going beyond traditional analysis-focused visualization systems. Charts are now frequently adopted in various forms of data stories such as news magazines, slide shows, and videos [55]. They are integrated with explanatory text to produce a synergistic effect. Charts can engage an audience and present a perceptually effective representation of data, while the text provides guidance to readers and explains the additional

context. Together, charts and text are essential for telling compelling stories and effectively conveying messages with data.

However, it is often challenging to synthesize information across two distinct media—the text and charts—as they are spatially separated apart [28]. The readers have to switch their attention back and forth to find references between text and visual marks on the charts that encode data values (e.g., bars, lines, points) and vice versa. This phenomenon is known as split-attention effect [6] in cognitive load theory [47]—or the contiguity principle in cognitive theory of multimedia learning [41]—that incurs a significant cognitive burden on readers’ working memory and can have a negative impact on learning [5, 14]. Several recent works have begun to explore ways of resolving this issue by demonstrating an explicit interactive linking between text and associated parts of charts [28, 32, 67] or through the use of signaling principle to incorporate visual cues to guide attention [41]. However, establishing and supporting such interactive references requires advanced programming knowledge that is mostly absent from authors of data-driven articles. Authoring tools for visualization and data storytelling [19, 24, 51] do not currently support the construction of such references.

In this work, we investigate ways to support the production of tight coupling between text and charts. To better understand their interplay,

• Shahid Latif and Fabian Beck are with University of Duisburg-Essen. E-mail: {shahid.latif, fabian.beck}@uni-due.de.

• Zheng Zhou and Nam Wook Kim are with Boston College. E-mail: {zheng.zhou, nam.wook.kim}@bc.edu.

• Yoon Kim is with Harvard University. E-mail: yoonkim@g.harvard.edu.

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org.

Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

we first analyzed text–chart pairs from existing sources. We expanded a collection from news media outlets containing bar charts [28] with additional chart types, including line charts, scatter plots, pie charts, and maps, and widened the scope to also include examples from scientific journals. We found that text–chart references have similarities to selection operations in a chart (e.g., point and interval selections). The textual phrases that refer to such selections can be aggregated in a hierarchical manner, similar to a semantic parse tree, and result in a *union* or *intersection* of the visual marks in the chart.

Based on the design space analysis of text–chart references, we developed Kori, a mixed-initiative interface that enables the synthesis of text and charts through interactive references; Fig. 1 shows the interface. It supports three main interactions: (i) accepting or rejecting suggested references, (ii) explicitly triggering suggestions ③, and (iii) manually constructing references using simple interactions ④ – ⑥. While authoring, Kori automatically identifies and suggests references ⑨. Our automatic suggestion approach leverages natural language processing to derive point- and interval-level references. Parsed dependencies group these references to generate higher-level references if two or more of them are semantically related.

Our evaluation of Kori is two-fold. First, we quantitatively evaluated the automatic reference suggestion approach. Second, we qualitatively evaluated the interface through a user study. We recruited eleven participants, including visualization experts, novices, and interface designers, and asked them to perform three tasks of reproducing and creating text–chart references, followed by a usability survey and a reflection interview. The results suggest that Kori provides a novel, intuitive, and easy-to-use interface to write an interactive data-driven document, that has the potential to advance the current practice of existing authoring tools.

2 RELATED WORK

We review the existing literature from three perspectives: the benefits of joint text–chart representation of data, the existing tool support for synthesis of such representation, and the use of natural language processing in visualization systems to coordinate text and charts.

2.1 Symbiotic Relationship of Text and Visualization

The synergistic association of text and visualization is essential for telling compelling stories with data [7, 21]. Traditional analytical visualizations focus on rapid analysis and exploration of data. They leverage perceptually effective channels to encode data, but usually do not involve textual elements. On the other hand, explanatory visualizations emphasize conveying messages to a broader audience—text plays a crucial role. Existing research indicates that people tend to allocate significant attention to the text while using these visualizations [11].

Several existing studies have investigated the benefits of combining text and visualizations. Zhi et al. [67] studied the impact of visualization–text linking—visual marks in a chart are highlighted when hovering over a relevant text phrase. In a controlled experiment, they found that participants recall information better when it is interactively linked across both representations, although they did not observe improvement in comprehension. In contrast, Barral et al. [32] evaluated a gaze-driven approach—rather than explicit linking [67]—and found that it improves comprehension for low-literacy participants [9]. When studying the impact of explicit visualization–text linking in the context of a Bayesian reasoning problem, Ottley et al. [46] found that people tend not to consolidate information well across the text and visualizations, suggesting the need for a better support of content integration.

Based on this empirical evidence, our aim is to technically support the creation of interactive references between text and charts. Unlike the specific examples of previous studies, we investigate full-fledged authoring support with automatic suggestions in an interactive document editor.

2.2 Systems for Authoring Explanatory Visualizations

For communicating complex data, data-driven storytelling—which combines text and visualization into engaging visual data stories—has been advocated [29, 38, 55]. Unlike analytical tools that focus on rapidly

generating data visualizations, tools that help create such data stories allow users to add textual descriptions and annotations, as well as to customize visual marks and layouts [16, 40, 59, 65]. Some of these tools focus on the production of a single narrative visualization [2, 25, 26, 39, 49, 50, 52, 66] or addition of direct annotations [48]. Others support creating a complete narrative with a sequence of visualizations and textual explanations [4, 13, 19, 24, 42, 51]. Existing systems also explore novel forms of data stories including videos [4], comics [24], and slideshows [13, 19, 51].

However, existing storytelling tools do not support the interactive synthesis of text and visualization. In these tools, textual parts are mostly considered passive supportive elements—semantically connected yet separated from the associated visualizations. Therefore, practitioners rely on programming frameworks (e.g., D3 [12] or Idyll [15]) in order to create interactive references between the two (see an interactive article about Boston’s subway system¹). But this requires advanced programming skills. Research prototypes showcase the benefit of visualization–text integration using specific examples [8, 30, 32, 58], but do not support the authoring process. Scientific publishers such as *Authorea* [1] and *Elsevier* [45] attempt to support this integration, but are limited to very simple linking (e.g., finding a related figure given an explicit figure reference in text). Kong et al. [28] recently used crowdsourcing to reconstruct references between textual phrases and visual marks on the charts in existing documents. Latif et al. [35, 36] suggest a programmatic, but declarative syntax using HTML attributes to establish interactive references.

In contrast to existing tools and programming frameworks, we target users who do not have programming expertise and aim to provide an accessible user interface for creating interactive links between text and charts. A recent work [60], published while our work was being reviewed, provides a comparable interface, but the linking method is purely image-based (e.g., color-based region selection). Our approach leverages underlying data semantics to enable a more expressive linking strategy with intelligent data-driven assistance.

2.3 Natural Language Understanding in Visualization

Our work leverages natural language processing to automatically infer potential references between text and charts. Natural language processing has already been used to interact with visualizations, for instance, for assisting data analysis and exploration tasks [57] or question-answering about statistical charts [22]. Only a few systems use it for establishing referencing between text and charts. Among these, *Elastic Documents* [8] extract words from a document and perform simple keyword matching to create interactive links to generated charts; Kim et al. [23] offer a similar method for tables. Similarly, Lai et al. [31] automatically annotate raster images of bar, pie, and scatter plots using user-generated text. Provided an entity name and its characteristics, they identify the relevant visual area using keyword matching.

In contrast, we focus on establishing references between existing text and charts in a *communication* context. Although *Elastic Documents* [8] and Lai et al.’s work [31] tackle a similar problem, they are limited to template-based matching for point-level references and a small set of charts. We aim to support a variety of charts and address interval-based references in addition to point references. Our system also generates higher-order groupings of the point and interval references following the syntactic parsing of a sentence (Fig. 3).

3 UNDERSTANDING TEXT–CHART REFERENCES

To understand how text and charts are referenced in data-driven documents, we analyzed a collection of news articles and scientific papers that contain text–chart pairs.

3.1 Data Collection and Analysis

We started with an initial collection of documents from Kong et al. [28]. It was limited to bar charts alone from 18 articles gathered from different news media. We expanded this collection with additional chart types and sources, resulting in 77 articles with 110 paragraph–chart

¹<http://mbtaviz.github.io/>

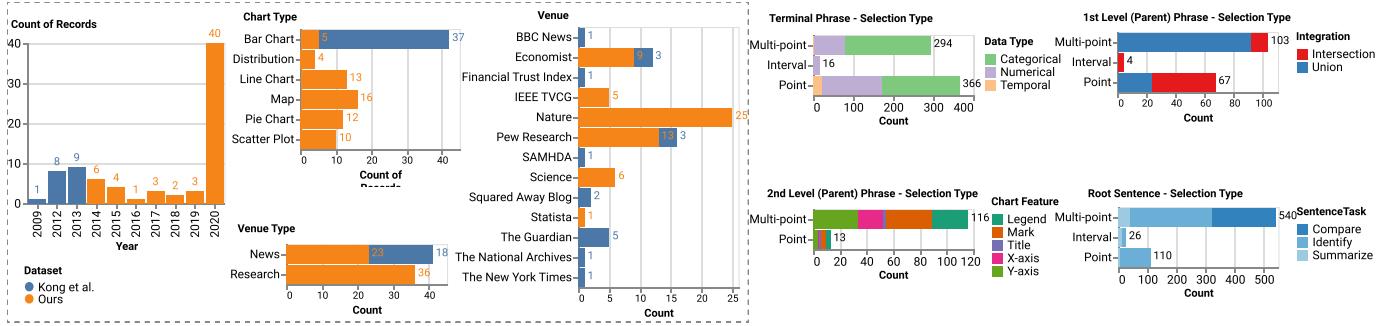


Fig. 2. Overview of our data collection. We augmented existing articles from Kong et al. [28] with additional venues and chart types (left). We observed point, multi-point, and interval selections in the text–chart references (right). They are often grouped together to generate aggregate selections, while also forming a hierarchical reference tree: *terminal phrase* → *1st level (parent) phrase* → *2nd level (parent) phrase* → *root sentence*; see Fig. 3 for an example.

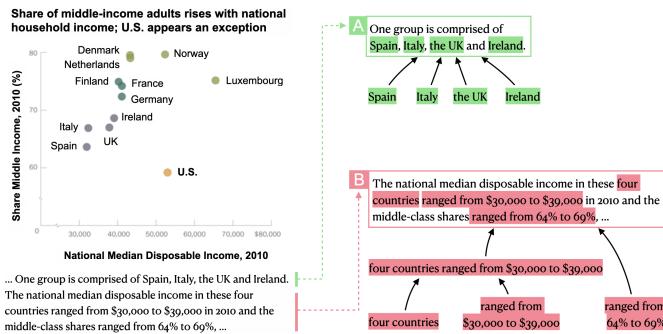


Fig. 3. Excerpt of an article from Pew Research [27]. Demonstration of text–chart reference grouping: The colored text phrases are minimal references, while **A** and **B** show how they can be grouped hierarchically.

pairs. We initially targeted three main sources: visualization papers, research articles published in *Nature*, and news articles. Within these sources, we manually picked examples for maximizing the diversity of charts. Fig. 2 shows the distribution of examples in our collection.

We divided each paragraph–chart pair into sentence–chart pairs for our analysis, resulting in 227 pairs. Following the same process as Kong et al. [28], we manually constructed *minimal* references between text and charts. A reference is minimal if it cannot be subdivided into meaningful smaller references. Two researchers followed an open coding process to analyze the sentence–chart pairs. The researchers used the collection of Kong et al. [28] as a basis to derive the initial codes. These codes were then applied to our collection. The resulting collection is an extended version that contains additional codes and greater diversity of publication venues and chart types.

3.2 Analysis Results

We report the results along five semantic groups of findings, building on each other. The focus of our analysis is always how text, data, and visualization interact with each other.

References resemble selections. A key insight is that every text–chart reference bears a similarity to a selection operation in a visualization. Just as a visualization offers *point* selections (such as clicking one or more visual marks) and *interval* selections (such as brushing a region of visual marks [54]), a text reference can describe one or more visual marks by directly referencing item names or an interval of data points by mentioning scale extents. For instance, in Fig. 3A, each country name in the phrase “Spain, Italy, the UK, and Ireland” selects each corresponding mark in the scatter plot, while the phrase “ranged from \$30,000 to \$39,000” in Fig. 3B refers to all marks falling into the range.

Out of 676 identified minimal references, we observed 366 point, 294 multi-point, and 16 interval references. The referenced chart features mainly included axes (283), individual visual marks (262),

and legends (109); references to axes and legends eventually lead to selecting a set of visual marks in the chart area. While the underlying data values of the referenced chart features are mostly categorical (407), numerical (244) and temporal (25) data values were also referenced. The sixteen interval references relate to 15 numerical and one temporal data attribute. Interestingly, we observed only four (out of 676) *visual* references such as “red arrow” or an “orange slice”. This is maybe because visual encoding in standard charts is rather self-explanatory.

References are grouped hierarchically. We found that minimal references can be grouped together to create higher-order references. The grouping is similar to how a parse tree represents the syntactic structure of a sentence. In a parse tree, constituents—a word or a phrase—are terminal nodes that serve as independent units, e.g., verbs, nouns. A minimal reference is a terminal text phrase in our reference tree that establishes an independent connection to a chart feature. For instance, in Fig. 3A, each country name is a minimal reference and can be combined with others to create a parent reference of the four countries. Eventually, the whole sentence lies at the root, referring to all the countries (Fig. 3A). On the other hand, the sentence in Fig. 3B has three minimal references: “four countries”, “ranged from \$30,000 to \$39,000”, and “ranged from 64% to 69%”. The first two can be integrated to create a parent or non-terminal reference node. This parent node can then be combined with the third minimal reference “ranged from 64% to 69%”, resulting in the root sentence (Fig. 3B). When references are combined, the leftover text—which is not part of any minimal reference—should be added to the parent reference, which otherwise becomes fragmented. That is, the final sentence contains all text rather than only the text that represents minimal references.

In our collection, we observed up to four levels of reference groupings, with the highest-level as root reference being the sentence itself. The minimal references lie at level 0 (leaf nodes in the parse tree). There were 175 first-level and 28 second-level ancestor references, with an average of 2.48 and 2.68 minimal references, respectively. While 131 sentences had at least first-level references, only 27 (out of 227) sentences had second-level references. At non-terminal levels, we observed more multi-point references compared to single-point references—103 versus 67 at the first level and 25 versus six at the second level. We observed four interval references at the first level and no interval reference at the second level; this might partly be because we coded it as *point* or *multi-point* when an *interval* reference was combined with either of them.

Reference grouping incurs data transformations. The grouping of multiple references results in either *union* or *intersection* of individual selections of data points. For instance, when the phrases “ranged from \$30,000 to \$39,000” and “ranged from 64% to 69%” referring to two different variables are combined, the resulting reference relates to the intersection of data points identified by the two ranges. On the other hand, grouping of “Spain”, “Italy”, “UK”, and “Ireland” results in a union of the corresponding data points. We observed more unions than

intersections: 115 unions versus 59 intersections at the first level. The relative difference increases as we climb up the reference hierarchy: 27 versus four at the second level and 25 versus one at the sentence level.

References relate to visualization tasks. A text phrase in a reference can relate to standard visual analysis tasks [34]. All minimal references represent *identification* tasks (e.g., identifying countries in Fig. 3A), while representative visual marks of these tasks vary from a single-point (361) to multi-points (315). References at a higher level describe advanced tasks. Within the first-level references, we identified *comparison* (33) and *summarization* (12) tasks, in addition to *identification* (130) tasks. On the second level, we observed *comparison* (11) tasks, in addition to *identification* (20) tasks. Overall, at the sentence level, we observed 149 *identification*, 49 *comparison*, and 29 *summarization* tasks. An example of a *comparison* sentence looks like “*Some upper-middle-income countries, like the Dominican Republic and Thailand, seem to have deadlier roads than much poorer places such as Liberia*” [63], while an example *summarization* sentence is “*But when countries reach a GDP [...] of about \$30,000, death rates usually start to come down*” [63].

Ambiguities, transformations, and abstractions exist and often relate to data. As can be expected when dealing with natural language, we observed ambiguities in the text–chart references. Almost half of the reference phrases (311 of 676) exactly matched the labels in charts, and hence, were clear. Eighty-two text phrases were partial matches and ambiguous. For instance, the phrase “\$1 increase” in a sentence partially matches to the chart label “*assuming a \$1 increase in the minimum wage*” [62]. Other reference variations include inferences (64)—e.g., “*former communist states*” in text while the chart shows explicit state names. In our collection, the inference problem was more common among the references that were solely inferred from the visual encoding because the corresponding charts did not have textual data labels. We also observed potential ambiguity issues such as the use of synonyms (64), stemming/lemmatization (29), abbreviations (8), and hypernyms (1). For the phrases referring to numerical ranges, we frequently observed approximate numbers (21), especially for large numbers or numbers with decimal points. Another observation is that text phrases may also represent derived measures such as mean, variance, or other computed numbers (17)—e.g., “*Nearly six-in-ten (58%) in the U.S.*” [61], which requires a transformation of the underlying data to be identified. Charts sometimes contain annotations showing these measures, but often they do not.

3.3 Limitations

Although our collection is much more diverse and expansive compared to the initial dataset of Kong et al. [28], the size of the sample and its representativeness are still limited, and it does not include advanced and coordinated multiple visualizations. Our qualitative analysis method could benefit from computational linguistic methods that may further discover semantic and structural insights into the space of text–chart references. We leave this for future work.

4 DESIGN CONSIDERATIONS

Based on the findings of our design space analysis, we designed Kori, an authoring tool, with a focus on establishing explicit referencing between a chart and related text. The main objective is to facilitate non-technical users in creating customizable and highly interactive data-driven documents. The system should assist users by detecting and suggesting potential references while they are composing a document. Besides, the system should provide an intuitive and easy-to-use interface for the manual construction of references beyond the automatic suggestions. We developed Kori along the following design rationale.

D1. Suggest possibilities. The system should not explicitly create references, rather suggest them to the user. One reason is that every automatic detection system would inevitably result in false positives as the matter is complex and the linking might be subjective to the user. Another reason is that users may get annoyed by the automatic creation of many references—not all wanted. Therefore, the system should only suggest them, and the user can either accept or reject them.

D2. Let users create. Users should not feel restricted to only what is suggested by the system—the automatic suggestion of potential references may not be enough. Users may want to create additional references or combine the suggested ones into a single higher-level reference. The construction of a reference involves the selection of visual marks (data items) on the chart and relating them to the text. To this end, our goal is to support smooth visual interactions for creating references. The interaction design should be intuitive, efficient, and generalizable to multiple chart types.

D3. Assist but do not distract. In general, the users should not get distracted by the additional features the tool provides. Users should be able to focus on creating the content while assistance for linking text and charts blends in smoothly. It might go unnoticed first but become a valuable tool when revising and polishing the document. The suggestions and options to create references might even inspire the authors to communicate a deeper analysis of the data as an understandable communication is easier to achieve.

5 KORI USAGE

Kori comprises an *editor* and a *viewer*. We discern two roles of users: *authors*, who create the content within the editor, and *readers*, who consume the content in the viewer. While Kori assists authors in creating interactive references, the readers profit from an improved synthesis of text and charts leveraging those interactive references. In the following, we describe a usage scenario that illustrates both perspectives, before Sect. 6 discusses the technical details. For the sake of demonstration, we take an example dataset and a collection of charts that describe the Covid-19 cases in the US (on federal and state levels) from January to late August 2020. Let us assume that Alice writes an article for Bob.

Authoring. Fig. 1 shows the editor interface of Kori. It consists of a chart gallery and an editing area. The chart gallery holds a collection of charts that can be inserted into the editor ①. The editor window provides a standard text formatting toolbar at the top. Charts can be dragged from the gallery to the editor.

Alice first wants to give an overview of the temporal development and adds the line chart (state level) to the editor ② and then starts writing text. While typing, she gets automatic suggestions (e.g., New York). Special decorations—dotted gray underline—notify her about the suggestions. Curious, she hovers over the suggestion “New York” to preview it. As a result, the line representing New York gets highlighted ②. Unsure about the spelling of Illinois, she types the ‘@’ symbol to trigger available suggestions while creating a reference for Illinois ③. The small chart avatars in the suggestion panel notify her about what chart the suggestion corresponds to. Alice observes an interesting pattern about cases dipping toward the end of April and wants to create a reference for that. She does so by first selecting the text phrase and activating the reference construction mode ④. Then, she selects the line chart in the *link setting* panel ⑤, chooses the filtering mode ⑥, adjusts the date value of the interval slider, and finalizes her reference construction.

Moving forward, she wants to provide a comparison of the three states that were hit hard by the pandemic during the second wave. She sees a suggestion for each state name, but she wants to highlight them at once to allow comparison. She simply does so by following similar steps as ④ and ⑤ but this time choosing the direct manipulation ⑦ mode (see Fig. 4B), and simultaneously selecting three corresponding timelines using a multi-point selection on the chart ⑧. Since these references are manually constructed, they are underlined with blue color.

To give an impression of the severity of the pandemic, she adds a map of total deaths to the editor ⑨. As she describes the states suffering more than 10,000 causalities, she gets a suggestion ⑩. She previews it to see whether it is correct and then accepts it. As she previews it, she realizes the opacity of faded-out regions was quite low, and they were hardly visible. She then adjusts the inactive opacity as desired ⑪.

Reading. Bob reads the composed article in the viewer. It is a restrictive version of the editor and offers a reading interface where Bob can

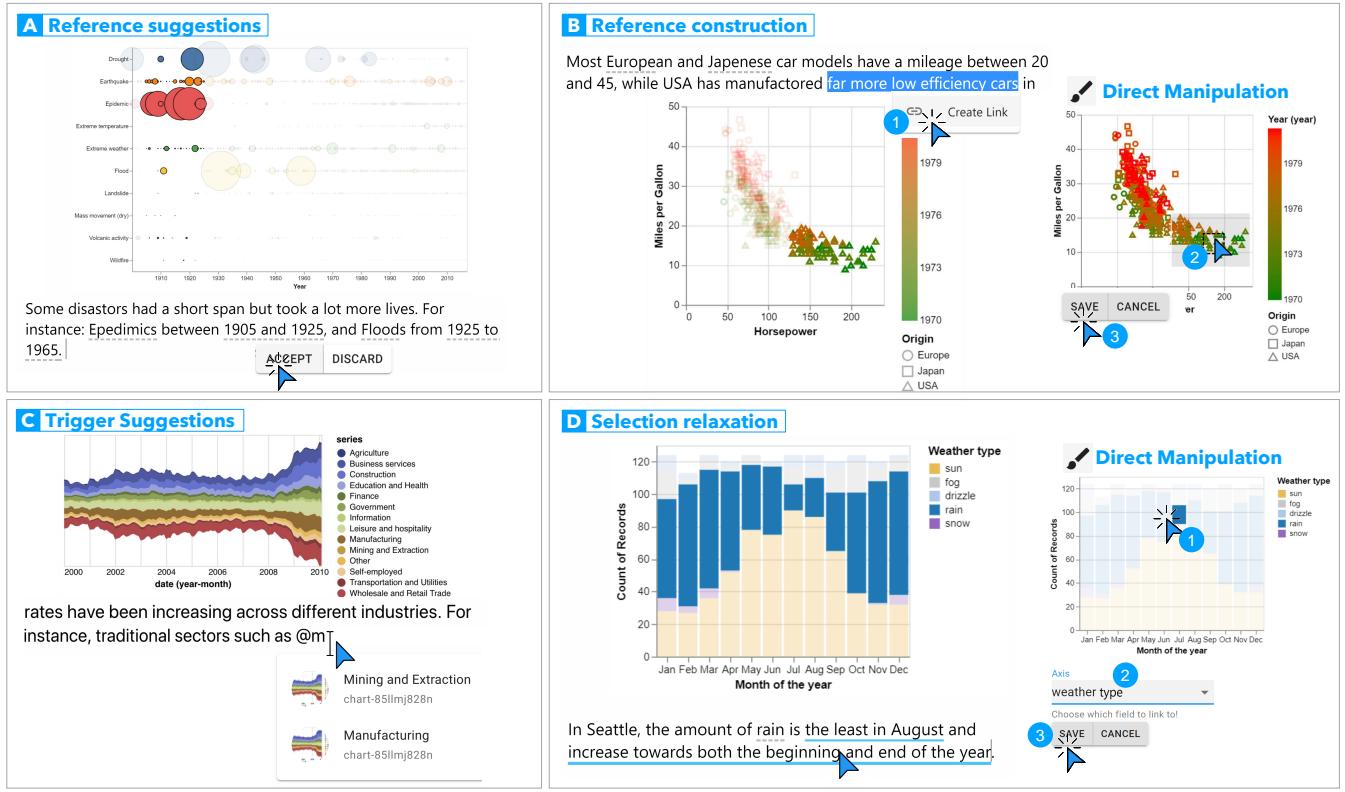


Fig. 4. Salient features of Kori. (A) It uses natural language processing to suggest potential references between charts and the text while a user types. (B) Users can construct references by directly manipulating the visual marks on the chart. (C) It is possible to manually trigger suggestions. (D) In direct manipulation mode, users can easily expand their current selection to multiple visual marks of the same type. The encircled numbers mark the sequence of interactions for each activity.

activate the explicit referencing by interacting with the reference text. We use visual highlighting to make the related parts of a chart stand tall. The default highlight scheme is the opacity channel.

6 THE KORI SYSTEM

Kori is a web-based system. The front-end is developed in JavaScript, React, Draftjs, and Vega-Lite, while the back-end is implemented in Python using Flask. The natural language processing tasks have been performed in Python using SpaCy and FastText [43].

6.1 Charts

We support a wide range of chart types, including but not limited to (stack or group) bar charts, (multi) line charts, scatter plots, distribution plots, heat maps, and choropleth maps. However, advanced visualizations (e.g., network diagrams, treemaps, etc.), as well as charts with coordinated views, are not supported. Similarly, interactive charts are not supported as existing interactions might conflict with our reference construction mechanism. We rely on Vega-Lite [53] for constructing and modifying charts as it offers an expressive and declarative syntax. Authors can load their data into Voyager [65] or Vega-Lite editor [3] to construct and export charts. These charts can then be imported to Kori by dragging the specification files and dropping them into the gallery.

6.2 Reference Detection

We define text–chart reference as an explicit linking of a text phrase to corresponding visual marks. Since both representations correspond to the same data items, our text–chart links are instantiations of conceptual cross-referencing between the charts and text. Kori supports *point*, *multi-point*, and *interval* references.

An automatic reference detection intends to speed up the composition process. We detect all three types of references and group references by combining numeric intervals with the corresponding axes of

the chart. Once the references are identified, their presence is communicated to the author via visual cues without interrupting the authoring process (D1, D3); they are underlined with a dotted gray line. Authors can inspect them in due time or safely ignore them as they would not appear in the viewer mode. Authors can preview them by hovering over before accepting or discarding (Fig. 4A). Once accepted, they are shown with blue underline and will appear in the viewer as interactive references. Fig. 5 shows our automatic reference detection approach that consists of the following steps:

Chart Feature Extraction. Charts are composed of different types of encoding that map data values to visual properties (e.g., position, color, shape) of marks. The first step is to extract these data properties (e.g., axes labels, axes values and scales, legends, data labels) of a chart. The expressive JSON syntax of Vega-Lite enables easy access to this information. We loop through all kinds of visual encodings and extract their values in the underlying data. We also extract the axes properties as this is particularly relevant for interval references. The extracted features serve as a knowledge base to match the user-typed text against and for suggesting potential references. For instance, the extracted features of the scatterplot in Fig. 4B are: x and y axes (*horsepower* and *miles per gallon*), legend categories (Europe, Japan, and USA), and name of all car models (denoted by each dot).

Point-level Matching. The next step detects the occurrences of chart features in a user-typed text. Our matching process uses vector representations of text obtained from FastText [43], a neural network-based approach to obtain text representations. It takes into account both word- and subword-level information (and therefore more resistant to noise than word-only approaches such as word2vec [44]). In addition to exact keyword matches, this vector-based matching process can tackle typographical errors, slight variations of the words or phrases (e.g., US, U.S., USA), abbreviations (e.g., EU, European Commission), synonyms (e.g., donating, contributing), and semantically similar words (e.g., Obama,

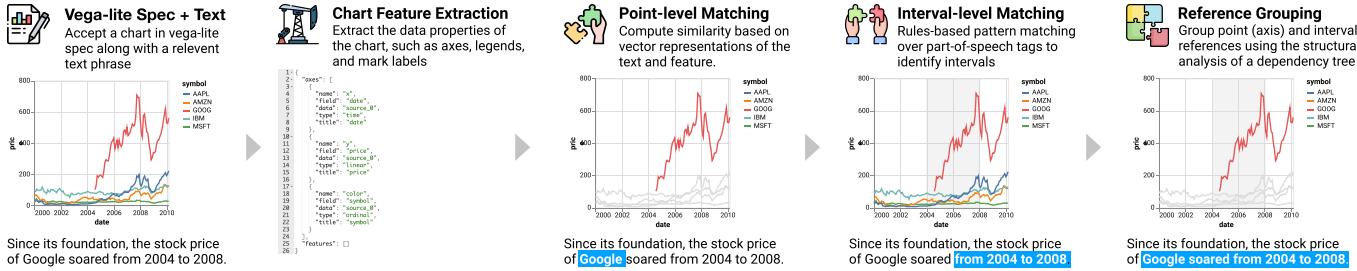


Fig. 5. Four stages of our automatic reference suggestion pipeline. It accepts text and Vega-Lite specifications as input and begins with extracting features of a chart. These features are then matched against user text to find point references. The third step identifies numerical intervals in the user text. Finally, related references are grouped together to form higher level references.

Democrat). We first use FastText to obtain vector representations of (i) chart features and (ii) n -gram representations in the sentence up to $n = 5$. For each chart feature, we then select the n -gram in the sentence that had the highest cosine similarity to the chart feature and present it as a potential link to the user if the threshold is greater than 0.5. Both the n -gram size and the similarity threshold were selected empirically to maximize the F_1 score against a small set of 55 manually-annotated examples (see Fig. 7).

Interval-level Matching. A rules-based approach processing the words and part-of-speech tags is used to identify intervals in a sentence. We derived heuristics based on our observations in the design space analysis (Sect. 3). We observed 116 sentences (16 from our collection, 100 gathered from diverse news articles on the web) that described one or more numerical intervals each and found the following list of frequent patterns (their frequencies in brackets):

more/less/fewer than X (45), X to/through Y (29), between X and Y (15), at least X (11), since/from X (4), below/above X (3)

The symbols X and Y denote either a number, date, or time, which is identified using the parts-of-speech tag NUM from Spacy. The combination of words and part-of-speech tags makes it convenient to derive compact rules to capture intervals. For instance, the rule ‘between X and Y’ is identified with the part-of-speech pattern ‘NUM CCONJ NUM’, which identifies phrases where two numbers (NUM) are joined by a coordinating conjunction (CCONJ); a complete list of patterns is contained in the supplemental material.

Reference Grouping. The final step is to combine the interval with the correct axis-reference on the chart. In some cases, we can infer this by comparing the interval occurrence to chart axis values, for example, in simple charts with a single numerical axis. However, if a sentence contains multiple intervals or a chart has multiple axes with overlapping numeric scales, inferring the correct interval–axis combination is not trivial. Surface-level heuristics that, for example, combine an interval with the nearest axis reference are often inadequate as they do not take into account the structure of the sentence. To better account for sentence structure, we use its dependency tree (again obtained from Spacy) to map interval occurrence to their corresponding axis references. Concretely, we map an interval occurrence to the axis reference that is closest in the dependency tree distance, where we treat the dependency tree as an undirected graph and use Dijkstra’s algorithm to compute the distance between words. For cases where axis reference and/or intervals consist of multiple words, we compute the tree distance between the phrase head words. As an example, in Fig. 6 (top), the sentence contains two axis names ($A_1 = \text{minimum temperature}$, and $A_2 = \text{maximum Temperature}$) of a scatter plot and two intervals ($I_1 = \text{between } 5 \text{ and } 10$ and $I_2 = \text{between } 10 \text{ and } 15$). The distance between A_1 and I_1 is four (orange and blue arcs) and, between A_1 and I_2 , it is five (orange and green arcs). Vice versa, the distance between A_2 and I_1 is five and, between A_2 and I_2 it is four. Combining an axis and interval with minimal distance, A_1 is paired with I_1 and A_2 with I_2 , as desired. In case of ties, we use the distance to the first common ancestor as a tiebreaker (i.e., the interval–axis combination that shares a closer ancestor is grouped together). If this second heuristic results in a

tie, too, then we resort to the surface-level distance—number of words between two candidate phrases—as the final tiebreaker. The approach works in many cases but has limitations—the bottom sentence in Fig. 6 highlights a failure case where our approach wrongly groups the axis price with the interval between 2006 and 2008 (distance 2 – orange and blue arcs) instead of combining it with over \$400 (distance 3 – orange and green arcs).

6.3 Reference Construction

Often, authors already have references in mind while writing about a chart. Thus, they may not have to wait for reference suggestions. Instead, they can explicitly trigger suggestions by typing the '@' symbol (Fig. 4C). It shows a list of features or labels of charts that are currently active in the editor. Each item in the suggestion list is preceded by a thumbnail of the related chart; this helps authors quickly see what feature refers to which chart. The list offers an auto-complete; it keeps on filtering as the user types. The list of suggestions is also triggered by selecting a portion of text.

The reference suggestions are limited and may not cover the full spectrum of possibilities that an author needs. Authors may want to reference a certain part of the chart, a few distinct visual marks that seem interesting in some context, or combine arbitrary sets of visual marks related to a message they are communicating. Kori offers a smooth interface to accomplish this using two distinct modes as shown in Fig. 14 – 6 and Fig. 4B.

The first is the *direct manipulation mode* (Fig. 4B). Authors can select a portion of text ① and then directly brush visual marks of interest ②. The system enables only meaningful selections for a chart (e.g., no rectangular brushing for maps). Kori also offers point and multi-point selections. Having selected a set of visual marks, authors can finalize the reference construction ③. The selection of visual marks is enhanced through relaxation of a selection [20]. Relaxing a selection is a way of saying “*select all items like this one*”. This is particularly useful in situations where users want to reference many visual marks that are of the same type as a single or couple of selected marks. Fig. 4D explains an example where an author likes to create a reference to data dimension weather type = rain; she can simply do so by selecting any blue rectangle (rain) ① and relaxing the selection to weather type by selecting it in the *Axis* (data dimension) drop-down ②.

While the direct manipulation mode is natural, it is not that flexible, especially for selecting overlapping marks or precisely combining multiple data dimensions (shown in a chart). Complementing this, the *filtering mode* (Fig. 14 – 6) allows setting a filter for every data dimension used in a chart. Values of the dimensions can be adjusted using a multi-selection search field for categorical variables or an interval slider for numerical ones. A combination of multiple filters corresponds here to the creation of a higher-level reference using intersection operations (see also Sect. 3).

Kori uses opacity as the default visual highlighting scheme when previewing the interactive references as opacity does not mostly interfere with the existing visual encoding. Each chart is provided with a configuration panel—like the one in Fig. 10—where the degree of

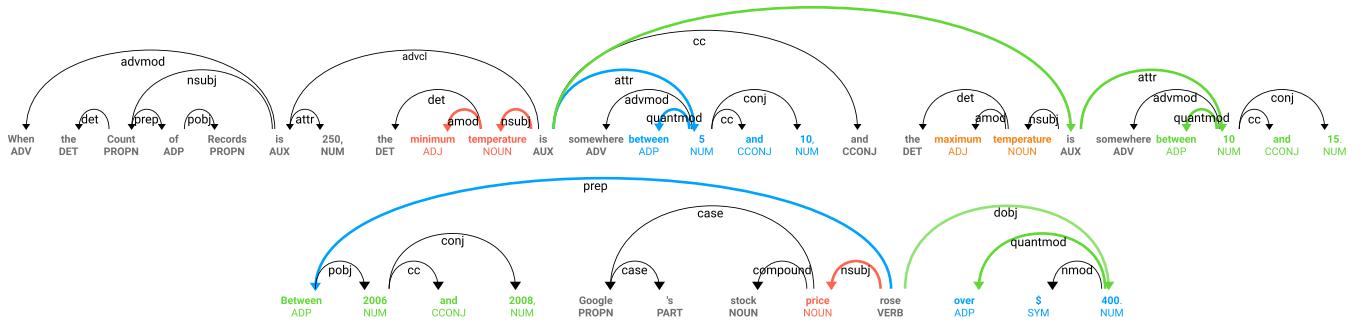


Fig. 6. Dependency parsing of two sample sentences. Success case: minimum temperature is successfully grouped with between 6 and 10 (top). Failure case: price is wrongly grouped with between 2006 and 2008 as it closer to price than over \$400 (bottom).

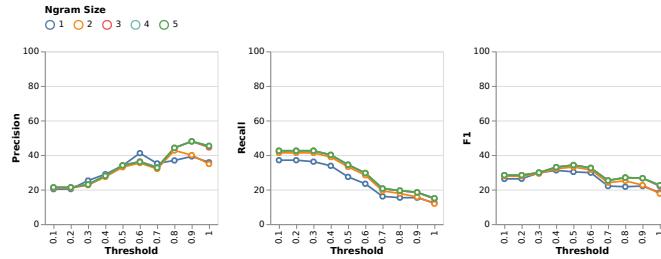


Fig. 7. Quantitative evaluation of the reference detection approach. Precision, recall, and F_1 at various maximum n -gram sizes for the validation dataset (55 text–chart pairs).

opacity can be modified. Also, users can choose the fill highlighting; we provide a color picker to select colors for active and inactive marks.

7 EVALUATION

To evaluate our approach, we tested the automatic reference detection pipeline and conducted a user study.

7.1 Algorithmic Evaluation: Reference Detection

To quantitatively evaluate our reference detection approach, we needed pairs of Vega-Lite chart specifications and corresponding text. Although we had collected 110 text–chart pairs (Sect. 3), we did not have access to the underlying data for charts in all those examples. Therefore, we curated a separate dataset for evaluation.

7.1.1 Dataset Curation

We had data for 42 of 110 text–chart pairs and their images (Kong et al.’s [28] collection). We re-constructed these charts in Vega-Lite. However, they were all bar charts, and included very few interval references. To increase the diversity and size of the sample, we added 50 pairs as follows: From our data collection (Sect. 3), we extracted 116 instances of sentences—including interval references—describing a variety of different charts. Additionally, we collected 34 diverse charts from example galleries of different visualization libraries (e.g., Vega, Vega-Lite, D3, Observables, etc.). Then, we mapped these charts to instances of the 116 sentences. One co-author manually rephrased the sentences to match the data on the chart yet keeping the essence as close to original sentences as possible. Another co-author, then, went through these sentences to make sure they are both syntactically and semantically correct. Finally, we annotated each text–chart pair for point, interval, and group references to create the ground truth. The resulting dataset includes 92 chart–text pairs with 15 different types of charts. This dataset was randomly split into a validation (60%, 55 pairs) and a test (40%, 37 pairs) dataset.

7.1.2 Results

Fig. 7 shows precision, recall, and F_1 score (harmonic mean of precision and recall) for varying similarity thresholds and maximum n -gram sizes on the validation dataset. We selected the threshold of 0.5 and the maximum n -gram size of 3 to maximize the F_1 score (0.34).

For the test dataset, our pipeline correctly identified 57 references (out of 137 true references), produced 90 incorrect references (false positives), and missed 93 references (false negatives). Kori correctly suggested 42 of 84 point, 9 of 26 interval, and 5 of 27 group references. While our approach worked better for identifying point references ($F_1 = 0.47$), the interval detection ($F_1 = 0.25$) and reference grouping ($F_1 = 0.26$) were more challenging.

When running our approach on Kong et al.’s dataset (42 text–chart pairs), we obtained an average distance ($1 - F_1$; lower values are better) of 0.57 from the ground truth (annotated examples by experts) compared to 0.39 produced by their approach. Although quantitatively, references extracted by Kong et al. are closer to those in the ground truth, we detect references automatically while they rely on user intervention to extract base references (distance to the ground truth of 0.54) and then automatically refine them that reduces the distance to 0.39.

7.1.3 Discussion

Several problems contribute to the lower F_1 scores in our approach. One problem with general purpose pre-trained vector representations of text is that it matches words that are too dissimilar in the context of a chart (e.g., “Obama” matches to both chart categories “Democrats” and “Republicans” with cosine similarity of 0.58 and 0.56 respectively). A similar problem occurs with numbers. Numbers are important in dealing with charts and can be described in different ways (e.g., 12, twelve). FastText had problems matching “60” to “sixty” in the sentence “Most movies have a rating between sixty and 100”, and instead matched “60” to “100”, presumably because Arabic numeral representations of numbers are closer in the vector space. While parts-of-speech tagger of Spacy NUM could identify numbers with units, it has no support for dates (e.g., months, days); they are tagged as proper nouns (NNP). Therefore, intervals like Apr to Jun could not be detected. Some failure cases in a grouping, like the one in Fig. 6 (bottom sentence), can be avoided if we consider the extents, scales, and units of numerical axes in the chart in addition to distances in the dependency tree.

7.2 User Study: Authoring

We conducted a user study to gain insights into the usefulness and usability of our system. We focus on evaluating how people interacted with the system to author an interactive document. We could not follow a comparative evaluation approach because comparative systems (e.g., work of Sultanum et al. [60]) were not available.

7.2.1 Participants

We recruited 11 participants (P1–P11; five male and six female) with diverse backgrounds ranging from undergraduate students (3 – P2, P9, P10) and graduate students (3 – P5, P6, P11) to visualization experts (4 – P1, P3, P4, P8) and a user interface designer (P7). All participants had experience using document editing tools like Word, Google Docs, or similar. All participants mentioned that they had created charts using data science toolkits (e.g., R, Python) or programming libraries (e.g., D3, Vega-Lite). All participants except P10 regularly created charts as part of their job or studies. Eight participants mentioned to have worked with charts in a word processing tool (e.g., Google Doc, Word).

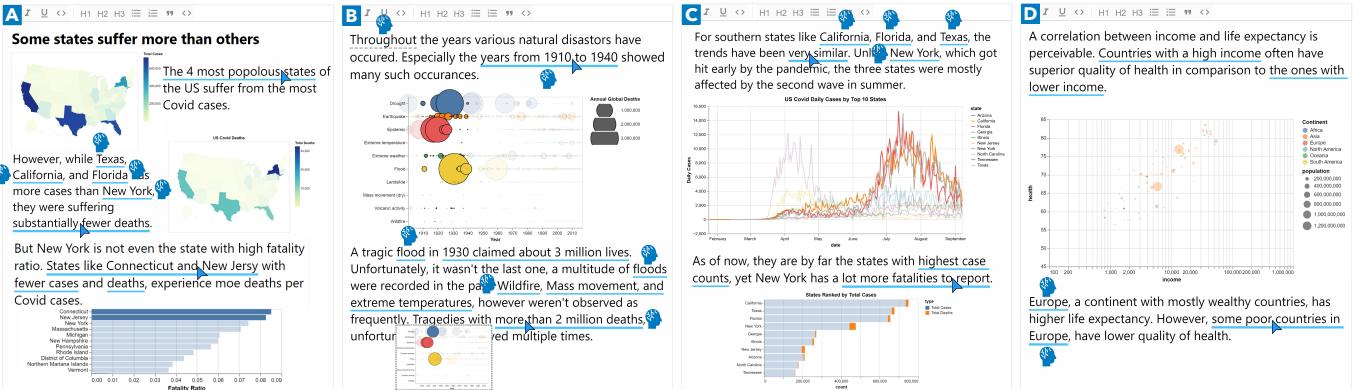


Fig. 8. Copy-edited excerpts of stories created by participants. The symbol marks the reference suggestions. (A) P3 relied on reference construction using direct manipulation . (B) In contrast, P8 got 9 suggestions (1 wrong). She manually constructed 2 references: “1930 claimed about 3 million lives.” and combined two suggestions (“mass movement” and “extreme temperatures”) into a single reference. (C) P5 used first reference “very similar” to verify a fact he was describing. (D) P6 got only two suggestions and created most references using filtering interface.

7.2.2 Procedure and Tasks

In sessions lasting about 60 minutes each, the participants used Google Chrome on their personal computers to access the tool in an online video call with an experimenter, sharing their screens. After collecting the above demographic information, every session began with a brief introduction of the project followed by a short tutorial. We demonstrated the main features of Kori using a variety of chart types.

In the main part of the study, the participants had to complete three tasks. First, to familiarize themselves with the system, we asked them to replicate an example from the tutorial, which contained a bar chart and a paragraph of text with three interactive references. Second, the participants had to reproduce a given but previously unseen example. Two charts (a scatterplot and a heatmap) and two paragraphs of text were provided with nine references. We marked the references in the text, and participants had to transform them into interactive references according to their understanding. We tried to maximize the diversity of references so that participants had to use different features of Kori to construct them. In the third task, the participants had to design a short story (5 to 6 sentences) on one of the three scenarios (they were free to pick any) that were provided to them in the form of one or more charts.

We concluded the session with a reflection survey on the usability and usefulness, rating different statements on a 5-point Likert scale (1 – strongly disagree, 5 – strongly agree). Moreover, we discussed the overall user experience, potential improvements, and limitations in semi-structured interviews.

7.2.3 Results

All participants successfully completed all three tasks within the limited session time. They created references in Task 1 (3 references) and Task 2 (9 references) with minimal intervention from the experimenter. For these two tasks, Kori suggested 6 references (2 for Task 1 and 4 for Task 2) as expected. In Task 3, every participant created a short story with one or more charts; Fig. 8 shows four examples. In total, Kori suggested 64 references for Task 3. (The small boxplots show the distribution of suggestions for all 11 participants). These 64 suggestions also include 25 instances where participants explicitly triggered suggestions. Among these, 48 references (including 25 explicitly triggered suggestions, all correct point references) were correct and 16 incorrect . The incorrectly suggested references were rarely ignored (3) and mostly discarded (13). Besides, the participants manually created 40 references .

For point references, they often relied on automatic suggestions. We observed comparatively fewer instances of explicit triggering by typing ‘@’, but more selecting a text phrase. While participants P3 (Fig. 8 A) and P6 (Fig. 8 D) largely relied on manual construction of references, P8 (Fig. 8 B) made use of more automatic suggestions. We observe the frequent use of direct manipulation mode for geographical maps,

bar charts, and line plots. Comparatively, the participants employed filtering mode more frequently for scatter plots and bubble charts.

Overall experience. Participants highly rated the overall experience (median, mode = 4; IQR = 0;) and usefulness of the tool (median, mode = 5; IQR = 0;). While participants found all features of the tool intuitive and self-explanatory, they rated the reference construction interface (median, mode = 4; IQR = 1;) higher than the automatic suggestions (median, mode = 4; IQR = 0;).

Manual and Automatic Linking. All participants mentioned that the tool is intuitive, easy to use, and has high learnability. Two participants (P3, P4) even said they would have easily discovered all the features without a demonstration. Participants specifically liked the reference construction interface. P4 praised the direct manipulation of regions on the map to construct a reference and described it as easy as “click regions of interest and done. Impressive!”. P5 stated “It is surprising that manual linking is so flexible and works with so many chart types.” P6 appreciated the two different modes of reference construction. P11 favored the filtering mode over direct manipulation, saying it is more flexible and precise. The participant further elaborated: “Finding one data point in millions of data points is just impossible in the brushing interface.” P9 complained about not being able to brush the color legend of a chart (Kori only supports the direct manipulation of marks inside the chart area).

All participants agreed that the automatic suggestions are valuable and complement the reference construction yet criticizing that they are not ‘smart’ enough. P9 stated that automatic suggestions are helpful and work reliably for simple cases. P1 commented “suggestions are cool but not too smart.” Nonetheless, none of the participants considered suggestions as distracting. On the contrary, P2 and P7 suggested making their presence more noticeable. However, wrongly identified references were a bother for one participant (P11). Three participants (P2, P7, P10) complained about delays of automatic suggestions. P8 remarked that the suggestions are often too short, and their grouping needs to be better supported using the interface—the participant wanted to group two references that were correctly suggested in a sentence (Kori does not support grouping of suggested references).

Surprises and Future Directions. Surprisingly, several participants (P5, P6, P9, P11) came up with a different use case that we did not have in mind. They used interactive references as a means to explore the data. P5 described his experience as “I wasn’t sure about a statement”—Task 3: he had an assumption that California, Florida, and Texas have similar trends for Covid-19 (Fig. 8 C)—“and I could verify it using interactive linking. Readers can do too.” P6 reported that interactive linking provides insights and helps in understanding the data. She described her opinion as “I would say it’s not only about writing but also [for] doing analysis while writing.”. Two participants (P1, P5) suggested having an embedded chart creation interface. Without such

interface, it is limited to what could be explored, P1 pointed out and said, “*Charts were limited and non-modifiable. [Kind of] limits the scope of investigative reporting.*”

Several participants highlighted the lack of some convenience features, such as editing a reference that has been suggested or created, deleting a reference without deleting its text, and a user interface to group the suggested references.

Usefulness. Most participants (P1–P6, P11) saw Kori’s value in creating information for reporting, presentation, and communication scenarios. P11 highlighted the benefit of interactive references as “*Readers may not interpret my intention correctly. This tool makes it much clearer by creating explicit references to make sure readers look at what I intended.*” Similarly, P8 mentioned that visual highlighting through interactive links may be better than adding direct annotations on top of a chart, which might increase visual clutter.

8 DISCUSSION

Reflecting on the results of the evaluation, we discuss the limitations of the current solution and directions for future research. This also leads us to a broader discussion on the interplay of exploration and explanation, as well as externalizing cognition through visualization in the context of data-driven storytelling.

8.1 Limitations and Opportunities

Currently, our automatic suggestion does not resolve the user’s intention on which chart to link to. As a result, we observed that it often does not link to the chart expected by the user, especially when the charts share the same underlying dataset. Resolving such ambiguity can be challenging without the user’s explicit input. A potential resolution method would consider the user’s current cursor or the distance from the text to the chart. A related issue is that we currently only support one-to-one mapping, resulting in a unidirectional link from one text phrase to a single chart. However, it is unclear whether supporting one-to-many or many-to-many mapping models is desirable. Not only the authoring interface can become more complicated, but also the benefit for reading may be limited or even adversarial. A user study could shed light on this.

Our user study focused on the authoring interface as it is our core contribution compared to existing research focusing on the reading experience. However, once the links are established, we foresee many opportunities to provide an improved reading experience beyond simply revealing references upon hovering a text phrase. For instance, the user might want to see all relevant text phrases linked to a specific chart for a better understanding of context. To meet this need, the tool can annotate related visual marks in the chart with the relevant text phrases. This is similar to the *gather* operation in exploring embedded word-scale visualizations by Goffin et al. [18], while it is the opposite approach to *Elastic Documents* [8] collecting relevant visualizations for a selected phrase.

A major technical challenge is to make the automatic linking as reliable and accurate as possible. Due to the small size of our dataset, we relied on various heuristics to identify the text–chart links. While some of these already leverage recent advances in natural language processing (e.g., a state-of-the-art neural dependency parser trained over representations from pre-trained language models [17]), others are simpler (e.g., template-based interval pattern matching). Our approach does not learn a joint model that integrates all these heuristics into one. Therefore, it would be interesting to explore a more data-driven approach that trains an end-to-end model from a body of text and its associated charts directly from the raw input. Such a system would require collecting a substantial training set of annotated examples, but may allow for the use of more sophisticated contextualized embedding models (e.g., BERT or Transformers as opposed to static FastText) that can be subsequently fine-tuned on the collected training set.

Integrating further automation is another promising direction. For instance, the generation of text that describes the data is possible [33, 56], and there has also been a flurry of recent work on data-driven table-to-text generation with deep learning techniques [64]. Adapting

such methods, we could suggest prompts about the interesting data facts in the context of chart-to-text generation.

8.2 Closing the Loop from Explanation to Exploration

One interesting insight we learned from the user study is that participants often use the authoring interface to explore data. Kori currently assumes that the users already explored the data and prepared charts for drafting the story. Based on this assumption, Kori does not support chart creation and thus provided pre-made charts to study participants. Since, in the simplified setting of the study, participants did not work with the data before, they used automatically suggested references as opportunities to examine details of the data and also actively inspected the data by trying out different selections in the manual construction interface. This behavior was particularly frequent for high data density in charts without data labels (e.g., a scatterplot with a multitude of data points). Our observation demonstrates a potential need for augmenting Kori to support data exploration, supporting the full lifecycle of data-driven storytelling. An interesting question is how we can support creating additional relevant charts starting from text phrases or existing references. That means, from explanation, going back to exploration and coming forth again.

8.3 Improving External Cognition in Data-driven Articles

Visualizations play a role as an external cognitive aid, offloading the mental load of memorizing and processing data into an external visual representation. Nowadays, visualizations are embedded into diverse media, bringing additional challenges for reading, from efficiently allocating our attention to synthesizing related information across different modes. How we can reduce the cognitive burden incurred from multimodal information is an exciting avenue for future research. In this work, we address the split-attention effect between text and charts primarily through interactive highlighting. According to the cognitive load theory and the cognitive theory of multimedia learning, our highlighting approach follows the signaling principle or temporal contiguity principle [41]. Both theories suggest alternative approaches. For instance, instead of integrating information at the temporal dimension (i.e., through highlighting on demand), we can leverage the spatial dimension as well, reducing the spatial proximity between the text and charts (e.g., using word-sized graphics integrated into the text [10]). Incorporating additional information modalities such as audio comments [37] might also be helpful to further reduce the distance in the information space.

9 CONCLUSION

We developed Kori, a mixed-initiative system to support authors in creating references between the text and charts. We analyzed text–chart references in existing articles to inform the design of Kori. Findings on point and interval references, as well as the grouping of the references, guided the development of the automatic suggestion pipeline. A flexible manual interface provides a complementary way to construct references. An algorithmic evaluation and a user study demonstrate the benefits of Kori for easily creating references with respect to different types of datasets and visualizations, as well as revealed limitations. Future work includes improving the reliability and accuracy of our automatic suggestion pipeline, while also improving the capability of the reference construction interface beyond the unidirectional linking from the text to a chart. Further evaluation of Kori in different system configurations may quantitatively analyze the interplay of manual and automated linking methods. Translating our approach to other forms of data stories such as data comics [24] provides relevant future directions.

ACKNOWLEDGMENTS

We wish to thank the reviewers for their constructive feedback. The project is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 424960846.

REFERENCES

- [1] Authorea. <https://www.authorea.com/>. Accessed: 2020-08-10.
- [2] Datawrapper. <https://www.datawrapper.de/>. Accessed: 2020-08-10.

- [3] Vega editor. <https://vega.github.io/editor>. Accessed: 2021-03-30.
- [4] F. Amini, N. Henry Riche, B. Lee, A. Monroy-Hernandez, and P. Irani. Authoring data-driven videos with dataclips. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):501–510, 2016. doi: 10.1109/TVCG.2016.2598647
- [5] P. Ayres and G. Cierniak. Split-attention effect. *Encyclopedia of the Sciences of Learning*, pp. 3172–3175, 2012.
- [6] P. Ayres and J. Sweller. The split-attention principle in multimedia learning. *The Cambridge Handbook of Multimedia Learning*, 2:135–146, 2005.
- [7] B. Bach, N. Henry Riche, S. Carpendale, and H. Pfister. The emerging genre of data comics. *IEEE Computer Graphics and Applications*, 37(3):6–13, 2017. doi: 10.1109/MCG.2017.33
- [8] S. K. Badam, Z. Liu, and N. Elmquist. Elastic Documents: Coupling text and tables through contextual visualizations for enhanced document reading. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):661–671, 2018. doi: 10.1109/TVCG.2018.2865119
- [9] O. Barral, S. Lallé, and C. Conati. Understanding the effectiveness of adaptive guidance for narrative visualization: a gaze-based analysis. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pp. 1–9, 2020. doi: 10.1145/3377325.3377517
- [10] F. Beck and D. Weiskopf. Word-sized graphics for scientific texts. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 23(6):1576–1587, 2017. doi: 10.1109/TVCG.2017.2674958
- [11] 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 Transactions on Visualization and Computer Graphics*, 22(1):519–528, 2015. doi: 10.1109/TVCG.2015.2467732
- [12] M. Bostock, V. Ogievetsky, and J. Heer. D3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, Dec 2011. doi: 10.1109/TVCG.2011.185
- [13] M. Brehmer, B. Lee, N. H. Riche, D. Tittsworth, K. Lytvynets, D. Edge, and C. White. Timeline storyteller: The design & deployment of an interactive authoring tool for expressive timeline narratives. In *Proceedings of the the Computation+ Journalism Symposium*, 2019.
- [14] J. C. Castro-Alonso, P. Ayres, and J. Sweller. Instructional visualizations, cognitive load theory, and visuospatial processing. In *Visuospatial processing for education in health and natural sciences*, pp. 111–143. Springer, 2019.
- [15] M. Conlen and J. Heer. Idyll: A markup language for authoring and publishing interactive articles on the web. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology, UIST ’18*, pp. 977–989. ACM, 2018. doi: 10.1145/3242587.3242600
- [16] K. A. Cook and J. J. Thomas. Illuminating the path: The research and development agenda for visual analytics. Technical report, Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2005.
- [17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pp. 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1423
- [18] P. Goffin, P. Isenberg, T. Blascheck, and W. Willett. Interaction techniques for visual exploration using embedded word-scale visualizations. In *CHI 2020 - Conference on Human Factors in Computing Systems*. ACM, 2020. doi: 10.1145/3313831.3376842
- [19] S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove, and M. Streit. From visual exploration to storytelling and back again. In *Computer Graphics Forum*, vol. 35, pp. 491–500. Wiley Online Library, 2016. doi: 10.1111/cgf.12925
- [20] J. Heer, M. Agrawala, and W. Willett. Generalized selection via interactive query relaxation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 959–968, 2008.
- [21] N. Henry Riche, C. Hurter, N. Diakopoulos, and S. Carpendale. *Data-driven storytelling*. CRC Press, 2018.
- [22] D. H. Kim, E. Hoque, and M. Agrawala. Answering questions about charts and generating visual explanations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, p. 1–13, 2020. doi: 10.1145/3313831.3376467
- [23] D. H. Kim, E. Hoque, J. Kim, and M. Agrawala. Facilitating document reading by linking text and tables. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, pp. 423–434, 2018.
- [24] N. W. Kim, N. Henry Riche, B. Bach, G. Xu, M. Brehmer, K. Hinckley, M. Pahud, H. Xia, M. J. McGuffin, and H. Pfister. Datatoon: Drawing dynamic network comics with pen+ touch interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 105. ACM, 2019. doi: 10.1145/3290605.3300335
- [25] N. W. Kim, H. Im, N. Henry Riche, A. Wang, K. Gajos, and H. Pfister. DataSelfie: Empowering people to design personalized visuals to represent their data. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 79, 2019. doi: 10.1145/3290605.3300309
- [26] N. W. Kim, E. Schweickart, Z. Liu, M. Dontcheva, W. Li, J. Popovic, and H. Pfister. Data-driven guides: Supporting expressive design for information graphics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):491–500, 2016. doi: 10.1109/TVCG.2016.2598620
- [27] R. Kochhar. Middle class fortunes in western europe. <https://www.pewresearch.org/global/2017/04/24/middle-class-fortunes-in-western-europe/>. Accessed: 2020-08-10.
- [28] N. Kong, M. A. Hearst, and M. Agrawala. Extracting references between text and charts via crowdsourcing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 31–40. ACM, 2014. doi: 10.1145/2556288.2557241
- [29] R. Kosara and J. Mackinlay. Storytelling: The next step for visualization. *Computer*, 46(5):44–50, 2013. doi: 10.1109/MC.2013.36
- [30] B. C. Kwon, F. Stoffel, D. Jäckle, B. Lee, and D. Keim. VisJockey: Enriching data stories through orchestrated interactive visualization. In *Poster Compendium of the Computation+ Journalism Symposium*, vol. 3, p. 3, 2014.
- [31] C. Lai, Z. Lin, R. Jiang, Y. Han, C. Liu, and X. Yuan. Automatic annotation synchronizing with textual description for visualization. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13. ACM, 2020. doi: 10.1145/3313831.3376443
- [32] S. Lallé, D. Toker, and C. Conati. Gaze-driven adaptive interventions for magazine-style narrative visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 2019. doi: 10.1109/TVCG.2019.2958540
- [33] S. Latif and F. Beck. VIS Author Profiles: Interactive descriptions of publication records combining text and visualization. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):152–161, 2019. doi: 10.1109/TVCG.2018.2865022
- [34] S. Latif, S. Chen, and F. Beck. A deeper understanding of visualization-text interplay in geographic data-driven stories. *Computer Graphics Forum*, 40(3):311–322, 2021. doi: 10.1111/cgf.14309
- [35] S. Latif, D. Liu, and F. Beck. Exploring interactive linking between text and visualization. In *EuroVis 2018 - Short Papers*. The Eurographics Association, 2018. doi: 10.2312/eurovisshort.20181084
- [36] S. Latif, K. Su, and F. Beck. Authoring combined textual and visual descriptions of graph data. In *EuroVis 2019 - Short Papers*. The Eurographics Association, may 2019. doi: doi:10.2312/evs.20191180
- [37] S. Latif, H. Turner, and F. Beck. Talking realities: Audio guides in virtual reality visualizations. *IEEE Computer Graphics and Applications*, pp. 1–1, 2021. doi: 10.1109/MCG.2021.3058129
- [38] 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. doi: 10.1109/MCG.2015.99
- [39] Z. Liu, J. Thompson, A. Wilson, M. Dontcheva, J. Delorey, S. Grigg, B. Kerr, and J. Stasko. Data illustrator: Augmenting vector design tools with lazy data binding for expressive visualization authoring. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 123. ACM, 2018. doi: 10.1145/3173574.3173697
- [40] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions On Graphics*, 5(2):110–141, 1986. doi: 10.1145/22949.22950
- [41] R. E. Mayer and L. Fiorella. 12 principles for reducing extraneous processing in multimedia learning: Coherence, signaling, redundancy, spatial contiguity, and temporal contiguity principles. In *The Cambridge handbook of multimedia learning*, vol. 279. Cambridge University Press, 2014.
- [42] G. McNeill and S. A. Hale. Viz-Blocks: Building Visualizations and Documents in the Browser. In J. Johansson, F. Sadlo, and G. E. Marai, eds., *EuroVis 2019 - Short Papers*. The Eurographics Association, 2019. doi: 10.2312/evs.20191177
- [43] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.
- [44] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Dis-

- tributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, vol. 26. Curran Associates, Inc., 2013.
- [45] P. Nowakowski, E. Ciepiela, D. Haręzlak, J. Kocot, M. Kasztelnik, T. Bartysiński, J. Meizner, G. Dyk, and M. Malawski. The collage authoring environment. *Procedia Computer Science*, 4:608–617, 2011.
- [46] A. Ottley, A. Kaszowska, R. J. Crouser, and E. M. Peck. The Curious Case of Combining Text and Visualization. In J. Johansson, F. Sadlo, and G. E. Marai, eds., *EuroVis 2019 - Short Papers*. The Eurographics Association, 2019. doi: 10.2312/evs.20191181
- [47] F. Paas and J. Sweller. Implications of cognitive load theory for multimedia learning. *The Cambridge handbook of multimedia learning*, 27:27–42, 2014.
- [48] D. Ren, M. Brehmer, B. Lee, T. Höllerer, and E. K. Choe. ChartAccent: Annotation for data-driven storytelling. In *2017 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 230–239. IEEE, 2017. doi: 10.1109/pacificvis.2017.8031599
- [49] D. Ren, T. Höllerer, and X. Yuan. iVisDesigner: Expressive interactive design of information visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2092–2101, 2014. doi: 10.1109/TVCG.2014.2346291
- [50] D. Ren, B. Lee, and M. Brehmer. Charticulator: Interactive construction of bespoke chart layouts. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):789–799, 2018. doi: 10.1109/TVCG.2018.2865158
- [51] A. Satyanarayan and J. Heer. Authoring narrative visualizations with Ellipsis. In *Computer Graphics Forum*, vol. 33, pp. 361–370. Wiley Online Library, 2014. doi: 10.1111/cgf.12392
- [52] A. Satyanarayan and J. Heer. Lyra: An interactive visualization design environment. vol. 33, pp. 351–360, 2014. doi: 10.1111/cgf.12391
- [53] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis)*, 2017. doi: 10.1109/TVCG.2016.2599030
- [54] A. Satyanarayan, R. Russell, J. Hoffswell, and J. Heer. Reactive vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):659–668, 2016. doi: 10.1109/TVCG.2015.2467091
- [55] E. Segel and J. 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
- [56] A. Srinivasan, S. M. Drucker, A. Endert, and J. Stasko. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):672–681, 2018. doi: 10.1109/TVCG.2018.2865145
- [57] A. Srinivasan and J. Stasko. Natural language interfaces for data analysis with visualization: Considering what has and could be asked. In *Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers*, pp. 55–59, 2017. doi: 10.2312/eurovisshort.20171133
- [58] M. Steinberger, M. Waldner, M. Streit, A. Lex, and D. Schmalstieg. Context-preserving visual links. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2249–2258, 2011. doi: 10.1109/TVCG.2011.183
- [59] C. Stolte, D. Tang, and P. Hanrahan. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):52–65, 2002. doi: 10.1109/2945.981851
- [60] N. Sultanum, F. Chevalier, Z. Bylinskii, and Z. Liu. Leveraging text-chart links to support authoring of data-driven articles with vizflow. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–17, 2021. doi: 10.1145/3411764.3445354
- [61] The Economist. Where are the world's best english-speakers? <https://www.economist.com/graphic-detail/2019/12/04/where-are-the-worlds-best-english-speakers>, 2019. Accessed: 2020-09-17.
- [62] The Economist. Higher minimum wages are linked to lower suicide rate. <https://www.economist.com/graphic-detail/2020/01/20/higher-minimum-wages-are-linked-to-lower-suicide-rates>, 2020. Accessed: 2020-09-17.
- [63] The Economist. The richest countries have the fewest road deaths. <https://www.economist.com/graphic-detail/2020/01/27/the-richest-countries-have-the-fewest-road-deaths>, 2020. Accessed: 2020-09-17.
- [64] S. Wiseman, S. Shieber, and A. Rush. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2253–2263, 2017. doi: 10.18653/v1/D17-1239
- [65] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):649–658, 2015. doi: 10.1109/TVCG.2015.2467191
- [66] H. Xia, N. Henry Riche, F. Chevalier, B. De Araujo, and D. Wigdor. DataInk: Direct and creative data-oriented drawing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 223. ACM, 2018. doi: 10.1145/3173574.3173797
- [67] Q. Zhi, A. Ottley, and R. Metoyer. Linking and layout: Exploring the integration of text and visualization in storytelling. In *Computer Graphics Forum*, vol. 38, pp. 675–685. Wiley Online Library, 2019. doi: 10.1111/cgf.13719