

Notable: On-the-fly Assistant for Data Storytelling in Computational Notebooks

Haotian Li
The Hong Kong University of Science and Technology
Hong Kong SAR, China
Microsoft Research Asia
Beijing, China
haotian.li@connect.ust.hk

Lu Ying
Zhejiang University
Hangzhou, Zhejiang, China
Microsoft Research Asia
Beijing, China
yingluu@zju.edu.cn

Haidong Zhang
Microsoft Research Asia
Beijing, China
haizhang@microsoft.com

Yingcai Wu
Zhejiang University
Hangzhou, Zhejiang, China
ycwu@zju.edu.cn

Huamin Qu
The Hong Kong University of Science and Technology
Hong Kong SAR, China
huamin@cse.ust.hk

Yun Wang*
Microsoft Research Asia
Beijing, China
wangyun@microsoft.com



Figure 1: This figure illustrates the interface of *Notable*, an on-the-fly assistant for data storytelling in computational notebooks. (a) is an input cell where the chart is specified. The original chart and illustrated data facts are shown in (b), the *plot widget*. The plot widget allows users to modify and select useful data facts. The selected data facts are arranged into a story and displayed in (c), the *organization panel*. The automatically organized story can be further improved by users in the organization panel.

ABSTRACT

Computational notebooks are widely used for data analysis. Their interleaved displays of code and execution results (e.g., visualizations) are welcomed since they enable iterative analysis and preserve the exploration process. However, the communication of data

findings remains challenging in computational notebooks. Users have to carefully identify findings from useless ones, document them with texts and visual embellishments, and then organize them in different tools. Such workflow greatly increases their workload, according to our interviews with practitioners. To address the challenge, we designed *Notable* to offer *on-the-fly* assistance for data storytelling in computational notebooks. It provides intelligent support to minimize the work of documenting and organizing data findings and diminishes the cost of switching between data exploration and storytelling. To evaluate *Notable*, we conducted a user study with 12 data workers. The feedback from user study participants verifies its effectiveness and usability.

*Yun Wang is the corresponding author.

CHI '23, April 23–28, 2023, Hamburg, Germany
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*, April 23–28, 2023, Hamburg, Germany, <https://doi.org/10.1145/3544548.3580965>.

CCS CONCEPTS

• **Human-centered computing** → **Visualization systems and tools; Interactive systems and tools.**

KEYWORDS

data visualization, data storytelling, computational notebooks

ACM Reference Format:

Haotian Li, Lu Ying, Haidong Zhang, Yingcai Wu, Huamin Qu, and Yun Wang. 2023. *Notable: On-the-fly Assistant for Data Storytelling in Computational Notebooks*. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*, April 23–28, 2023, Hamburg, Germany. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/3544548.3580965>

1 INTRODUCTION

Computational notebooks (e.g., Jupyter¹ and RStudio²) have been widely used in exploring data and deriving insights for decision making [39]. Each computational notebook consists of multiple input and output *cells* for code editing and result presentation [5]. The design integrates the code and results in a single interface and thus suits the needs of data analysis: it allows iterative updates of code, facilitates quick inspection of the results [16], and documents the procedure of data analysis for easier collaboration [39].

Though computational notebooks have multiple advantages, they also introduce challenges to users [5]. Since users often explore data iteratively in computational notebooks, lots of intermediate or useless cells may be kept in the notebooks as well [16]. Furthermore, the sequence of data exploration in computational notebooks may not entirely fit the narrative structure of a data story [22]. Therefore, it is necessary for users to select the important findings, document the findings with text descriptions and visual highlights, and organize them into a logically coherent data story for communication [22]. According to our interviews with experienced data analysts, such workflow often require them to use multiple tools (e.g., Microsoft PowerPoint³ and Google Slides⁴), which increases their workload due to (1) *the additional operations to transfer data findings between tools* and (2) *the distraction led by multiple user interfaces with inconsistent appearances and interactions*.

To cater the need for convenient data communication, several existing tools are developed to support data storytelling. For example, Viola [54] and Nbconvert [20] enable convenient format conversion from notebooks to presentations. However, they only support direct format conversion and cannot facilitate other steps in data storytelling, such as story organization. To fill the gap, a recent tool, NB2Slides [69], adopted advanced machine learning techniques to organize an accomplished notebook into slides based on pre-defined templates. However, it assumes that users finish the entire data analysis process before making data stories, which does not align well with the empirical observation of switching between data exploration and story creation [14, 26]. Users can hardly consider data storytelling and exploration comprehensively in such settings. Given the drawback of existing tools, we would like to

explore *how we can provide on-the-fly support to data storytelling during data exploration in computational notebooks*.

To answer the research question, we first attempted to figure out the challenges and desired assistance regarding data storytelling in computational notebooks. We conducted formative interviews with six data analysts with diverse backgrounds and summarized five design requirements of the storytelling tool in computational notebooks. The interviewees frequently mentioned their need for assistance in data finding *documentation and organization*. Furthermore, they mentioned the *inconvenient switching between data exploration and storytelling* when using computational notebooks. Based on their feedback, we designed and implemented *Notable*, a computational notebook extension, to offer on-the-fly assistance to storytelling, as shown in Figure 1. When the users plot a chart to inspect the data, *Notable* can automatically search for potential data facts that may interest the users and illustrate them with visual embellishments and text descriptions (Figures 1(b)). By doing so, the manual efforts of documenting findings can be eliminated. Then the user can select a data fact, and *Notable* organizes the story based on the selected data facts and the new one (Figure 1(c)). The user may customize the data fact and the story organization as well. At the end of data analysis, *Notable* supports exporting the story as presentation slides, one of the most commonly used formats of data stories [18]. To evaluate whether *Notable* is helpful to users, we conducted a user study with 12 participants. The feedback from them demonstrates the usability and effectiveness of *Notable*. Finally, we concluded our research by discussing the lessons learned and potential future directions.

To summarize, the contributions of our paper include:

- The design requirements for storytelling tools in computational notebooks;
- A computational notebook extension, *Notable*, which offers on-the-fly assistance to data storytelling;
- The design lessons and future opportunities learned from the research.

2 RELATED WORK

In this section, we review research on visual analysis in computational notebooks, data storytelling, and data fact recommendation.

2.1 Visual Data Analysis with Computational Notebooks

Computational notebooks have been widely applied in data analysis since it enables the integrated display of code and results, which facilitates the iterative nature of data analysis better [39]. In existing computational notebook environments, visual data analysis is supported by packages for convenient plotting (e.g., Altair [52], Bqplot [2], Plotly [36], and Matplotlib [19]) and tools targeting at visualizing specific machine learning models (e.g., TimberTrek [59] and Calibrate [65]).

The aforementioned plotting packages are widely used but limited to showing the visualizations specified by the user and do not provide additional assistance. Recent studies have been proposed to augment them by providing intelligent assistance to data exploration (e.g., [8, 27, 55, 64, 68]). For example, Lux [27] recommends lists of static visualizations to users without requiring them

¹<https://jupyter.org/>

²<https://www.rstudio.com/>

³<https://www.microsoft.com/en-us/microsoft-365/powerpoint>

⁴<https://www.google.com/slides/about/>

to specify visualizations explicitly. Considering the importance of interactions in visual analysis [66], PI_1 recommends interactive visualizations when users query data from databases [68]. Informed by the advantages of multi-view visualizations over single-view charts [30], PI_2 [8] extends PI_1 by generating multi-view interactive visualizations. Besides recommending visualizations, EDAssistant [29] and LodeStar [38] suggest code snippets for data exploration. Fork It [60] allows users to keep multiple versions of code in notebooks for convenient alternative explorations.

Though these tools assist users in data exploration by recommendation, they still require a considerable amount of effort to communicate the findings, which is an essential stage in data analysis [62]. Existing extensions allow users to directly convert notebooks into other formats for presentation, such as Voilà [54] and Nbconvert [20]. However, they only help format conversion and cannot cover other steps in data storytelling. In our research, we propose *Notable*, a notebook extension, to facilitate data storytelling during data exploration. Despite generating slides for presenting the data story, *Notable* also offers assistance in (1) illustrating data findings with text description and visual embellishments; and (2) organizing selected data facts into stories.

2.2 Data Storytelling

Data storytelling, as an effective way to communicate information in data, has gained more attention in recent years [50]. It serves as the final stage in data analysis and is closely connected with data exploration [62]. To facilitate data storytelling, various types of tools have been proposed. According to the level of automation, they can be roughly classified into three clusters, authoring tools, automatic generation tools, and tools with intelligent support [6].

Authoring tools provide interactive interfaces for users to create data stories freely. For example, Idyll Studio [10] and VizFlow [47] support data article creation by integrating chart creation and article editing in a unified interface. CLUE [14] and InsideInsights [33] allow users to create data presentations with the identified findings in data exploration. ToonNote [21] integrates a data comic authoring interface with computational notebooks. To reduce the considerable manual efforts in creating data stories, fully automatic data story generation tools have been explored. They leverage machine learning techniques to analyze datasets and generate narrative visualizations. Under this category, Wang *et al.* [58], Shi *et al.* [43], and Lu *et al.* [32] automate fact sheet, data video and scrollytelling creation from datasets. Similarly, Roslingifier [45] detects important events in time series data and generates animation automatically. Chen *et al.* [7] attempted to synthesize data stories according to the analytical provenance. InfoMotion [57] creates animated infographics according to the structure of information inside infographics. However, automatic story generation tools limit users' participation in storytelling and thus may result in bias and untrust [28].

To achieve the collaboration of humans and machines, tools that provide intelligent support to data storytelling have gained increasing interest (*e.g.*, [13, 34, 48, 61, 67]). For example, Erato [48] recommends new data facts based on user-selected facts for completing data stories and generating infographics. According to a recent survey [6], most of the existing tools support infographic or

video and animation creation, while making slides receives little attention. According to Hullman *et al.* [18], slides are frequently used for communicating data stories. However, making slides requires non-trivial efforts from data scientists [35]. To assist with slide creation, Zheng *et al.* [69] developed NB2Slides to convert human-made computational notebooks to slides based on pre-defined templates. However, NB2Slides takes a one-way style that generates slides after a complete data analysis session. Such features may not suit the common workflow of data analysts as they often switch between data story creation and data exploration, according to previous research [14, 26]. To better fit the workflow, we propose to offer on-the-fly assistance in *Notable* to support story creation during data exploration. Users are able to take storytelling into consideration when exploring data. Furthermore, our tool is integrated into widely used computational notebooks, which is preferred over new tools, such as Erato [48], as indicated in our formative study.

2.3 Data Fact Recommendation

Since manual visual data exploration requires data analysis skills and enormous effort [1], recent research introduces automatic *data fact* recommendation methods to address this challenge [25]. These methods examine data characteristics automatically and recommend data facts that may interest users, *e.g.*, outliers and trends of data. Data facts are also called *data insights* in other publications (*e.g.*, [12, 49]).

The purpose of recommending data facts can be diverse [25]. For example, QuickInsights [12], Top-K Insights [49], and SeeDB [53] recommend data facts to facilitate a quick exploration of a database before in-depth data analysis. Duet [24] suggests similar data facts when analyzing a specific subset of data, which happens in the data analysis with a clear target. The line of research that inspires our study applies data facts in data communication (*e.g.*, [44, 58]).

Regardless of the purposes, most existing studies above attempt to explore data column combinations and recommend potentially interesting data facts from a dataset. However, most data workers still prefer to plot charts and observe data manually instead of relying on the recommendation [1]. To facilitate their needs, in this paper, we explore how to reduce the workload of data exploration based on the charts created by them. In our formative interviews, we identified that manually documenting data findings in a chart required considerable effort. Users need to write text descriptions and sometimes highlight key data points. To automate this procedure, we propose to use the algorithm of data fact recommendation to infer the potential data facts from the charts that users have created and may feel interested in. Then we illustrate the data facts for users' selection to facilitate data insight interpretation and communication [46]. According to users' feedback, the illustrated data facts can save their time and facilitate storytelling.

3 FORMATIVE STUDY

To derive the design requirements of storytelling tools in notebooks, we conducted a formative study where we interviewed data workers from different domains in a semi-structured manner.

3.1 Interviewees

In our formative study, we recruited six data workers (3 male and 3 female, $Age_{mean} = 27.5$, $Age_{std} = 3.5$, $Experience_{mean} = 4.67$ years, $Experience_{std} = 0.47$ year) by sending invitations through social media. They were from both academia and industry with diverse backgrounds. They had at least 4-year experience in using computational notebooks for data analysis. All of them explored data and communicated data findings frequently (*i.e.*, at least once a week). Their demographic information is in Table 1.

3.2 Procedure

Our study was conducted through one-on-one online meetings. Each meeting began with an introduction to computational notebooks to recall interviewees' experiences. Then we asked for the interviewees' consent to recording the meeting and using their demographic information before the interview. The semi-structured interview study had three parts. In the first part, the nature of the interviewees' data work was enquired to learn their background. For example, we asked them "what is your daily work?" In the second part, interviewees introduced their workflow of communicating data exploration results in computational notebooks. We also asked about their pain points in the current workflow. In the last part, the interviewees discussed how to improve their workflow with us. Each study lasted about half to an hour. The authors took notes during the meetings. After all interviews, the first author organized the feedback according to the notes and video recordings. Then the co-authors discussed and summarized the organized feedback into findings iteratively with the help of recordings. Finally, five design requirements were derived from the findings.

3.3 Findings

All participants agreed that it took considerable effort to make data stories based on findings in exploration. According to our interviewees, when data findings were observed, they first *documented* useful ones with texts or visual embellishments. Then the documented findings would be *organized* into data stories. To present data stories, the interviewees needed to *prepare* slides. We summarize three key findings regarding the pain points and their expected tools.

3.3.1 Manual documenting and organizing data findings. In the interview, all interviewees mentioned the difficulties of **documenting the discovered data findings** during data exploration. First, they had to manually record the data findings and highlight the key data points on the chart. P5 said that he would like to highlight the key findings on the charts but often found that it was challenging to do it by programming. Second, five among six interviewees had to rely on additional tools to document their data findings. P2 took notes using paper, while others used note-taking applications such as Notion⁵ or OneNote⁶. Only P3 used the markdown cells in computational notebooks to record the findings. Due to the two problems, documenting data findings when using notebooks often distracted them from data exploration, which has been mentioned in a previous study [56] as well.

Despite the documentation of data findings, **the organization of findings** was also mentioned by four interviewees (P1-P3, P5) as a pain point. P1 commented that she documented a large number of data findings during the exploration. It was hard for her to remember the relationship between the findings. Therefore, the story organization was highly challenging, which echoed the observation in previous research [22]. She hoped that the data findings could be automatically organized according to their common features. P3 complained that she needed to "spend half of the time in data exploration and another half in preparing presentations", indicating the great effort she took to organize data findings and make slides.

3.3.2 Inconvenient switching between data exploration and storytelling. All six participants commented that they commonly switched between data exploration and storytelling. P5 mentioned "organizing data findings is like organizing my mind". He could notice logical flaws in his data analysis during authoring data stories and then went back to data exploration. P4 also said "I repeatedly iterate between exploration and storytelling" since he could get new ideas when preparing his stories. The feedback reveals that storytelling is bi-directionally connected with data exploration, which aligns with previous observations [14, 26, 62]. Making the story also inspires and guides the process of data exploration.

Five interviewees (P1-P4, P6) complained about the inconvenience of switching between data exploration and storytelling when using computational notebooks. The issue was led by **the usage of multiple tools**. When analyzing data, they explored datasets with computational notebooks and identified useful findings. These findings in exploration were documented using various approaches, such as writing on paper or using note-taking applications. They commonly organized findings and prepared slides using presentation tools, including Microsoft PowerPoint and Google Slides. The interviewees indicated two-fold drawbacks led by the issue. First, it is not easy to transfer data findings between tools. Three interviewees (P1, P2, P5) mentioned that they had to take screenshots of charts and import them to the presentation tool. P6 even needed to copy and paste data between tools manually. The comments aligned with a previous study [3], where the authors describe transferring data findings as a "tedious" process. Second, the usage of different tools introduced additional mental load and led to distraction. For example, P1 felt that her exploration was interrupted when she needed to transfer the findings from computational notebooks to another application.

3.3.3 Design of storytelling tools. At the end of the interviews, the participants were encouraged to describe their expectations about future storytelling tools. All participants welcomed intelligent assistance to address their challenges. Besides, they have two common suggestions. First, four interviewees (P1, P2, P5, P6) emphasized **the necessity of convenient customization** in storytelling tools. P2 worried that existing techniques for automatic data exploration and story generation were only able to mine the apparent data patterns. His team usually looked for the causes of data patterns leveraging domain expertise, and therefore it was important to let users involve in story content creation. P1 pointed out that different users could have different approaches to organizing the findings into stories. P6 further mentioned the necessity of exporting modifiable formats of stories since most companies have requirements on the

⁵<https://www.notion.so/>

⁶<https://www.microsoft.com/en-us/microsoft-365/onenote/digital-note-taking-app>

Table 1: The table records the demographic information of the interviewees in our formative study. We report their genders, ages, jobs, domains, experiences in years, and frequency of data exploration and communication.

ID	Gender	Age	Job	Domain	Experience (Year)	Frequency
P1	Female	26	Postgraduate researcher	Data visualization	4	Everyday
P2	Male	28	Data analyst	Finance	5	Twice a week
P3	Female	26	Research fellow	Computational social science	4	Everyday
P4	Male	25	Postgraduate researcher	Economics	5	Everyday
P5	Male	35	Applied data scientist	Computer system	5	Everyday
P6	Female	25	Data scientist	E-commerce	5	Once a week

appearance of stories. Therefore, it is crucial to enable users’ customization, and the customized story should have higher priority than the automatically organized story. Second, the tools are better to be **integrated with existing tools**, *i.e.*, existing computational notebook environments and commonly used presentation tools. P1 expressed her doubts about an entirely new tool due to the learning curve. She preferred an extension with simple interactions so that she would not spend extra effort on learning the usage of the tool. P4 echoed P1’s opinion by saying “*it will be great if the tool is an extension and can be integrated into widely accepted tools*”.

3.4 Design requirements

We derived five key design requirements of a storytelling tool in the computational notebook according to the findings. In the remaining parts of the paper, **R1-R5** refer to the requirements. The requirements are:

R1. Offering on-the-fly assistance to data storytelling. The storytelling tool should support concurrent data storytelling and exploration to eliminate the cost of switching. With the tool, users can swiftly transfer data findings into story pieces; users can also reflect on their exploration with data stories.

R2. Facilitating data finding documentation. The tool should assist the user in finding documentation. It should reduce the work of illustrating findings with text descriptions and visual embellishments. Furthermore, the illustrated findings should be easily accessed in the tool.

R3. Automating the organization of documented data findings. The organization of documented data findings should be automated to reduce the users’ workload. The organized story can facilitate the quick reflection of data exploration. Users can identify potential logical flaws and unexplored data subsets through the organized story.

R4. Supporting customization of data stories. Users should be allowed to customize their data stories conveniently. The tool should allow users to revise story content and organize stories freely in the tool.

R5. Integrating with existing tools. The storytelling tool should be integrated with common computational notebook environments and presentation tools. Therefore, users do not need to adjust their formed habits. They can explore data and improve the data stories with familiar computational notebooks and presentation tools.

4 NOTABLE

In this section, we first present an overview of *Notable* (Section 4.1). Then we introduce the definition of data facts (Section 4.2) and the modules that support data storytelling during data exploration (Sections 4.3 and 4.4).

4.1 Overview

Notable is a computational notebook extension that provides assistance to data story authoring during data exploration. Following **R5**, it is integrated with one of the most widely used computational notebook environments, JupyterLab. The tool supports exporting data stories as PowerPoint files, which facilitates follow-up modification and presentation.

Figure 1 shows how *Notable* looks in the notebook interface. It composes of multiple *plot widgets* (*e.g.*, Figures 1(b)) and an *organization panel* (Figure 1(c)) for each notebook. Each plot widget presents the user-created chart and illustrated data facts. The organization panel presents the organized data facts and allows the user to adjust the story organization. Plot widgets and the organization panel are the *interactive modules* in *Notable*. These modules, together with the input cells in computational notebooks, enable exploring datasets and creating data stories in a unified tool (**R1**).

To support the functionalities of interactive modules, three *computation modules*, *fact illustration*, *fact organization*, and *slide generation*, are designed. Their relationship is illustrated in Figure 2. The fact illustration module accepts users’ chart specifications in Vega-Lite [41] as the input and illustrates the potential data facts. The advantage of Vega-Lite is that it is a high-level visualization specification grammar with a relatively low learning cost. In the future, it is possible to support other approaches of chart specification, *e.g.*, Matplotlib [19]. The illustrated facts are displayed in a plot widget. Once a data fact is selected by the user, the fact organization module suggests a potential position of the fact in the data story. The organization panel displays the current data story that the user is working on. After the story is compiled, the slide generation module generates a slide deck for further editing. The three modules enable transferring data facts from exploration to storytelling seamlessly (**R1**).

4.2 Data Fact Definition

In *Notable*, facts are considered as the basic units in a data story. *Notable* first identifies the facts in specified charts and illustrates them. Then the selected facts are organized into a story. Following a previous study [58], we characterize data facts using seven

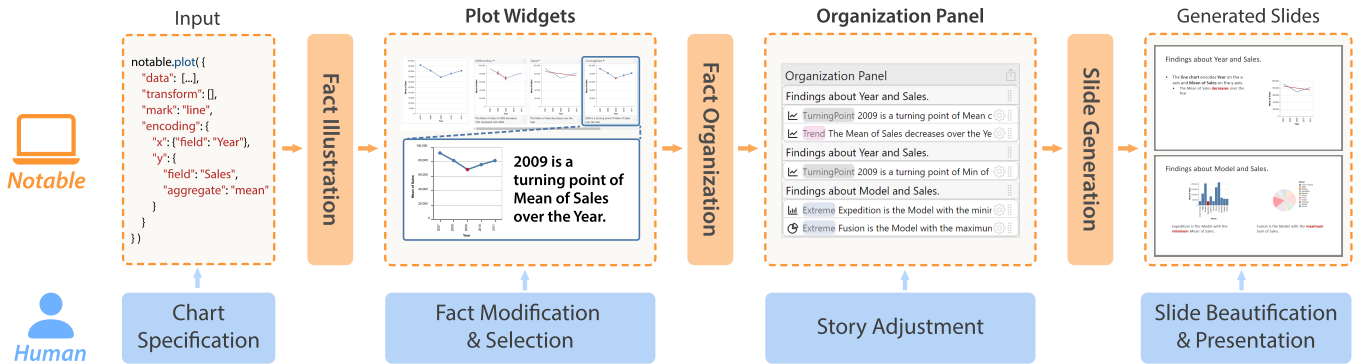


Figure 2: This figure shows the workflow of *Notable*. *Notable* contains five major modules: fact illustration, fact organization, slide generation, plot widget, and organization panel. The last two are displayed in the computational notebook interface.

attributes: *subspace* to record the filters that are applied to gain the visualized data subset; *measures* to indicate the dependent variables in the chart; *dimension* to represent the independent variable in a chart; *type* of data facts (e.g., trend and outlier); *parameters* to describe the details of the fact, such as the direction of trends; *focus* to document the data point that is emphasized in the fact; and *score* to measure both how the fact matched users' intent of exploration and how important the fact is.

4.3 Computation Modules

In this section, we introduce the computation modules that support the interactive modules.

4.3.1 Fact Illustration. According to our interviews and previous research [56], documenting data findings is one of the pain points in storytelling with computational notebooks. To tackle the issue, we propose to illustrate facts automatically to reduce the workload of manual documentation (R2). If users are interested in some illustrated facts, they can directly add them to stories without additional operations. The added facts will be shown in the organization panel for convenient inspection later. To achieve automatic fact illustration, it is essential to infer what fact types the user is interested in and then identify those important facts. We adopt fact mining algorithms [12, 58] to extract facts that will be illustrated.

When a chart is plotted using *Notable*, the subspace, the measure, and the dimension are extracted directly from its specification. With the three attributes, *Notable* first transforms the input dataset by applying filters and aggregations. Then *Notable* attempts to search for potential data facts in the transformed dataset. For example, it identifies whether outliers exist using the widely recognized three-sigma rule⁷ and conducts regression to check the existence of trends. After this step, *Notable* constructs a collection of potential data facts with their fact types, focuses, and parameters. At the end of fact computation, all potential facts are sorted according to their scores, and top- k facts among all are illustrated to users. k is set to 3 by default due to the limited screen space and can be configured by users. To ensure the diversity of recommended data facts, *Notable* first selects the data facts with the highest score of every type and recommends the top- k facts. If the number of

recommended facts is less than k , the other facts will be sorted in descending order according to scores and be recommended until k facts are presented.

In previous research, the score of a fact represents its importance and is composed of two parts, i.e., *impact* and *significance*. *Impact* represents the coverage of the data subspace over the entire dataset, while *significance* measures how obvious the data fact is. For example, DataShot [58] computes the score of a fact as $score = \sum_{i \in [significance, impact_f, impact_c]} w_i * score_i$, where $impact_f$ is the impact of the focus and $impact_c$ is the impact of the context.

However, the previous definition of fact score does not entirely fit our scenario. We would like to infer users' intent of exploration when plotting charts. Therefore, we further consider the *suitability* score between the fact type and the chart type. The selection of chart type can be related to the users' analysis and presentation purposes since different chart types are effective for different purposes [40]. For example, when a line chart is plotted, the creator may care more about the trend of data instead of the fact where a data point occupies the majority of the overall values. The suitability score is computed as the probability of representing a fact using a certain chart type. The probability is derived from the statistics between fact types and chart types by Wang *et al.* [58]. Their statistics summarize the usage of chart types against fact types in data stories. For example, according to the statistics, 42 in 57 trend facts are represented with line charts. Then the suitability score of illustrating a trend fact in a line chart is $42/57 = 0.74$. Since the ultimate goal of our paper is data storytelling, it is more suitable to use the relationship between fact types and chart types in data stories rather than in other scenarios (e.g., [40]).

Furthermore, the impact of context is meaningless in our setting since all facts derived from one chart have the same context. Therefore, we only keep the impact of focus and calculate it as the proportion of data points in the focus over the dataset. For example, if the focus is a turning point in a dataset of five rows, its $impact_f$ is $1/5 = 0.2$. Since significance scores concern data patterns, we mainly follow QuickInsights [12] and DataShot [58] to compute them and the computation methods depend on fact types. For example, the significance score of difference facts is the normalized relative difference between two data points. In a data column, (1, 5, 15, 14), the relative differences between consequent

⁷https://encyclopediaofmath.org/wiki/Three-sigma_rule

Table 2: The table presents templates of text descriptions generated by the fact illustration module.

Fact Type	Template
Majority	The category of {focus} accounts for the significant amount {ratio} of {measure}.
Extreme	{dimension} has the maximum/minimum {measure} at {focus}.
Outlier	{dimension} has an outstanding {measure} at {focus}.
Turning point	{focus} is a turning point of {measure} over the {dimension}.
Difference	The {measure} of {focus ₁ } increases/decreases {ratio} compared with {focus ₂ }.
Trend	The {measure} increases/decreases over the {dimension}.

data points are 4, 2, and 0.07. Therefore, *significance* of the difference fact regarding (5, 15) is $(2 - 0.07)/(4 - 0.07) = 0.49$. To consider three scores jointly, *Notable* computes the fact score as $score = \sum_{i \in [significance, impact_f, suitability]} w_i * score_i$. Based on the previous approach of parameter selection [58], we empirically set the weights to be 0.5, 0.2, and 0.3, respectively. The weights are adjustable to fit personal preferences. For example, if a user prefers not to consider the suitability of charts, $w_{suitability}$ can be set to 0.

Based on extracted facts, we illustrate the original chart with text descriptions and visual embellishments. The usage of descriptions and embellishments is intended to reduce users' effort in documenting data facts and facilitate understanding data facts [46]. Following previous practice in generating explanations for visualization [31], we design templates for different types of data facts. The list of templates are shown in Table 2. These templates contain information about dimension, measure, focus, type, and parameters. The reason why subspace is not included by default is that we would like to keep the text description simple and concise, which aligns with slide design rules [15]. Users can enable subspace in the description as well. Furthermore, we design three types of visual embellishments for different types of data facts. First, *Notable* highlights the focused data point in data facts that only consider a single value, such as extreme and outlier (Figure 3(a)). Second, to handle facts showing the difference between two data points, *Notable* links two data points and highlights both data points with two arrows that indicate the direction of differences, *i.e.*, increasing and decreasing (Figure 3(b)). At last, since the fact regarding data trends considers all data points, *Notable* adds an additional trend line to demonstrate the trend (Figure 3(c)). The visual highlights are added to the charts by modifying the chart specification, which facilitates potential manual chart improvement. Then the illustrated data facts, together with the original chart, are passed to the plot widgets for users' inspection.

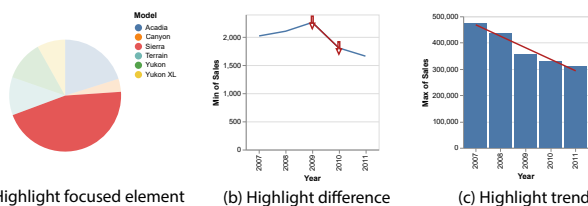


Figure 3: This figure presents how *Notable* illustrates data facts with different visual embellishments.

4.3.2 Fact Organization. Users are allowed to freely organize their identified data facts in *Notable*. However, according to our interviewees, organizing facts into data stories requires considerable effort. To reduce the effort, we provide the fact organization module to suggest a potential arrangement of facts according to the data relationship (**R3**).

After an illustrated data fact is selected through plot widgets, *Notable* organizes all selected data facts at once. *Notable* first goes through the entire slide deck and searches for a suitable slide where the fact can be inserted. In our algorithm, a suitable slide has two criteria: (1) all facts in the slide are identified in the same chart; (2) the slide has less than three facts. The rationale behind the two criteria is to minimize the diversity and quantity of information in one slide [15]. If a suitable slide exists, the new fact will be inserted, and the sequence of facts in the slide will be re-arranged. If there is no suitable slide, a new slide will be created for the fact, and the sequence of slides will update.

To arrange the sequence of facts, we follow Hullman *et al.* [18] and Kim *et al.* [23] to minimize the sum of transition costs between adjacent facts in the fact sequence. Each transition cost is calculated based on several factors mentioned in previous research [18], including the temporal relationship between facts and the consistency between fact focuses. Similarly, *Notable* minimizes the transition costs between slides. When computing the transition costs between slides, one issue is that a slide may contain facts based on different data attributes or subspaces. The issue appears when the user manually places facts based on different charts in one slide. In such a situation, *Notable* cannot directly estimate the transition costs led by changes in data attributes between slides. Inspired by DataShot [58], *Notable* first extracts the shared measure, dimension, subspace, and focus as the *topic* of the slide. The topic of a slide can reveal the core idea of the slide. Therefore, we are able to estimate the transition cost between two slides as the transition costs between topics using the approach in previous research [18]. The slide title is also generated with the shared data attributes in its topic. Another special consideration is the sequence of chart creation in the notebook. The chart sequence may reveal the user's logic flow. To preserve users' logic flow in the organized story, *Notable* considers chart position relationship in computing transition cost as well.

Since the users are allowed to adjust the sequence of facts and slides manually, the fact organization module follows the principle that the users' actions prioritize automatic organization (**R4**). Once the sequence has been updated by users manually, the newly inserted slides or facts will not affect the manually arranged sequence.

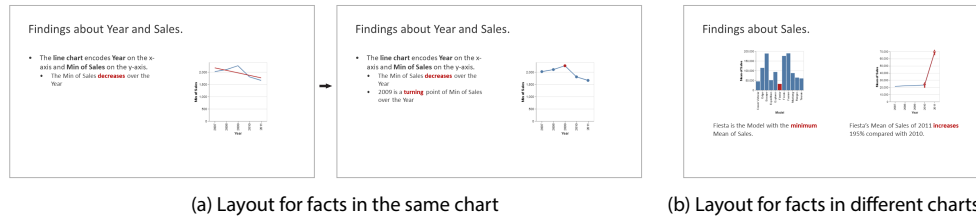


Figure 4: This figure shows two layout designs in the slide generation module.

4.3.3 Slide Generation. The last computation module generates slides when users feel comfortable with the organized story. The slide generation module can export the story as PowerPoint slides. By doing so, we aim to allow users to further edit slides with familiar tools, e.g., changing the slide template and adding animation (R4, R5). Users can also easily re-use the results of data exploration without opening notebooks again.

The generated slides have two layouts to accommodate different arrangements of facts. The first layout is designed to present multiple facts observed in the same chart progressively (Figure 4(a)). On the left, an introduction to the encoding of charts is placed at the top. Then the facts are introduced one by one. The chart on the right changes as a new description of the fact is added. The design aims to let the audience focus on one fact at one time. The second layout presents facts derived from different charts (Figure 4(b)). The side-by-side design can better facilitate the need to present the relationship between two facts. Due to the limited screen space, the second layout does not have an introduction to chart encodings. The slide generation module also highlights key texts in facts such as the fact types and fact parameters. In this way, the data facts can be better conveyed to the audience [15].

4.4 Interactive Modules

This section introduces the interactive modules in the notebook interface. They present the results of the computation modules to users and enable seamless story creation (R1) and customization during data exploration in computational notebooks (R4).

4.4.1 Plot Widget. As shown in Figure 1, one plot widget appears after the input cell. It is designed to support browsing, editing, and selecting facts for storytelling during exploration (R1). A plot widget presents the original chart at the leftmost position (Figure 5(a)) and illustrated data facts at the right (Figure 5(b)). The data facts follow a decreasing order of their scores. The layout is inspired by Lux [27]. Users can scan through all suggested data facts without additional interaction. The card of a data fact contains the fact type at the top (Figure 5(c1)), the illustrated chart in the middle (Figure 5(c2)), and the text description at the bottom (Figure 5(c3)). Users are also allowed to select the fact type with a dropdown list, modify the text description in a text entry box, and click on the data point to highlight it. Such interactions enable users to create customized facts by assigning fact types, documenting related findings, and highlighting key data points. These customized facts can be added to the story. *Notable* currently supports manually highlighting a single data point and can be extended to multi-point selection in

the future. Users can click the cross icon \otimes at the top right corner (Figure 5(d1)) to delete a fact and click on the plus icon \oplus in the middle of an empty card (Figure 5(d2)) to add a fact. When a useful data fact is noticed, clicking the plus icon \oplus at the bottom right corner can add the fact to the story (Figure 5(e1)). Once a fact is in the story, its card will be highlighted with a blue shadow, and the plus icon \oplus turns into a minus icon \ominus (Figure 5(e2)). Clicking the minus icon will remove the fact from the story.

4.4.2 Organization Panel. When data facts are selected, they will be added to the story, and the organized story is shown in the organization panel. The organization panel presents the outline of the story and allows customization (R4), as shown in Figure 6. It also supports the need for convenient fact documentation (R2) by presenting users' selected facts. Each card in the organization panel represents a slide (Figure 6(a)), while each list item in a card encodes a fact (Figure 6(b)). The top part of each card is a modifiable slide title (Figure 6(c)). Due to screen space constraints, the illustrated data fact can hardly be presented in the organization panel. To mitigate the issue, a glyph of chart type (Figure 6(d1)), the fact type (Figure 6(d2)), and the text description (Figure 6(d3)) are provided. Similar to plot widgets, the fact description can also be customized. Furthermore, when clicking the gear icon \textcircled{g} (Figure 6(e1)), more operations are provided, i.e., removing the fact from the story and creating a new slide for the fact (Figure 6(e2)). The organization panel supports the sequence adjustment by dragging facts or slides and dropping them to the desired positions with the grip icons (Figures 6(e3) and (e4)). When users are satisfied with their slides, they can export the slides by simply clicking the icon \textcircled{u} at the top right corner of the panel (Figure 6(f)).

When designing the organization panel, we first thought about showing the story with a tree of facts [33, 34]. However, one concern of tree-based design is the scalability issue [17, 51]. When the story has many facts, it will be challenging to display them. Furthermore, since the slideshow follows a linear sequence, the users need additional mental effort to map a hierarchical tree of facts to a linear presentation. Another possibility is to show all output slides [69] and allows direct slide manipulation. It facilitates an intuitive preview of generated slides. However, given the limited screen space, the users can hardly gain an overview of the story and adjust the sequence. Furthermore, *Notable*, as a lightweight extension, is challenging to provide the complete functionality of slide editing as PowerPoint does. Therefore, we decided to present the story's overview, similar to PowerPoint's outline view. Such a design shows the complete sequence of facts while preserving

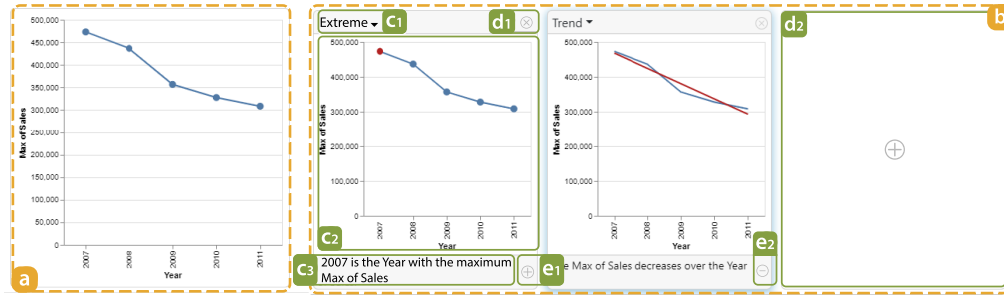


Figure 5: This figure demonstrates a plot widget in *Notable*. It contains the original plot in (a) and suggested data facts in (b). The plot widget supports fact customization with (c1)-(c3). (d1) and (d2) allow users to delete existing facts and create new facts. (e1) and (e2) enable adding and removing facts from the story.

the hierarchical relationship between facts and slides. At the same time, some basic functionalities, such as revising the description, are provided. Users can improve the slides further, such as adding animations, with their familiar presentation tools (R5). In the future, it might be more helpful to provide both the outline view and the slide view in *Notable* and support more functionalities.

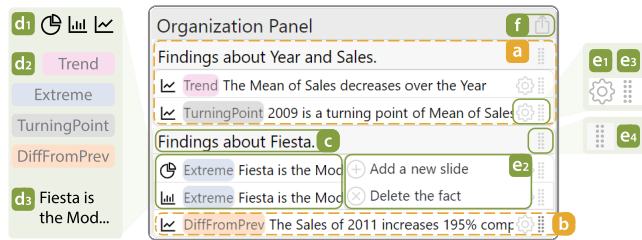


Figure 6: This figure demonstrates the organization panel in *Notable*. In this figure, the center part is the organization panel in *Notable*. The left and right parts with a green background are for illustration. The organization panel provides an outline view of the organized story and supports story adjustment. (a) and (b) represent a slide and a fact, respectively. (c) is the slide title. (d1)-(d3) show the chart type, fact type, and description of a fact. (e1)-(e4) enable fact sequence arrangement. (f) is for exporting the slides.

5 USAGE SCENARIO

In detail, we describe a usage scenario to illustrate how *Notable* assists data storytelling in computational notebooks. Imagine Dora, a business analyst from BMW who plans to explore the car sale dataset and report findings in an upcoming meeting. The dataset records the sales of cars from different brands and categories within five years. There are five columns, including one quantitative attribute (*Sales*), three categorical attributes (*Brand*, *Model*, *Category*), and a temporal attribute (*Year*).

Exploring data by creating charts. To gain an overview of BMW sales, Dora first probes into the relationship between time and sales with a line chart (Figure 7(a)). Based on the chart, *Notable* illustrates three data facts regarding the difference between two consecutive years, the overall trend, and the turning point. Dora

quickly scans through the illustrated facts and selects the fact regarding the trend to give a big picture of sales. She also notices the year 2009 as a turning point over the five years, which may be worth reporting. She adds the fact into her story. Dora plots the sales of all brands over the past years to learn whether the year 2009 is only a turning point of BMW sales or all brands' sales. She finds actually all brands' sales experience a similar trend and has the year 2009 as a turning point. To facilitate the comparison, she moves the facts about data trends and turning points and places them with corresponding BMW's facts in the same slide (Figure 7(a1)). Next, Dora investigates the sales in 2009 as it is a turning point. She checks the relationship between categories and sales (Figure 7(b)). *Notable* highlights two facts about extreme values, the maximum at compact models and minimal at sporty models. By comparing the two facts, she realizes the unbalanced sales of different categories. Both facts are included in her story. She also wonders about the sales of car models in 2009. She plots a bar chart and notices BMW Z4 is the model with the worst sales (Figure 7(c1)). The fact about Z4 is added to her story as well. Then Dora reads her story in the organization panel. The story starts with an overview and then drills down to the sales by categories and models in 2009, a turning point. A question comes to her: "do BMW Z4's low sales lead to the overall unsatisfactory sales in 2009?" She plots the relationship between models and average sales over the past years. However, there is no obvious difference between BMW Z4's sales in 2009 and its average sales (Figure 7(c2)). Therefore, her hypothesis is rejected. Furthermore, she notices BMW X3 has unusually low performance in 2009 compared to its average sales. The finding may help explain the overall low sales in 2009. To report the finding, she creates two new facts about BMW X3 with *Notable* and writes down her observation. The bars of BMW X3 are highlighted by clicking on them (Figure 7(c1) and (c2)). Finally, she adds both facts to her story and ends her exploration.

Organizing data facts and exporting. While adding data facts into the organization panel, Dora observes that *Notable* organizes facts automatically. She notices that the sequence of slides follows the drill-down pattern of data stories (Figure 7(b2)). The slide about sales and categories in 2009 is after the slides that introduce the trend over five years. Moreover, Dora notices that *Notable* generates slide titles (e.g., "Findings about Sales and Year") simultaneously (Figure 7(b1)). After exploration, satisfied with most slide sequences,



Figure 7: This figure illustrates the usage scenario. (a)-(c) demonstrate the procedure of creating a data story with *Notable*. (d) shows the exported slides. (d1) and (d2) correspond to the facts identified in (a); (d3) and (d4) are generated based on (b); and (d5) shows the facts in (c).

Dora merges two facts about BMW X3 in one slide (Figure 7(c3)) and removes the fact about BMW Z4. Ultimately, Dora clicks the export button and downloads the slides (Figure 7(d)). She adjusts the style of slides with PowerPoint and shares them with her team.

6 USER STUDY

We conducted a user study to verify the effectiveness and usability of *Notable*. The setup of our user study is introduced in Sections 6.1-6.3 and the results are reported in Section 6.4.

6.1 Participants

In our user study, we recruited 12 data workers (11 male and 1 female, $Age_{mean} = 28.75$, $Age_{std} = 5.90$, $Experience_{mean} = 5.67$ years, $Experience_{std} = 2.72$ years) from our institution through

social media and word-of-mouth (denoted as U1-U12). They were one software engineer (U1), five postgraduate researchers (U2, U3, U6, U7, U9), three applied data scientists (U4, U10, U12), two research scientists (U5, U11), and one product manager (U8). Their experiences in data analysis ranged from two years to ten years. Since *Notable* is developed based on Python and JupyterLab, we required the participants to be familiar with them as well.

6.2 Task and Dataset

In our user study, we designed an open-ended data analysis task that involves data exploration and storytelling. Participants were asked to explore a given dataset with charts and prepare a slide deck to tell a data story about the findings. We required that the participants explored the dataset with at least six charts and made

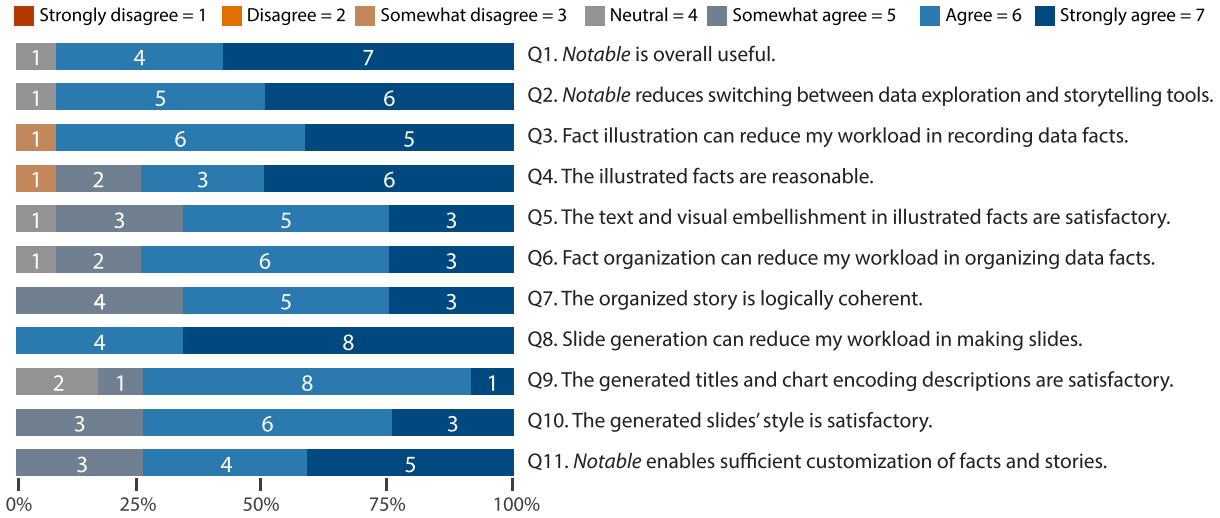


Figure 8: This figure shows the results of our quantitative evaluation of the effectiveness.

a story with at least five charts. By doing so, we would like to ensure that participants explored the dataset with *Notable* sufficiently and were able to make a story. We selected *movies* dataset from Vega Datasets⁸, which have been widely used in evaluating the effectiveness and usability of intelligent visualization tools, including Voyager2 [63] and DashBot [11]. Another advantage of using *movies* dataset is that its data attributes are familiar to the general public [40]. Therefore, the performance of users will not be affected by their limited knowledge of datasets. To control the time of our user study, we reduced the dataset size to 9 attributes and 392 rows. The remaining attributes still covered three major types of tabular data (*i.e.*, nominal, quantitative, and temporal). We also removed null values to eliminate the workload of data cleaning and let the participants focus on data exploration and storytelling.

6.3 Procedure

All studies were conducted through one-on-one offline meetings. Before the user study, we briefed the procedure and asked for the participants' consent to record the study. In the user study, we first introduced different components of *Notable* and related interactions. Then the participants were asked to try *Notable* for around 10 minutes until they felt familiar with it. Next, we asked the participants to finish the task using *Notable*. The participants ended the task until they were satisfied with their slides. The task took around 30 to 40 minutes. In the end, we interviewed the participants regarding their experiences. They were also asked to fill a 7-point Likert questionnaire to rate their effectiveness and usability. In the questionnaire, 1 point means "strongly disagree" while 7 points mean "strongly agree". The eleven questions about the effectiveness are shown with results in Figure 8. Q1 and Q2 evaluate the overall performance of *Notable*. Q3-Q5, Q6-Q7, and Q8-Q10 evaluate the fact illustration, fact organization, and slide generation modules, respectively. Q11 asks whether *Notable* enables users' sufficient

customization. The questions regarding usability were from System Usability Scale (SUS) [4], a widely adopted approach to measure the usability of an application. It took around 20 to 30 minutes to finish the interview and the questionnaire. The whole user study lasted about 1-1.5 hours. Each participant received \$7.5 as compensation. The authors took notes to record feedback during the study.

6.4 Results

In this section, we report the quantitative and qualitative results (Sections 6.4.1 and 6.4.2) of the user study.

6.4.1 Quantitative results. The quantitative results of our user study reflect participants' ratings on both effectiveness and usability. The effectiveness ratings are shown in Figure 8. As the results indicate, most of the participants felt satisfied with *Notable*. They agreed that the three key computation modules, fact illustration, fact organization, and slide generation, were useful and able to achieve their expectation. Furthermore, we noticed that all participants agreed that *Notable* had sufficient support for story customization (R4). The usability score of *Notable* reaches 86.1, which is higher than 95% of applications, according to Sauro and Lewis [42]. The results show that the participants thought highly of the usability and, the interactive modules were intuitive to them.

6.4.2 Qualitative results. This section reports participants' qualitative feedback about the overall experience, illustrated facts, organized stories, and generated slides. To derive the qualitative results, two co-authors summarized the participants' feedback after reading the recording transcripts and the notes taken during the study individually. Then co-authors discussed together and reached a consensus on the common findings revealed in the user study.

Notable can help users create data stories during data exploration. All participants appreciated the overall experience of using *Notable*. For example, U1 expressed his feeling by saying that "*the tool is amazing*". U2 could not wait for the release of *Notable* and said that he would like to install it soon. They believed that

⁸<https://github.com/vega/vega-datasets/>

the workflow of *Notable* is reasonable and has the potential to reduce their workload in creating data stories with findings from data exploration. For example, U12 said “*the tool can integrate data processing, data exploration, chart plotting, and slide generation and thus reduces switching between multiple tools*”. U10 expressed a similar opinion by saying “*the overall workflow is promising*”. Their opinion verified that the design requirement **R1** had been fulfilled. Furthermore, they felt that *Notable* is integrated into computational notebooks seamlessly. U12, as an experienced commercial software developer, believed its good integration into a commonly used notebook environment could attract a broad range of users.

Illustrated facts have the potential to benefit users from multiple perspectives. Among all functionalities in *Notable*, fact illustration has been mentioned frequently as one of *Notable*'s most helpful functions. Most of the participants agreed that automatically illustrated facts are satisfactory and are likely to reduce their efforts in recording and highlighting data findings (**R2**).

Besides the reduction of fact recording efforts, some other benefits brought by fact illustration were also noticed. First, U1, U2, and U8 commented that the illustrated facts revealed widely examined data patterns such that they could save time in writing and running the analysis code. For example, when data trend is illustrated in the charts, the regression analysis may be eliminated. U8 said “*the automatic extreme fact generated by the system is a great help, especially for huge datasets*”.

Second, U6 and U11 also mentioned that the illustrated data facts could guide his data exploration. When U6 started exploring the movie dataset in our user study, he did not have a clear idea about what interesting story might be distilled from the data. The illustrated data facts (Figure 9(b)) served as “*hints*” to him and guided him to continue the exploration of movies that belongs to the genre of drama. Finally, he was able to create a complete data story as Figures 9(c1)-(c5) show. In the story, he first identified drama as the most frequent genre (Figure 9(c1)). Then he reported that the average rating of drama movies is the third highest among all genres (Figure 9(c2)). The trend of ratings of drama movies was further shown in the next two slides (Figures 9(c3) and (c4)). At the end of the slide deck, he highlighted that the production cost of drama films was low though their ratings were great, to attract the audience's interest (Figure 9(c5)).

Though the fact illustration function was thought highly, it has room for future improvements. First, the diversity and complexity of illustrated facts should be enhanced in the future. U10 was the only user who expressed unsatisfactory with the performance of fact illustration in Q3 and Q4 of the questionnaire (Figure 8). The reason was that he felt the facts were not “*in-depth*”. In the user study, he attempted to identify whether the gross of a film was predictable and then presented the results to his team. First, he identified that the relationship between movie genres and average gross might be used for prediction. Then he further confirmed whether the distribution of movie gross was concentrated in each genre and whether the sample size of each genre was large enough. Such results could help him to determine whether the relationship was robust. However, *Notable* failed to consider these analyses, and thus he had to record data findings manually. He suggested that *Notable* can be further improved to consider the complex and

diverse insights according to the users' intention, such as identifying relationships for prediction. U3 and U5 also expressed their expectation for more diverse facts such as data clusters. Second, the description can be improved. Though most of the users were satisfied with the results (Figure 8), U2, U6, and U7, as experts in natural language processing, pointed out that our template-based descriptions were less flexible and thus resulted in some unnatural expressions, e.g., “*Director has the maximum Mean of US Gross at James Cameron*”. To address the issue, they suggested the usage of advanced language generation models, such as T5 [37], but also warned that such models were less controllable and might increase the latency of description generation.

Fact organization facilitates more than storytelling. According to our observation, the participants followed the suggested data fact organization most of the time and did not often arrange facts in stories manually. The observation is also supported by the quantitative results in Figure 8. The results show that our organized stories were generally considered logically coherent (**R3**) and users' effort in creating stories is possible to be eliminated.

On top of its advantages to storytelling, we received feedback in which the fact organization function and the organization panel were reported to benefit their data exploration. For example, U7 felt that the panel could give him an overview. U4 commented that “*the organization panel helps taking notes and collecting interesting insights*” and “*(the organized story) facilitates and guides the data exploration in the next step*”. U6's comments supplemented U4's comment by mentioning that the organized story helped him identify logical flaws in the exploration. U1 and U10 further emphasized the importance of presenting organized facts in the organization panel when they conducted complex data analysis. U10 mentioned that “*the panel will be really helpful when conducting complex data analysis, especially those analyses where I need to explore data back and forth. For example, it helps track what directions I have explored*”. Such comments verify that organizing data facts into stories can have a positive effect on data exploration as well, which demonstrates the value of providing on-the-fly assistance to storytelling during data exploration (**R1**).

We also notice one potential improvement on *Notable*'s fact organization module. As introduced in Section 4.3.2, our algorithm of fact organization minimizes the transition cost of data stories. However, the semantic relationship between data facts is not thoroughly considered and thus leads to a suboptimal case. In U6's data story (Figure 9), he would like to express that the budgets of drama movies were low though the ratings were high using slides in Figures 9(c2)-(c5). However, the fact organization module did not consider the text description and placed the slide in Figure 9(c5) between slides in Figures 9(c1) and (c2). The reason was that slides in Figures 9(c1), (c2), and (c5) were about movies of all genres, while slides in Figures 9(c3) and (c4) only concerned drama movies. The fact organization module considered the analysis from slides in Figures 9(c1), (c2), and (c5) to slides in Figures 9(c3) and (c4) as a drill-down analysis and therefore arranged the sequence in a sub-optimal approach. U6 had to arrange the facts according to the semantic meanings manually. Such a mistake reveals the necessity of considering user-specified semantic information of data facts in future improvements.

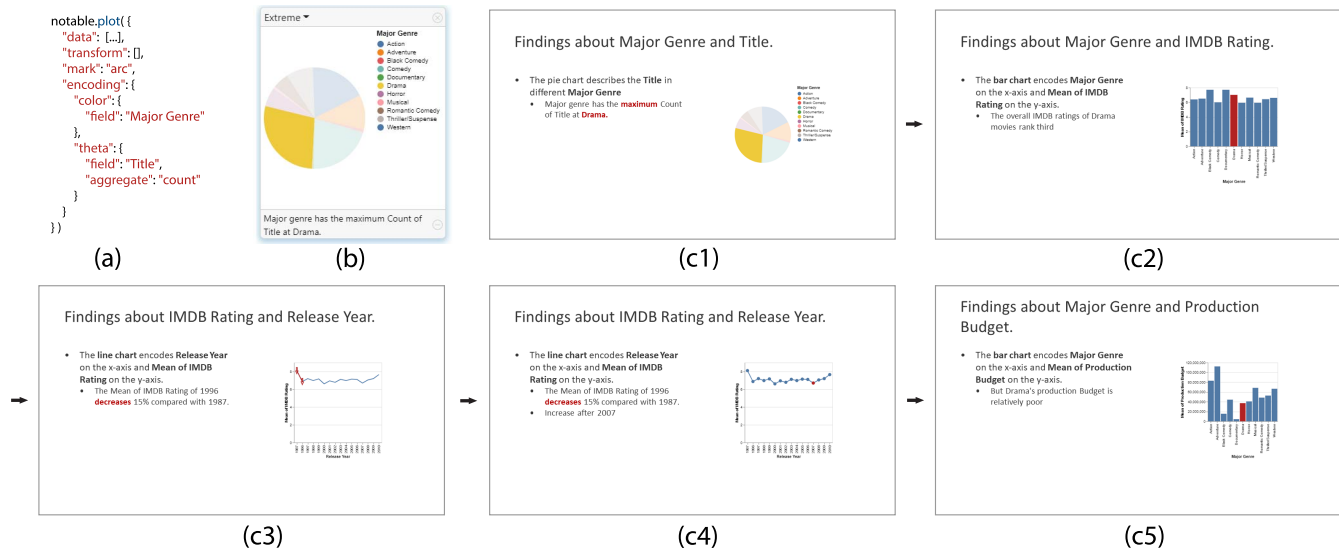


Figure 9: This figure presents a case in our user study. When the participant used (a) as inputs, *Notable* illustrated a data fact (b). In the fact, drama was highlighted as the genre which has the most movies. The data fact guided the participant to conduct a series of explorations on drama movies. The participant finally created a data story as (c1)-(c5) show.

Generated slides will be better with a more personalized design. The slide generation function was appreciated by the participants since it is likely to reduce their effort in making neat slides with data findings. U11 described the functionalities as “one-click generation”, which demonstrates that he considered our slide generation as a convenient function.

However, regarding the style of slides, different participants hold diverse opinions. For example, U2 appreciated the generated slides since they were “organized and neat”. The highlighted texts in the slides were well received by U9. However, U1 expressed a different opinion. Though he agreed that the generated slides can be presented in informal meetings, he would spend great effort to beautify them before presenting them to his supervisor. Furthermore, different users had different opinions about the quantity of information in generated slides. For example, U10 considered the chart description is too long. In practice, he only wrote bullet points such as “column A vs. column B”, e.g., “Major Genre vs. US Gross”. On the contrary, U5 commented that the current design is less informative than his slides. He preferred to introduce more about data facts with images and texts. According to the feedback, we summarize two implications for future tools. First, to fulfill various requirements, the tool should provide diverse slide templates including the content and style. For example, chart descriptions can be bullet point-style or sentence-style. Second, it is important to allow future modification of exported slides with presentation tools. Even though diverse templates are provided, it may not be possible to fulfill users’ personal needs. For example, U9 mentioned that he would like to add animation to slides. Therefore, facilitating further improvements is necessary.

7 DISCUSSION

In this section, we first highlight two lessons learned from the research and future directions about *human-machine collaboration in Notable* (Section 7.1) and *connection between data exploration and storytelling* (Section 7.2). Then the limitation of our research is discussed (Section 7.3).

7.1 Human-machine collaboration in Notable

In our paper, we propose *Notable*, a computational notebook extension, to introduce machines to the loop of data exploration and storytelling. According to our evaluation results, the workflow involving *Notable* is highly appreciated by participants since it both provides necessary assistance, such as highlighting data points and transferring data, and allows humans to control the entire process. With *Notable*, the user is responsible for deciding what data to be explored while *Notable* only illustrates the potentially interesting data facts according to the user-created charts (Section 4.3.1) and organizes user-selected data facts. After that, the user takes charge of reviewing the logical flow of the story when creating stories. On the other side, machines take responsibility for several repetitive tasks, such as transferring data from the exploration stage to the storytelling stage and highlighting key data points. Though these tasks seem not to be challenging, they actually take great effort from users, according to our formative interviews (Section 3). Furthermore, potential errors that may be introduced by humans in data transferring could be avoided [3].

We also receive some comments on refining the workflow. U4 proposed that the machine could take some more steps to facilitate users further. He thought that *Notable* should suggest more related data facts when a data fact is selected, similar to the recommendation in Lux [27] and Erato [48]. He also mentioned that *Notable*

might recommend and change the chart types of the original chart to present data facts more effectively. If these changes are applied, we consider that machines are not only responsible for repetitive tasks but also attempt to guide or correct humans, which can be “double-edged swords” [28]. When machines take further steps, humans need to spend extra mental load to understand why these steps were taken. When the results are worse than expected, humans may lose confidence in machines. The bias led by machines should also be aware. In the future, we will continue the research on human-machine collaboration to study how to maximize the values of machines and humans in data storytelling.

7.2 Connection between data exploration and storytelling

Data exploration and storytelling are two necessary stages that are closely connected theoretically. However, previous studies have mentioned that they are loosely connected practically, considering the gap in converting analysis results to data stories [3, 9, 14].

From the feedback in the evaluation, we find *Notable* has the potential to help users quickly create a data story in the form of presentation slides simply with several clicks (Section 6.4), which matches the design requirements of *Notable*. On top of that, it is interesting to verify that presenting the organized story in the organization panel has a positive influence on data exploration. Several user study participants indicated that the organized story could help them identify the logical flaw and remind them of explored facts (Section 6.4.2). Feedback from our user study indicates that the design of *Notable* not only facilitates the connection between the two stages but further enhances the bi-directional connection of both data exploration and storytelling. It also informs us of the necessity of enhancing the bond between them in future tools. In the future, some steps can be taken toward bridging the gap between exploration and storytelling. In the formative study, P4 pointed out that he would like to transfer data findings among various data exploration tools (e.g., Stata⁹). Such requirements encourage us to develop multiple versions of *Notable* for different data exploration tools. Our computation modules can be re-used and the interactive modules require refinement to fit different interfaces. Furthermore, the support of other storytelling formats, such as reports, is possible to be integrated into *Notable* by augmenting the slide generation module. It will also be interesting to investigate the factors that affect the bi-directional connection between exploration and storytelling such as the approach of presenting data stories in storytelling tools. The results can be applied to optimize the design of *Notable*.

7.3 Limitations

Our research is not without limitations. In this section, we discuss the limitations of our research from the functionalities and the evaluations of *Notable*.

7.3.1 Functionalities. The functionalities of *Notable* can be further extended. First, *Notable* is limited by supporting data facts in basic charts. Currently, *Notable* only supports five basic chart types: bar chart, pie chart, line chart, area chart, and scatter plot. As mentioned by user study participants, U3 and U10, the support

to other chart types (e.g., heatmaps and box plots) and multi-view visualizations could be introduced. Second, the consideration of user-customized facts is limited. As mentioned in Section 4.4, users are allowed to modify the illustrated facts and create new facts. However, *Notable* is not able to understand users’ input completely. It cannot infer three attributes in a data fact, parameters, the focus, and the score. Therefore, the organization of these customized facts will only consider the other four attributes. Furthermore, the semantic information of created descriptions is not considered in fact organization. As described in Section 6.4.2, failing to consider the semantic information might lead to some suboptimal fact sequences. In the future, we plan to extend *Notable* further to handle users’ input more comprehensively and improve its functionalities.

7.3.2 Evaluations. We conducted an in-lab user study where the participants explored the *movies* dataset and created a data story with *Notable*. There are three perspectives to improve the evaluation. First, comparing *Notable* with users’ familiar real-world workflow may reveal more insights, such as differences in slide quality, preparation time, and the workload of creating slides. Second, more long-term evaluation is desired. The *movies* dataset may not be as complex as real-world datasets. The participants commonly finished the task in 30 to 40 minutes. It will be interesting to learn users’ feedback when *Notable* is applied in their daily work for a longer period. Finally, the participants in both the formative study and the user study have limited coverage. Though we have attempted to improve the diversity of participants, e.g., by recruiting participants with diverse backgrounds and various experiences, we acknowledge some limitations, including the imbalance of gender distribution in the user study and the missing of some types of data workers in the formative study (e.g., data journalists). In the future, we hope to deepen our understanding of data exploration and storytelling in the long-term real-world usage of *Notable* by diverse users.

8 CONCLUSION

To communicate data findings in computational notebooks, users have to spend considerable effort in turning them into data stories. In our research, we explore offering on-the-fly assistance to users to facilitate effective data storytelling during data exploration. We first conducted formative interviews with data analysts with diverse backgrounds to derive the design requirements. Then based on the requirements, *Notable*, a computational notebook extension, is proposed to facilitate fact documentation and organization with intelligent support. *Notable* was generally appreciated by the users in a user study with 12 data workers. In the future, we hope to further improve *Notable* by considering more data fact types (e.g., data clusters) and enabling personalized slide generation. It will also be interesting to investigate other approaches to reduce users’ burden, such as simplifying the input format and recommending facts based on users’ preferences. Furthermore, a long-term evaluation has the potential to reveal its pros and cons in a real-world setting.

ACKNOWLEDGMENTS

The authors would like to thank the reviewers, Aoyu Wu, and Liwenhan Xie for their constructive suggestions and all participants in our studies. The research was partially supported by the Hong

⁹<https://www.stata.com/>

Kong Research Grants Council (GRF16210722) and the National Natural Science Foundation of China (U22A2032).

REFERENCES

- [1] Sara Alspaugh, Nava Zokaei, Andrea Liu, Cindy Jin, and Marti A Hearst. 2018. Futzing and Moseying: Interviews with Professional Data Analysts on Exploration Practices. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2018), 22–31.
- [2] bqplot. 2022. Bqplot: Plotting Library for IPython/Jupyter Notebooks. <https://github.com/bqplot/bqplot>. Accessed on July 18, 2022.
- [3] Matthew Brehmer and Robert Kosara. 2021. From Jam Session to Recital: Synchronous Communication and Collaboration Around Data in Organizations. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2021), 1139–1149.
- [4] John Brooke. 1996. SUS-A Quick and Dirty Usability Scale. *Usability Evaluation in Industry* 189, 194 (1996), 4–7.
- [5] Souti Chattopadhyay, Ishita Prasad, Austin Z Henley, Anita Sarma, and Titus Barik. 2020. What’s Wrong with Computational Notebooks? Pain Points, Needs, and Design Opportunities. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 600, 12 pages.
- [6] Qing Chen, Shixiong Cao, Jiazhe Wang, and Nan Cao. 2022. How Does Automation Shape the Process of Narrative Visualization: A Survey on Tools. *arXiv preprint arXiv:2206.12118* (2022), 20 pages.
- [7] Siming Chen, Jie Li, Gennady Andrienko, Natalia Andrienko, Yun Wang, Phong H Nguyen, and Çağatay Turkey. 2020. Supporting Story Synthesis: Bridging the Gap between Visual Analytics and Storytelling. *IEEE Transactions on Visualization and Computer Graphics* 26, 07 (2020), 2499–2516.
- [8] Yiru Chen and Eugene Wu. 2022. PI2: Slide-to-end Interactive Visualization Interface Generation from Queries. In *Proceedings of the 2022 International Conference on Management of Data*. ACM, New York, NY, USA, 1711–1725.
- [9] Fanny Chevalier, Melanie Tory, Bongshin Lee, Jarke van Wijk, Giuseppe Santucci, Marian Dörk, and Jessica Hullman. 2018. From Analysis to Communication: Supporting the Lifecycle of a Story. In *Data-Driven Storytelling*. AK Peters/CRC Press, Boca Raton, FL, USA, 151–183.
- [10] Matthew Conlen, Megan Vo, Alan Tan, and Jeffrey Heer. 2021. Idyll Studio: A Structured Editor for Authoring Interactive & Data-driven Articles. In *Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 1–12.
- [11] Dazhen Deng, Aoyu Wu, Huamin Qu, and Yingcai Wu. 2022. Dashbot: Insight-driven dashboard generation based on deep reinforcement learning. *IEEE Transactions on Visualization and Computer Graphics* (2022). Early Access.
- [12] Rui Ding, Shi Han, Yong Xu, Haidong Zhang, and Dongmei Zhang. 2019. Quick-Insights: Quick and Automatic Discovery of Insights from Multi-dimensional Data. In *Proceedings of the 2019 International Conference on Management of Data*. ACM, New York, NY, USA, 317–332.
- [13] Tong Ge, Bongshin Lee, and Yunhai Wang. 2021. CAST: Authoring Data-Driven Chart Animations. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 24, 15 pages.
- [14] Samuel Gratzl, Alexander Lex, Nils Gehlenborg, Nicola Cosgrove, and Marc Streit. 2016. From Visual Exploration to Storytelling and Back Again. *Computer Graphics Forum* 35, 3 (2016), 491–500.
- [15] Emily P Green. 2021. The Basics of Slide Design. In *Healthy Presentations*. Springer, Berlin, Germany, 37–62.
- [16] Andrew Head, Fred Hohman, Titus Barik, Steven M Drucker, and Robert DeLine. 2019. Managing Messes in Computational Notebooks. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 270, 12 pages.
- [17] Jeffrey Heer and Stuart K Card. 2004. DOITrees Revisited: Scalable, Space-constrained Visualization of Hierarchical Data. In *Proceedings of the 2004 Working Conference on Advanced Visual Interfaces*. ACM, New York, NY, USA, 421–424.
- [18] Jessica Hullman, Steven Drucker, Nathalie Henry Riche, Bongshin Lee, Danyel Fisher, and Eytan Adar. 2013. A Deeper Understanding of Sequence in Narrative Visualization. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2406–2415.
- [19] John D Hunter. 2007. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* 9, 03 (2007), 90–95.
- [20] Project Jupyter. 2022. Jupyter Notebook Conversion. <https://github.com/jupyter/nbconvert>. Accessed on July 18, 2022.
- [21] DaYe Kang, Tony Ho, Nicolai Marquardt, Bilge Mutlu, and Andrea Bianchi. 2021. ToonNote: Improving Communication in Computational Notebooks using Interactive Data Comics. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 727, 14 pages.
- [22] Mary Beth Kery, Marissa Radensky, Mahima Arya, Bonnie E John, and Brad A Myers. 2018. The Story in the Notebook: Exploratory Data Science using a Literate Programming Tool. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 174, 11 pages.
- [23] Younghoon Kim, Kani Wongsuphasawat, Jessica Hullman, and Jeffrey Heer. 2017. GraphScape: A Model for Automated Reasoning about Visualization Similarity and Sequencing. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 2628–2638.
- [24] Po-Ming Law, Rahul C Basole, and Yanhong Wu. 2018. Duet: Helping Data Analysis Novices Conduct Pairwise Comparisons by Minimal Specification. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2018), 427–437.
- [25] Po-Ming Law, Alex Endert, and John Stasko. 2020. Characterizing Automated Data Insights. In *Proceedings of the 2020 IEEE Visualization Conference*. IEEE, Piscataway, NJ, USA, 171–175.
- [26] Bongshin Lee, Nathalie Henry Riche, Petra Isenberg, and Sheelagh Carpendale. 2015. More than Telling a Story: Transforming Data into Visually Shared Stories. *IEEE Computer Graphics and Applications* 35, 5 (2015), 84–90.
- [27] Doris Jung-Lin Lee, Dixin Tang, Kunal Agarwal, Thyne Boonmark, Caitlyn Chen, Jake Kang, Ujjaini Mukhopadhyay, Jerry Song, Micah Yong, Marti A. Hearst, and Aditya G. Parameswaran. 2021. Lux: Always-on Visualization Recommendations for Exploratory Data Science. *Proceedings of the VLDB Endowment* 15, 3 (2021), 727–738.
- [28] Quan Li, Huanbin Lin, Chunfeng Tang, Xiguan Wei, Zhenhui Peng, Xiaojuan Ma, and Tianjian Cheng. 2021. Exploring the “Double-Edged Sword” Effect of Auto-Insight Recommendation in Exploratory Data Analysis. In *Joint Proceedings of the ACM IUI 2021 Workshops*. ACM, New York, NY, USA, 1–13.
- [29] Xingjun Li, Yizhi Zhang, Justin Leung, Chengnian Sun, and Jian Zhao. 2022. EDAssistant: Supporting Exploratory Data Analysis in Computational Notebooks with In-Situ Code Search and Recommendation. *ACM Transactions on Interactive Intelligent Systems* (2022). Just Accepted.
- [30] Yanna Lin, Haotian Li, Aoyu Wu, Yong Wang, and Huamin Qu. 2022. DMiner: Dashboard Design Mining and Recommendation. *arXiv preprint arXiv:2209.01599* (2022), 14 pages.
- [31] Can Liu, Liwenhan Xie, Yun Han, Datong Wei, and Xiaoru Yuan. 2020. AutoCaption: An Approach to Generate Natural Language Description from Visualization Automatically. In *Proceedings of the 2020 IEEE Pacific Visualization Symposium*. IEEE, Piscataway, NJ, USA, 191–195.
- [32] Junhua Lu, Wei Chen, Hui Ye, Jie Wang, Honghui Mei, Yuhui Gu, Yingcai Wu, Xiaolong Luke Zhang, and Kwan-Liu Ma. 2021. Automatic Generation of Unit Visualization-based Scrollytelling for Impromptu Data Facts Delivery. In *Proceedings of the 2021 IEEE 14th Pacific Visualization Symposium*. IEEE, Piscataway, NJ, USA, 21–30.
- [33] Andreas Mathisen, Tom Horak, Clemens Nylandstedt Klokmoose, Kaj Grønbeak, and Niklas Elmquist. 2019. InsideInsights: Integrating Data-Driven Reporting in Collaborative Visual Analytics. *Computer Graphics Forum* 38, 3 (2019), 649–661.
- [34] Humphrey O Obie, Dac Thanh Chuong Ho, Iman Avazpour, John Grundy, Mohamed Abdelrazek, Tomasz Bednarz, and Caslon Chua. 2022. Gravity++: A Graph-based Framework for Constructing Interactive Visualization Narratives. *Journal of Computer Languages* 71 (2022), 101125.
- [35] David Piorkowski, Soya Park, April Yi Wang, Dakuo Wang, Michael Muller, and Felix Portnoy. 2021. How AI Developers Overcome Communication Challenges in a Multidisciplinary Team: A Case Study. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW1 (2021), 1–25.
- [36] Plotly. 2022. Plotly: The Interactive Graphing Library for Python. <https://github.com/plotly/plotly.py>. Accessed on July 18, 2022.
- [37] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* 21 (2020), 140:1–140:67.
- [38] Deepthi Raghunandan, Zhe Cui, Kartik Krishnan, Segen Tirfe, Shenzhi Shi, Tejaswi Darshan Shrestha, Leilani Battle, and Niklas Elmquist. 2022. Lodestar: Supporting Independent Learning and Rapid Experimentation Through Data-Driven Analysis Recommendations. *arXiv preprint arXiv:2204.07876* (2022), 10 pages.
- [39] Adam Rule, Aurélien Tabard, and James D Hollan. 2018. Exploration and Explanation in Computational Notebooks. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 32, 12 pages.
- [40] Bahador Saket, Alex Endert, and Çağatay Demiralp. 2018. Task-based Effectiveness of Basic Visualizations. *IEEE Transactions on Visualization and Computer Graphics* 25, 7 (2018), 2505–2512.
- [41] Arvind Satyanarayan, Dominik Moritz, Kani Wongsuphasawat, and Jeffrey Heer. 2016. Vega-Lite: A Grammar of Interactive Graphics. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2016), 341–350.
- [42] Jeff Sauro and James R Lewis. 2012. *Quantifying the User Experience: Practical Statistics for User Research*. Morgan Kaufmann, San Francisco, CA, USA.
- [43] Danqing Shi, Fuling Sun, Xinyue Xu, Xingyu Lan, David Gotz, and Nan Cao. 2021. AutoClips: An Automatic Approach to Video Generation from Data Facts. *Computer Graphics Forum* 40, 3 (2021), 495–505.
- [44] Danqing Shi, Xinyue Xu, Fuling Sun, Yang Shi, and Nan Cao. 2020. Calliope: Automatic Visual Data Story Generation from a Spreadsheet. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 453–463.
- [45] Minjeong Shin, Joohee Kim, Yunha Han, Lexing Xie, Mitchell Whitelaw, Bum Chul Kwon, Sungahn Ko, and Niklas Elmquist. 2022. Roslingifier: Semi-Automated Storytelling for Animated Scatterplots. *IEEE Transactions on Visualization and*

- Computer Graphics* (2022). Early Access.
- [46] Arjun Srinivasan, Steven M Drucker, Alex Endert, and John Stasko. 2018. Augmenting Visualizations with Interactive Data Facts to Facilitate Interpretation and Communication. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2018), 672–681.
- [47] Nicole Sultanum, Fanny Chevalier, Zoya Bylinskii, and Zhicheng Liu. 2021. 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*. ACM, New York, NY, USA, Article 16, 17 pages.
- [48] Mengdi Sun, Ligan Cai, Weiwei Cui, Yanqiu Wu, Yang Shi, and Nan Cao. 2022. Erato: Cooperative Data Story Editing via Fact Interpolation. *IEEE Transactions on Visualization and Computer Graphics* (2022). Early Access.
- [49] Bo Tang, Shi Han, Man Lung Yiu, Rui Ding, and Dongmei Zhang. 2017. Extracting Top-K Insights from Multi-dimensional Data. In *Proceedings of the 2017 International Conference on Management of Data*. ACM, New York, NY, USA, 1509–1524.
- [50] Chao Tong, Richard Roberts, Rita Borgo, Sean Walton, Robert S Laramée, Kodzo Wegba, Aidong Lu, Yun Wang, Huamin Qu, Qiong Luo, and Xiaojuan Ma. 2018. Storytelling and Visualization: An Extended Survey. *Information* 9, 3 (2018), 65.
- [51] Ka Wing Tsang, Haotian Li, Fuk Ming Lam, Yifan Mu, Yong Wang, and Huamin Qu. 2020. TradAO: A Visual Analytics System for Trading Algorithm Optimization. In *Proceedings of the 2020 IEEE Visualization Conference*. IEEE, Piscataway, NJ, USA, 61–65.
- [52] Jacob VanderPlas, Brian Granger, Jeffrey Heer, Dominik Moritz, Kanit Wongsuphasawat, Arvind Satyanarayan, Eitan Lees, Ilia Timofeev, Ben Welsh, and Scott Sievert. 2018. Altair: Interactive Statistical Visualizations for Python. *Journal of Open Source Software* 3, 32 (2018), 1057.
- [53] Manasi Vartak, Samuel Madden, Aditya Parameswaran, and Neoklis Polyzotis. 2014. SEEDB: Automatically Generating Query Visualizations. *Proceedings of the VLDB Endowment* 7, 13 (2014), 1581–1584.
- [54] Viola. 2022. Voilà Turns Jupyter Notebooks into Standalone Web Applications. <https://github.com/voila-dashboards/voila>. Accessed on July 18, 2022.
- [55] April Yi Wang, Will Epperson, Robert A DeLine, and Steven M Drucker. 2022. Diff in the Loop: Supporting Data Comparison in Exploratory Data Analysis. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 97, 10 pages.
- [56] April Yi Wang, Dakuo Wang, Jaimie Drozdal, Michael Muller, Soya Park, Justin D Weisz, Xuye Liu, Lingfei Wu, and Casey Dugan. 2022. Documentation Matters: Human-Centered AI System to Assist Data Science Code Documentation in Computational Notebooks. *ACM Transactions on Computer-Human Interaction* 29, 2 (2022), 1–33.
- [57] Yun Wang, Yi Gao, Ray Huang, Weiwei Cui, Haidong Zhang, and Dongmei Zhang. 2021. Animated Presentation of Static Infographics with InfoMotion. *Computer Graphics Forum* 40, 3 (2021), 507–518.
- [58] Yun Wang, Zhida Sun, Haidong Zhang, Weiwei Cui, Ke Xu, Xiaojuan Ma, and Dongmei Zhang. 2019. DataShot: Automatic Generation of Fact Sheets from Tabular Data. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2019), 895–905.
- [59] Zijie J Wang, Chudi Zhong, Rui Xin, Takuya Takagi, Zhi Chen, Duen Horng Chau, Cynthia Rudin, and Margo Seltzer. 2022. TimberTrek: Exploring and Curating Sparse Decision Trees with Interactive Visualization. *arXiv preprint arXiv:2209.09227* (2022). Accepted in IEEE VIS 2022.
- [60] Nathaniel Weinman, Steven M Drucker, Titus Barik, and Robert DeLine. 2021. Fork It: Supporting stateful alternatives in computational notebooks. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 307, 12 pages.
- [61] Thomas Winters and Kory W Mathewson. 2019. Automatically Generating Engaging Presentation Slide Decks. In *Proceedings of the 2019 International Conference on Computational Intelligence in Music, Sound, Art and Design*. Springer, Berlin, Germany, 127–141.
- [62] Kanit Wongsuphasawat, Yang Liu, and Jeffrey Heer. 2019. Goals, Process, and Challenges of Exploratory Data Analysis: An Interview Study. *arXiv preprint arXiv:1911.00568* (2019), 10 pages.
- [63] Kanit Wongsuphasawat, Zening Qu, Dominik Moritz, Riley Chang, Felix Ouk, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2017. Voyager 2: Augmenting Visual Analysis with Partial View Specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 2648–2659.
- [64] Yifan Wu, Joseph M Hellerstein, and Arvind Satyanarayan. 2020. B2: Bridging Code and Interactive Visualization in Computational Notebooks. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. ACM, New York, NY, USA, 152–165.
- [65] Peter Xenopoulos, Joao Rulff, Luis Gustavo Nonato, Brian Barr, and Claudio Silva. 2022. Calibrate: Interactive Analysis of Probabilistic Model Output. *IEEE Transactions on Visualization and Computer Graphics* (2022). Early Access.
- [66] Ji Soo Yi, Youn ah Kang, John Stasko, and Julie A Jacko. 2007. Toward a Deeper Understanding of the Role of Interaction in Information Visualization. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1224–1231.
- [67] Lin-Ping Yuan, Ziqi Zhou, Jian Zhao, Yiqiu Guo, Fan Du, and Huamin Qu. 2021. InfoColorizer: Interactive Recommendation of Color Palettes for Infographics. *IEEE Transactions on Visualization and Computer Graphics* 28, 12 (2021), 4252–4266.
- [68] Qianrui Zhang, Haoci Zhang, Thibault Sellam, and Eugene Wu. 2019. Mining Precision Interfaces from Query Logs. In *Proceedings of the 2019 International Conference on Management of Data*. ACM, New York, NY, USA, 988–1005.
- [69] Chengbo Zheng, Dakuo Wang, April Yi Wang, and Xiaojuan Ma. 2022. Telling Stories from Computational Notebooks: AI-Assisted Presentation Slides Creation for Presenting Data Science Work. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, Article 53, 20 pages.