From Dashboard Zoo to Census: A Case Study with Tableau Public

Arjun Srinivasan, Joanna Purich, Michael Correll, Leilani Battle, Vidya Setlur, Anamaria Crisan

***Abstract*—Dashboards remain ubiquitous tools for analyzing data and disseminating the findings. Understanding the range of dashboard designs, from simple to complex, can support the development of authoring tools that enable end-users to meet their analysis and communication goals. Yet, there has been little work that provides a quantifiable, systematic, and descriptive overview of dashboard design patterns. Instead, existing approaches only consider a handful of designs, which limits the breadth of patterns that can be surfaced. More quantifiable approaches, inspired by machine learning (ML), are presently limited to single visualizations or capturing narrow features of dashboard designs. To address this gap, we present an approach for modeling the content and composition of dashboards using a graph representation. The graph decomposes dashboard designs into nodes featuring content "blocks'; and uses edges to model "relationships", such as layout proximity and interaction, between nodes. To demonstrate the utility of this approach, and its extension over prior work, we apply this representation to derive a census of 25,620 dashboards from Tableau Public, providing a descriptive overview of the core building blocks of dashboards in the wild and summarizing prevalent dashboard design patterns. We discuss concrete applications of both a graph representation for dashboard designs and the resulting census to guide the development of dashboard authoring tools, making dashboards accessible, and for leveraging AI/ML techniques. Our findings underscore the importance of meeting users where they are by broadly cataloging dashboard designs, both common and exotic.**

***Index Terms*—visualization, dashboard, interaction, survey**

I. Introduction

Dashboards are an essential tool for supporting data-driven decision-making across a broad spectrum of domains, including medicine, finance, education, and science. Their applications range from initial exploration of data to monitoring changes in real-time, and finally as a communication tool that can support persuasion and learning [1], [2], [3], [4], [5], [6]. A broad and carefully considered examination of a dashboard corpus is an important precursor to many downstream visual analytics research topics. Its results may reveal not only the diversity of designs across application domains but also surface common design patterns and potential pain points that could inform the requirements of authoring tools. For AI/ML-supported tasks such as dashboard recommendation or interactive guidance, such an examination allows researchers to assess the quality and suitability of the corpus to serve as training data, as well as design filters to identify high-quality source data to boost model performance.

Visualization research has predominantly adopted a 'close reading' qualitative approach to investigate dashboard designs [7], [8] – that is, a detailed analysis of a small number of dashboards. While this approach reveals a rich design space, it also imposes some notable constraints. First, a manual inspection of dashboards is subjective, time-consuming, and practical only at the limited scale of dozens or perhaps hundreds of examples – a small fraction of what exists, making it difficult to capture the significance of dashboard design patterns. Second, many of the dashboards from these prior studies hand-pick examples from different sources across the internet, including news websites, dashboard galleries, and social media. It is not clear how representative these examples are of design practices in general. They may instead represent idealized dashboard designs, authored through a variety of tools at different stages of the design process – reflecting the results of highly skilled designers proficient in many tools rather than a more typical creator. The representation of visualizations and dashboards for use in ML/AI applications (e.g., VizML [9], DMiner [10]) offers an alternative approach, which relies predominately on the extraction of features. However, in addition to largely overlooking interactions and coordinated views, this approach has not been explored for *describing* dashboard corpora to identify and summarize design patterns. In short, existing work on dashboard design has predominately focused on a few hand-picked "zoos" of interesting examples (akin to Heer et al.'s [11] "visualization zoo" of "more exotic(but practically useful) forms of visual data representation").

Borrowing a term and methodology from the digital humanities, we propose a complementary "distant reading" [12] approach that allows for a broader overview of dashboards, essentially a 'census.' ***To derive a census of dashboard design patterns, we propose a graph representation that decomposes dashboards into "blocks" and "connections"*** These blocks can describe the content of a dashboard, while connections reveal their relationships, such as spatial proximity and interactivity. Our approach integrates disparate abstractions from prior studies of visualization and dashboard corpora [9], [13], [10], [8], [7], while also incorporating interactions and non-visualization elements that prior work omits. ***We demonstrate the utility of this schema by deriving a census of 25,620 dashboards on Tableau Public, a large and well-established collection of dashboards that captures a myriad of design goals and applications.*** Our analysis showcases the diversity and prevalence of design patterns that are often overlooked or underappreciated in prior studies as dashboards. For instance, we find a tight coupling of story-driven *text elements* with visualization elements with dashboards, the widespread use of *simple and canonical visualization types* across dashboards over bespoke or novel forms, and the use of *interaction* in dashboards. The prevalence of these patterns, which are less prominent in prior studies of hand-picked corpora, suggest *unmet challenges* for dashboard authoring support.

We distill our findings toward applications of our schema and the use of dashboard corpora for varied downstream

Primary Contact: ana.crisan@uwaterloo.ca

tasks, such as dashboard authoring, accessibility, and AI/ML-supported guidance. In summary, our research makes the following contributions:

- A schematic representation of dashboards as node-link graphs representing the core design elements as well as spatial and interactive relationships between them.
- A case study using Tableau Public to create a census of visual and interaction design patterns. We also release the anonymized corpus of 25,620 dashboards[1]
- Applications and future research trajectories for dashboard authoring tools informed by a design 'census.'

Dashboards remain an essential tool for extracting actionable insights from data. We contribute an approach for visualization researchers to appraise a greater diversity of dashboard design patterns that can in turn be leveraged to improve end-user experiences toward authoring dashboards.

II. Related Work

A. Dashboards as Objects of Study

Sarikaya et al. [7] point to a disconnect between the ubiquity of dashboards in visualization *practice* and their lack of consideration in visualization *study*. More recent work has sought to remedy this gap by 1) further clarifying the various forms and goals of dashboard designers and users and 2) codifying or testing design rules or recommendation systems for automating aspects of dashboard design.

Exploring dashboards (and other visualization practices) is often done through an analysis of dashboards in a particular context of use or population of users [14]. For instance, Tory et al. [15] explore dashboard usage among "data workers." While valuable, these analyses require access to both the people and visualizations they work with and rely on qualitative and subjective judgments of intent or goal, limiting both the scale and generalizability of results. For instance, Sarikaya et al. [7], Bach et al. [8], Al-Maneea et al. [16] all explore dashboards and multiple view visualizations with an eye toward their visual structure and topology, but rely on a manual process of coding dashboard features and connections. These manual inspections are valuable and afford inferences about qualitative information that would be difficult to determine automatically.

The analysis and observation of dashboards are often performed in order to compare these dashboards to existing design guidelines or recommendations from both the academic and practitioner communities [7], [17], [18], [19], [20], [16], [8], [21]. For instance, Qu and Hullman [20] examine how users attend to design inconsistencies between visualizations within the same dashboard and translate their observations into explicit design guidelines for keeping coordinated visualizations consistent. Kristiansen et al. [22] extend these consistency constraints by allowing users to specify *relations*. Similarly, Langner et al. [23] perform an observational study of dashboard use and design in large display environments to inform the design of their coordinated view system. Other dashboard authoring or recommendation systems, especially those that use machine learning, attempt to create meaningful layouts and content [24], [25], [26], [27], but rely on a substantial training corpora of well-designed or useful dashboards.

Our research explores how we can examine dashboard designs at scale. We integrate and extend elements of prior research to propose an extensible and machine-readable schematic representation of dashboard designs.

B. Analyses of Visualization Corpora

Analyses of large corpora of visualizations have been performed for a variety of reasons. For instance, to describe the flexibility of a specific tool and the habits of its users [28], [29], [30], to create and evaluate datasets for training machine learning models [9], [31], or to simply enumerate the sheer diversity and structure of a design space [32], [33], [34]. While our motivations span these categories, we note specific structures in how these corpora are collected and analyzed.

Existing corpora can be divided along three dimensions: data collection that manual [7], [16] versus automated [9], [31], [29], [35], [36], [37], [30], [38], annotation that is manual [7] versus automated [9], [31], [38], [39] (or both [29]), and analyzing visualizations as static [29], [9], [38] versus dynamic [7] (i.e., interactive) objects. Each dimension involves trade-offs in the richness, scope, and quality of analyses supported by the annotated data. For example, automated extraction allows for thousands of examples to be collected, but managing the heterogeneity exhibited in massive corpora can lead to a relatively limited set of features available for analysis based on what extraction and annotation programs can reliably detect *en masse* [29], [38]. Moreover, assessing the quality of data in this corpus is also difficult and may require explicit validation steps [31]. In contrast, manual data collection and annotation can lead to richer input data and thus a wider variety of potential analyses [7], but sacrifice scale in return since manual data collection and annotation involve significant expenditures of time and effort.

A notable exception occurs when a large, consistently formatted corpus is available, enabling richer and broader analyses. For instance, the VizML project [9] processed over a million Plotly visualizations, creating a dataset suitable for training deep learning models. DMiner [10] investigates approximately 850 Tableau dashboards collected from GitHub – similar to VizML [9], they take advantage of a common specification, in this case, Tableau XML, to prepare their corpora for a recommendation task. Both VizML and DMiner use different approaches to represent visualizations and dashboards by either extracting features or using a graph-based analysis. However, these features focus on the individual visualizations and do not capture other elements present in the dashboard. Moreover, neither comments on the variety of visualizations or dashboards (such as the prevalence of emergent patterns and designs) nor do they discuss interaction.

We present a unique opportunity to analyze and share thousands of dashboard designs in a consistent format amenable to systematic, quantitative analysis. Our approach also calls attention to an often overlooked consideration of corpora content and its suitability for downstream research tasks.

[1]We currently provide the corpus at: https://osf.io/r5cfk

C. Graph-Based Analysis of Visualization Designs

Ease in authoring and analyzing visualizations is often linked to the way that a visualization is *specified*. When the initial specifications are not readily available, one could use alternative techniques, such as image segmentation [29], [39], [40], [41], [42], [43] to derive approximate representations of visualizations. However, the heterogeneity of visualization images—and thus, their approximate representations—limits our ability to analyze them at scale. It is also hard to precisely extract higher-level semantics such as layout and interaction properties from images without rich metadata. In our work, given that we are interested in the *relationships* that bind discrete elements within a dashboard together, we rely on *graph-based representations* for our analysis.

A number of works explore graph-based representations of visualization and dashboard designs. For example, visualization recommendation algorithms often represent the visualization design space as a graph, where nodes represent specific encoding or data transformation choices and edges reflect relationships between these design decisions [44], [45], [46]. Dashboard designs can also be represented as a graph to capture relationships between different elements, such as directional relationships between interactions in one element that change the encodings or data transformations in another element [47], [48]. Recent research from Kristiansen et al. [22], [49] proposes a technique for content-driven graph layout for creating multi-view visualizations, including dashboards.

VizML and KG4Vis are most similar to our research. VizML [9] uses a feature base approach and while KG4Vis [13] apply a graph structure for mapping dataset properties to low-level design decisions within static visualizations. KG4Vis takes this idea one step further by computing embedding vectors over the knowledge graphs created for individual visualizations, producing a numeric representation that can be compared for generating and ranking visualization recommendations. However, these methods do not capture relationships between multiple visualizations or non-visual elements like interaction widgets, text, and multimedia. This orchestration of visual, interactive, and textual elements is what distinguishes dashboards from other genres of charts. The generalizability of these approaches to the analysis of dashboards at scale is yet to be demonstrated.

We extend the ideas of prior research while also demonstrating their utility beyond single visualizations. We present a blocks and connections representation of dashboard content and composition that can use feature-based and graph-based analyses to represent both the layout and interactions within an individual dashboard and summarize these design patterns at the level of the entire dashboard corpora.

III. Dashboard Graphs: A Schematic Representation of Dashboard Design

Here, we present a graph representation of dashboard designs. We motivate the need for such a representation, define its elements, and describe its applicability and extensibility.

A. The need for consistent dashboard design representations

To illustrate the challenges of analyzing dashboard designs, consider Figure 1—a small slice of the diversity in dashboard designs—which presents three dashboards that are composed of different visual elements including (but not limited to) data visualizations, and with different levels of interactivity. While these examples are all derived from Tableau Public, a dashboard can be specified programmatically using visualization libraries (e.g., D3, Vega-lite, ggplot) or through direct manipulation via authoring systems (e.g., Tableau, PowerBI, Looker). Each of these approaches has its own mechanism for creating individual visualizations, laying them out, and coordinating interactions between them. To gain insights into the design of these visualizations, it would be necessary to examine their specification via the tool they were created with. However, the task is onerous and has questionable value in summarizing the dashboard's design.

An alternative approach would be to consolidate a summary of their designs and formalize them into a framework or design space description—an approach adopted by prior work [7], [8]. However, this requires human labor to construct and organize artifacts. The manual and subjective nature of this process makes it challenging to apply to large dashboard corpora. We argue that there also exists a gap between the findings from these studies and the ability to express these design patterns in a machine-readable way, for example, as is done with VizML [9] or DMiner [10]. Notably, even these prior attempts to represent visualization and dashboard design patterns in a machine-readable format have been inconsistent and fail to provide consistent coverage of dashboard elements, non-visualization features (e.g., text, widgets, multimedia), interaction, and layout.

We summarize these challenges as an abstraction gap between the low-level programmatic specifications of the dashboard and the resulting design and higher-level emergent patterns. Prior research, such as work by Bach [8], Hu [9], Li [13], and Lin [10] are mid-level abstractions that emphasize different elements of dashboard designs. We argue that these abstractions are not interoperable, nor do they fully cover the range of possibilities illustrated in Figure 1. For these reasons, we propose an integrative schema that defines dashboard designs as blocks and connections.

B. Proposed Schema: Blocks and Connections

We present a node-link (graph) representation of dashboard designs that comprise ***blocks*** (nodes) that contain content elements and ***connections*** (edges) that capture the composition of these elements in a dashboard. This schematic representation integrates aspects of prior work that examines dashboard content qualitatively [8], [7], individual visualizations at scale [9], [13], and smaller collections of dashboard corpora [50]. We now describe these components of the node-link representation and how it can be used to capture design patterns, including interaction, in dashboard corpora.

Blocks represent individual content elements of a dashboard. Blocks do not only include visualizations but can also represent text, legends, filter widgets, and multimedia

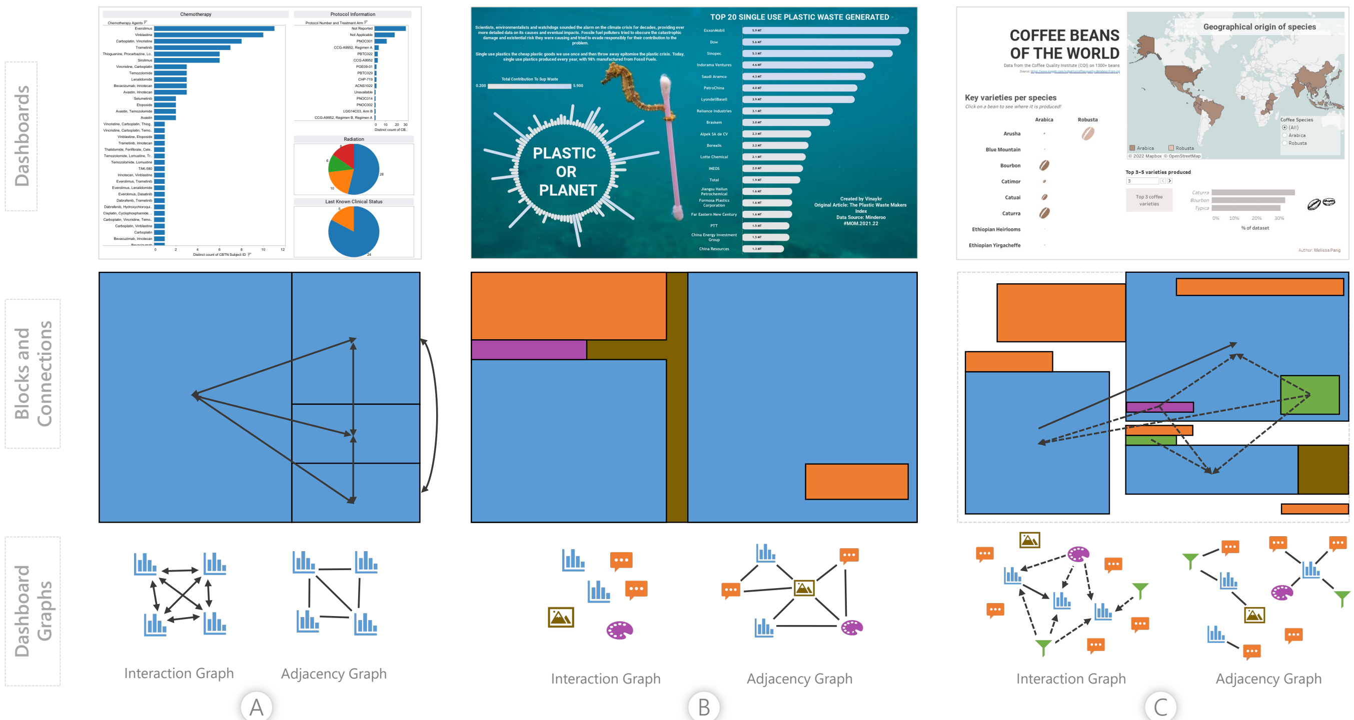


Fig. 1. Example dashboards are decomposed into block-connection form and reconstructed as interaction and adjacency graphs. Blocks are represented via icons and colored by type (chart, legend, filter, text, multimedia). Directed edges show interactions between blocks, while undirected edges indicate that two blocks are adjacent. Block adjacency is algorithmically computed, based upon how elements are laid out and spaced in the dashboard (see Section IV-C). Solid links in the block diagrams and interaction graphs represent chart → chart interactions, and dashed links represent filter - - → chart or legend - - → chart interactions. Since (B) has no interactive blocks, its interaction graph contains no edges. Likewise, the adjacency graph in (C) is disconnected due to the usage of white space between blocks.

elements such as images or embedded web pages. We note that prior research (see Section II) primarily captures visualization elements without consideration of other elements that may exist in the dashboard (Figure 1). The precise composition of a block can be flexibly defined based on the desired level of granularity. For example, a faceted chart can be represented by a single block (as we do in Section IV) or as multiple blocks representing each facet. Each block has a set of properties that can be ascribed to it. All blocks contain *positional properties* that capture their spatial position in the dashboard as coordinates, size, and aspect ratio. Blocks also contain *descriptive properties* based upon the content type. For example, visualization elements can contain sets of features described, such as those described in VizML [9]. Text or images will contain different sets of properties (e.g., topics, semantic aspects). Importantly, the richness of descriptive properties can vary across elements of the same type – some visualizations have richer features than others.

Connections capture relationships between blocks to represent their composition in a dashboard. Two of the primary types of connections that we focus on here are layout and interaction. Layout considers the position of blocks in a dashboard. We can establish a connection if blocks share a common edge or overlap spatially (e.g., the bar chart on the left in Figure 1A is adjacent to all other charts, the text block on the bottom-right corner in Figure 1B is overlain on the bar chart). Interactions with one block that influence others (e.g., cross-filtering) establish interaction connections. Just as with blocks, additional type-specific properties can be utilized to capture supplemental information, such as the interaction type (e.g., filter or highlight). Other types of connections can be considered, for example, shared dimensions across elements, such as a data attribute used across multiple visualizations and referenced in text elements.

We now describe how blocks and channels can be organized into one or more graphs that represent the design of a dashboard. Here, we propose the use of two graphs to represent layout and interaction connections between a common set of nodes. The ***adjacency graph*** is an undirected graph that codifies the spatial layout blocks of a dashboard according to their positional properties and connections. The ***interaction graph*** is a directed graph showing how blocks influence each other through interaction connections. These two graphs can be jointly analyzed for a holistic analysis of design patterns [51]. *While it is technically possible to represent multiple types of connections on a single graph, including layout and interaction, we recommend against this for the following reasons*. First, interactive elements have directions, and omitting them can result in a loss of information. For example, prior qualitative research [8], [43] presents design patterns like ‘drill-down,’ which can be identified by proper consideration of interaction directionality. Second, edges can have different meanings, and rather than overloading edge properties, it can be more useful to represent them separately.

Prior work uses a single graph because they model just one type of connection between dashboard elements (e.g., DMiner [10]) or model something unrelated to dashboard design (e.g., KG4VIS [13], VizML [9]).

C. Leveraging Dashboard Graphs for Distant Reading

We now describe the ‘distant reading’ affordances enabled through a blocks and connections lens of dashboard patterns.

Blocks and connections enable a consistent decomposition of dashboard design. In Figure 1, we show how three different types of dashboards are represented using the block and connections representation. Figure 1A displays a classic multiple coordinated views style of a dashboard for interactive exploration of cancer treatment statistics. Figure 1B showcases a single-use plastic and its impact on the ocean. It is an example of using additional elements besides visualizations. Finally, the dashboard in Figure 1C displaying data on coffee beans around the world shows the diversity of blocks and connection types. By applying our schema, we can quickly spot and compare several pertinent design considerations between these different dashboards. For example, Figure 1A and 1C have many interactions between elements, while Figure 1B has none. Moreover, the types of interactive connections (cross filtering between visualization (→) vs filtering via widgets (- - →)) are different between these two examples. Figures 1B and 1C have distinct cliques, grouping common information, whereas Figure 1A does not. In aggregate, these kinds of assessments establish design patterns and their prevalence.

Blocks and connections are flexible and extensible. By focusing on the core elements of dashboards and how they relate to one another, our schematic representation generalizes to the most common dashboard designs

Our approach uses blocks and connections to represent dashboards in a way that is independent of how these elements were created. This method is versatile, capturing a wide range of design elements, including visualizations, text, and even non-visual elements like web pages. It applies to any dashboard, regardless of whether it was manually designed, automatically generated, or programmatically specified using visualization language (Figure 2). For dashboards in other formats, such as print, alternative techniques like image segmentation [52], [40], [41], [43] can be used to identify elements. The strength of our schema lies in its ability to provide a consistent representation across different tools.

Enabling scalability for large dashboard corpora. Graph data structures offer numerous methods for scaling the analysis of large corpora. Prior analyses of dashboard corpora have not presented the design patterns of dashboards in a way that allows us to leverage methods for scaling the analyses of corpora. Still, a rate-limiting factor is the data preparation necessary to convert bespoke representations of dashboards, be they images or code for different programming languages, into a graph schema. In Section IV, we demonstrate this process using Tableau Public, with our graph schema serving as the target for data preparation. Our analysis in Section V then highlights insights into dashboard design patterns.

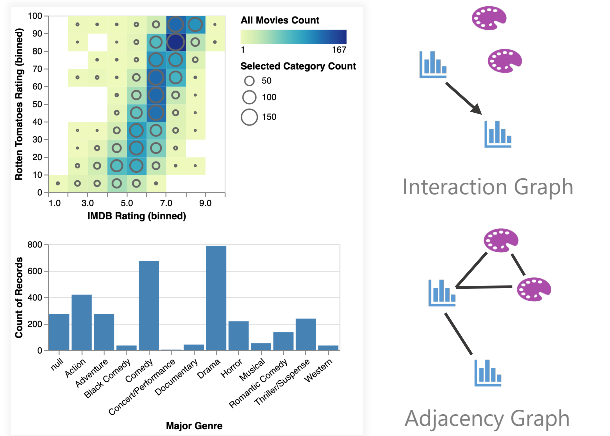


Fig. 2. An example of interaction and adjacency graphs extracted from a dashboard specified using Vega-Lite. Such graphs can be used to collectively analyze dashboards agnostic of creation methods or tools.

IV. Case Study: A census of Tableau Public

In this section, we apply the blocks and connections graph schema to generate a census of a dashboard corpus derived from Tableau Public. We define a census as a survey of a dashboard corpora. Thus, our goals are *descriptive* in nature. We describe our process for deriving and analyzing blocks and connections, as well as the attendant adjacency and interaction graphs. We describe layout and interaction design patterns that we observe and their prevalence.

A. Motivation and Research Questions

Surveying user-created artifacts can provide valuable insights into what information is most important to users, how they represent this information, and how they organize it. A survey can also suggest gaps in support for further investigation and follow-up. In this case study, we survey dashboard artifacts derived from Tableau Public [2], a well-established platform that supports the authoring and dissemination of dashboards. In total, it comprises approximately 5 million workbooks created over a 14-year period. The full workbook corpus represents a wide range of uses across multiple domains, including public health, finance, journalism, and others. Moreover, it attracts a wide variety of end-users, from students to journalists to data professionals, and captures dashboard designs ranging from complex and interactive to simplistic and static. In addition to the diversity of dashboard patterns, the choice of Tableau Public is also pragmatic. Like prior research [9], [10], we can make use of a common specification format to simplify the processes of deriving blocks, connections, and their type-specific properties or features. In Section III-C we provide an example of how dashboards specified with different tools can be analyzed once they are translated into our schematic representation.

We derive a census and present an analysis to answer the following research question on dashboard designs:

[2]https://public.tableau.com/app/about

- **RQ1:** What constitutes a visualization "block" within dashboards and what are the spatial relationships between visualization "blocks" and other types of "blocks"?
- **RQ2:** Is interactivity common in dashboards, and are there common patterns of interactions between "blocks"?
- **RQ3:** Can we detect and characterize high-level dashboard design patterns?

The first two research questions aim to provide a descriptive overview of the composition and arrangement of design elements within a dashboard. We place so-called visualization "blocks"' at the center of our analyses and seek to get an overview of what other design elements appear alongside them (e.g., text, multimedia) and how they influence each other (e.g., cross-filtering). The third research question examines to what extent these individual dashboard designs can be clustered to reveal design patterns and their prevalence.

B. Preparing the Corpus

Analyzing the full corpus of 5 million workbooks is infeasible. The primary reason is that workbook specifications have changed over time, impacting how visualizations and dashboards are defined. Workbooks can also be inaccessible due to user-set permissions or deprecation. Moreover, not all workbooks contain dashboards, and among those that do, many can be low quality. We describe winnowing the corpus to arrive at a subset for our dashboard census in Figure 3.

1) Winnowing: From the total corpus of 5 million workbooks, a total of 1,342,794 workbooks (∼25% of all workbooks on Tableau Public) had been published or recently updated to conform to a contemporary Tableau work version; this addresses the issues of older workbooks. Users can create one or more dashboard objects within a Tableau workbook. Using Tableau's definition, only 150,276 (11%) contained at least one dashboard. An initial exploration of these dashboards revealed that many were very low quality – often containing a single visualization (typically a just bar chart). We hypothesize that these dashboards may represent just trial and error exploration of using the Tableau Public platform.

To increase the likelihood of higher-quality dashboards, we used page views (how often a dashboard is viewed by someone on the internet) as a surrogate metric. We observed that the distribution of total impressions across the 150,276 workbooks was left-skewed with a heavy tail with values ranging from just one impression per workbook to over 32 million; impressions did not strongly correlate with the publication date. Given this distribution, we elected to sample the top 10% of workbooks based on impressions (≥42 impressions), yielding a set of 15,090 workbooks that contained 42,951 Tableau dashboards.

2) Extracting Valid Dashboards: We opted to further limit the corpus to dashboards that had *two or more* visualizations elements, allowing us to enhance the possibility of multiple coordinate views. Applying this criterion resulted in a final set of 25,620 dashboards that fit prior definitions of the term as commonly used in visualization research (e.g., "*a visual data representation structured as a tiled layout of simple charts and/or large numbers*" [7]). **We provide the anonymized version of this dataset at https://osf.io/r5cfk**

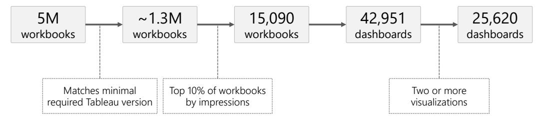


Fig. 3. An overview of the dashboard winnowing process

C. Deriving Dashboard Block and Connection Graphs

We now describe how we extracted and defined blocks and connections from existing Tableau workbook specifications.

1) Overview of Workbook Specifications: Workbooks are XML documents that, among other things, contain specifications for visualizations and dashboard elements. Within Tableau, individual data visualizations are constructed in `worksheets` by dragging and dropping dataset attributes onto so-called "shelves" (i.e., row, column) or to specific encoding channels (i.e., color, size, etc.). A visualization is automatically suggested or user-specified by selecting a mark type. `Workbooks` can contain one or more `worksheets`. A `dashboard` is composed of one or more worksheets that can be arranged in a grid (default) or fluid layout. Regardless of the layout, all dashboard content is captured as a `zone`. The contents of a `zone` need not be a visualization but could also contain text, images, or layout elements of the dashboard. Finally, a user can specify `actions` that add interactivity between dashboard zones, including highlighting, (cross-)filtering, and page navigation.

2) Detecting Blocks: We analyze workbook `dashboards`, not individual `sheets`, to define and extract blocks. Specifically, we parse the zone objects to derive five block types: charts containing visualizations, filters containing widgets like dropdown menus and sliders, legends displaying data mappings for graphical encodings like size and color, text blocks including the dashboard title, caption, or additional commentary, and finally, multimedia blocks containing images or embedded web pages.

The chart block type links to the original worksheet that describes the data visualization, which we use to extract the visualization type (e.g., bar chart, map, scatterplot, treemap, Sankey diagram) from the specified marks (e.g., bar, line, circle) and encodings (e.g., row, column, color). We capture additional properties of the block, such as its spatial coordinates in a dashboard and the data attributes for visualizations.

3) Deriving Layout Connections: We establish layout connections between blocks by determining their spatial proximity and adjacency within a dashboard. We first construct a bounding box around each block from its spatial coordinates and size (width, length). Whether a dashboard uses a grid or floating layout affects how we establish whether two blocks are adjacent. In grid layouts, blocks can be placed side-by-side, either above, below, or on either side of another block. In floating layouts, the position of a block is more flexible, and blocks can be placed on top of each other. We enumerate all pairs of blocks and classify into four configurations:

- *Partial Overlap.* In a floating layout, two blocks may partially overlap, but neither block is contained entirely within the other.

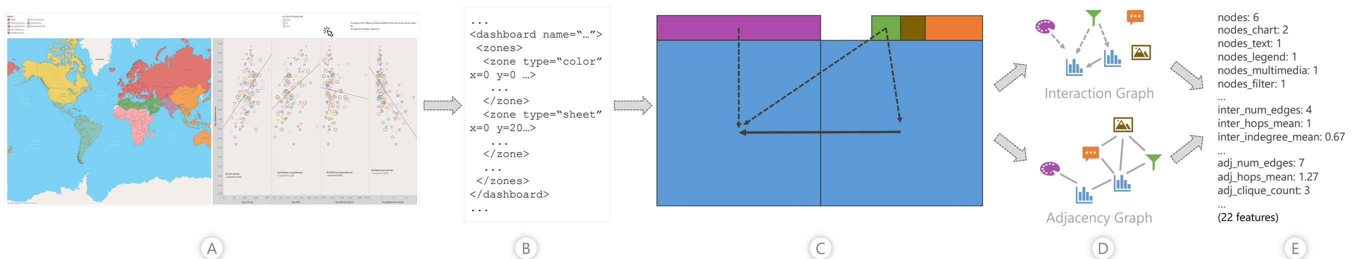


Fig. 4. An overview of the feature extraction process. Given a Tableau dashboard (A), we parse the underlying XML specification file (B) to detect the different blocks and connections between blocks (C). We then model two graphs depicting the interactive and spatial configurations of the dashboard (D). From these graphs, we extract 22 features that we use for our analyses (E).

- *Containment.* In floating dashboard layouts, one block can be contained entirely within another. For example, a text block may be contained entirely with a chart block when it is used to annotate an outlying mark in the data visualization. In this scenario, the coordinate range of one block entirely overlaps with its pair.
- *Adjoining.* Primarily, in grid layouts, two blocks can share an edge when adjacent to one another (e.g., two chart blocks containing different visualization types could be placed next to one another). Compared to partially overlapping blocks, these adjoining configurations have very limited coordinate overlap, often a few pixels, and require separate treatment to be accurately detected.
- *Non-adjacent.* A pair of blocks were established not to be adjacent as they shared no related spatial coordinates.

To allow for flexibility in determining adjacency, we use a tolerance criterion of 10 pixels that allows two blocks to be positioned a very small distance apart (no shared coordinates) but still be considered adjoining.

4) Deriving Interaction Connections: Finally, after detecting the dashboard blocks, we extract `actions` from the XML specification to define interaction connections between blocks. Each action provides the interaction type (e.g., filter, highlight) as well as the source and target blocks in the dashboard that we use to record connections. The `action` specification establishes whether there exists cross filtering between blocks that contain visualizations (chart → chart), or visualization is filtered by another type of block, for example, a filter (filter --→ chart) or a legend widget (legend --→ chart).

5) Constructing Adjacency and Interaction Graphs: Having extracted blocks and establishing the structure of their layouts and interactions, it is then simple to construct the adjacency and interaction graphs. In both of these graphs, the blocks are nodes. In the adjacency graph, undirected edges between these nodes are formed when pairs of blocks have either partial overlap, containment, or adjoining adjacency. In the interaction graph, edges are directed and formed between nodes where some interaction has been established between blocks. In both graphs, we check whether there exist duplicate edges and self-loops and remove them.

D. Methodology for Deriving a Census

The graph representation provides a consistent description of dashboards that we can use to conduct our census – much like a common set of questions is used to conduct a census of human populations. Moreover, much like a census of people we can aggregate over individual results to get an overview of a population, or in our case a dashboard corpora.

1) Census Summaries via Descriptive Statistics: To address **RQ1** and **RQ2** we conduct a descriptive statistical analysis. We enumerate the total number of blocks and block types across all dashboards and describe their distribution (via median and mode). We also summarize the co-occurrence of block types within a dashboard. We do so by applying a clique-detecting algorithm to the adjacency graphs of each dashboard. We then enumerate and sort commonly occurring cliques by their prevalence. We also apply descriptive statistics to understand the extent of interactivity in dashboards and the prevalence of interaction between visualizations and other types of blocks.

Having a descriptive overview of the blocks has several uses. First, it can be used to identify common and recurrent structures. These can represent design patterns that multiple users find useful because they either independently arrive at the same choice or borrow it from others. Second, by focusing on prevalence, we can see the diversity of dashboard content via blocks. Low diversity may signal pain points. Finally, understanding different patterns and their diversity can enable judgments on the suitability of the corpus, or some subset of it, for downstream tasks.

2) Summarizing Design Patterns via Cluster Analysis: We also explored whether there were emergent dashboard design patterns (**RQ3**). Specifically, we used the features derived from the graph representations (Section IV-D) to perform an unsupervised cluster analysis using the hierarchical density-based clustering (HDBSCAN) [53] algorithm. In the supplemental materials, we provide more details on our choice of clustering algorithm (i.e., the choice of HDBSCAN vs K-means) and a sensitivity analysis for our parameter choices.

Ahead of clustering, we derived a set of 22 features from the properties of nodes and the topological structures of the adjacency and interaction graphs, summarized as follows:

- *Descriptive Features.* The number and the types of blocks within both the adjacency and interaction graphs are identical, allowing us to extract a common set of features from both. These features include the total number of blocks in each graph and the presence of specific block types. We use one-hot encoding to represent the presence (and absence) of block types. We standardize the total number of blocks by the mean degree, by its mean, and by unit variance (standard scaling). We also summarize the total number of edges and the mean degree of nodes and apply the standard scaling transform.

- *Adjacency-Specific Features.* We derived features that add context to the spatial layouts. For all pairs of blocks in a graph, we compute the average shortest path. As an additional proxy of layout complexity, we examine adjacency graphs for the presence of one or more *maximal cliques* and, when detected, compute the average size of all cliques. We apply the standard scaling transformation to the path lengths, number of maximal cliques, and mean clique size.
- *Interaction-Specific Features.* For the interaction graph, we compute the average in-degree and out-degree of the nodes, also applying a standard scaling transformation. We tabulate the presence of the three edge types (filter --→ chart, legend --→ chart and chart → chart) that describe the interactive relationships between two blocks; these edge types are also one-hot encoded.

There are a variety of different features that can be derived to provide a different or simply more nuanced lens of visualization design patterns. For example, VizML [9] presents a list of visualization-specific features that we could have used here. The features we have chosen emphasize the topological characteristics of the adjacency and layout graphs. We argue this is more reflective of the level that design patterns are explored in prior qualitative research [8], [7]. However, by making our dataset and analysis available, others may derive alternative analyses that answer different questions.

V. Results

In this section, we provide an overview of what our census reveals, according to the descriptive statistical analysis and feature-based unsupervised clustering. We present these results in accordance with our research questions.

A. Content and Composition of Dashboards

Addressing **RQ1**, we examine the visual composition of dashboards focusing on blocks and their spatial relationships.

1) What are the visual components of a dashboard?: We identified a total of 250,794 blocks across the 25,620 dashboards. The number of blocks per dashboard ranged from 2 to 267 (median: 8, mode: 4). We observed that 121,068 out of 250,794 blocks (49%) were charts, followed by 53,267 text blocks (21%), and subsequently filter (36,472 or 15%), legend (22,446 or 9%), and multimedia (17541 or 7%). We note the importance and centrality of text and charts as the building blocks of dashboards: together, over 70% of the blocks in our analysis were one of these two block types. Figure 5A shows these results in detail.

For the chart blocks, we also examined the distribution of visualization types across dashboards. We found that bar charts were the most common visualization type, appearing in 15,392 out of 25,620 (60%) dashboards. The next most frequently used charts were line charts (n=6,524; 25%), maps (n=6,454; 25%), and finally, tables (n=6,154; 24%). Besides other simple chart types (e.g., scatterplots, bar charts), there were also instances of more bespoke visualizations, such as Sankey diagrams and waterfall charts, but they were present in only 116 dashboards (<0.5%). These findings show that only a very small subset of our corpora constitute sophisticated designs that are often explored in 'close reading' of dashboard corpora. Our census suggests that authors generally create very simple dashboards constituting two or three simple charts.

Considering the distribution of blocks and block types across dashboards, we can derive several important takeaways. First, text plays a prominent role in the construction of dashboards. Note that text related to titles or axes labels of visualizations are retained within a charts block; thus text blocks are deliberately added to include additional information. The amount of text and the number of text blocks was highly variable amongst dashboards. Another finding was the high prevalence of multimedia elements, primary images, that accompany dashboards. Images are often contextually related to the dashboard content. It is noteworthy that prior research on dashboard designs [7], [8] and recommendation algorithms [10] emphasize what we would call charts blocks to the exclusion of other widely used types.

2) What are common structural relationships between visual components in a dashboard?: We analyzed the adjacency graphs for the 25,620 dashboards to identify potential design patterns around block layouts. The graphs had 2-267 nodes (median: 8, mode: 4) and 0-4926 edges (median: 11, mode: 3). For each graph, we extracted the list of all maximal cliques, capturing the block type for each node in a clique (e.g., {chart – chart – chart}, {chart – filter}, {multimedia}). Aggregating these maximal clique patterns across all graphs, we found a total of 1,430 unique block patterns. The smallest-sized cliques contained just 2 blocks, and the largest contained 60, and the median clique size was 9. The most frequently occurring clique patterns contained just two blocks, typically including a chart and one of the other block types; a summary of the most frequently occurring patterns is in Figure 6A.

Focusing on chart blocks specifically, we also examined the creation of juxtaposed views. Of all 1,430 unique clique patterns, more than half (n=747) contained a spatial arrangement of two or more charts; we list the five most frequently occurring clique patterns in Figure 6B. The most dominant patterns are cliques containing only chart block types, varying in size from 2 to 4. When two chart blocks occur with other blocks, it was more common for those other blocks to be a text or multimedia block.

The clique analysis shows that *like* is often juxtaposed with *like* in dashboards: common dashboard elements are visually grouped together. As with the aforementioned block composition analysis, the results emphasize simpler dashboard

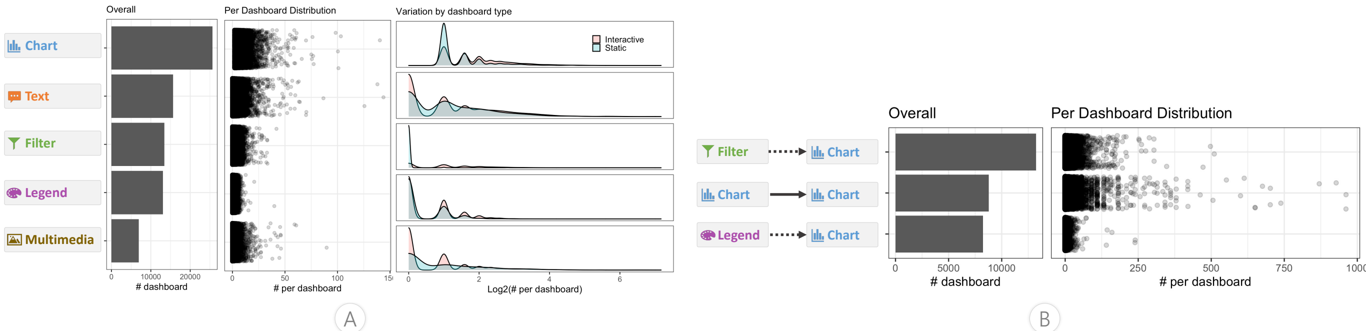


Fig. 5. Summary statistics for (A) distribution of block types across dashboards and (B) distribution of interactive connection types across dashboards.

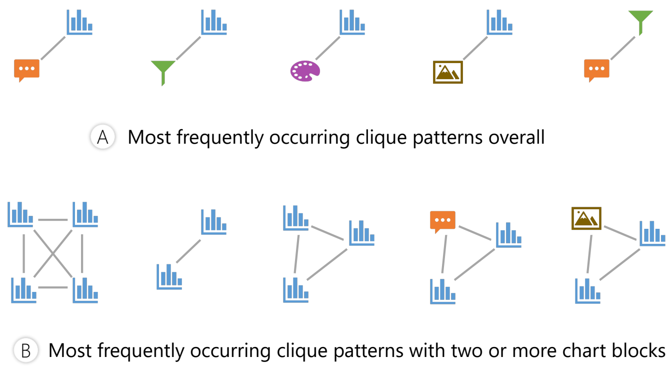


Fig. 6. Frequently occurring clique patterns from the adjacency graph analysis. Nodes map to different block types including chart, text, filter, legend, and multimedia.

designs but reiterate that authors do experiment with more complex designs that mix block types within close spatial proximity, such as having text or multimedia blocks that are connected to multiple charts.

B. Interactivity in Dashboards

Complementing our analysis of the visual components of dashboards, we also analyzed the connections between blocks to understand what types of interactions dashboards commonly support (**RQ3**). Note that we refer to a dashboard as *interactive* if clicking on one block updates another block (e.g., by filtering, highlighting, or changing visualized data fields), as opposed to other forms of interactivity that only involve individual blocks (e.g., hovering over a point in a single chart to generate a tooltip).

1) Is interactivity common in dashboards?: We found that 19,304 of the 25,620 dashboards in our corpus (75%) were interactive and that their design patterns varied considerably. In particular, the number of interaction connections between blocks ranged from 1 (e.g., a filter acting as a control for a single chart) to 992 (median: 6, mode: 2) interaction connections in a dashboard. Recall that a single block can be the source or target of multiple interactions. The maximum possible interaction connections in a Tableau dashboard are bound by (charts $-$ 1 $+$ legend $+$ filters) $*$ charts. On average, 58% of the possible interactions between blocks were applied (median: 50%, mode: 100%), suggesting that when authors add interactions, they tend to make a considerable portion of the dashboard interactive.

2) How does interaction commonly manifest?: Of the three Tableau blocks that support interaction connections (chart, filter, and legend), filter --→ chart was most common with 13,228 out of the 19,304 interactive dashboards (69%) supporting this style of interaction, followed by chart → chart (46% of dashboards) and legend --→ chart (43% of dashboards). These results are summarized in Figure 5B and suggest that there are multiple strategies for interaction design or entry points for users of an interactive dashboard.

Collecting both spatial adjacency and interactivity in graph structures sharing common nodes allows us to assess the relationship between these two factors. While one might assume that the blocks that *control* a particular portion of the dashboard would be spatially next to each other, we found that this was not always the case. In fact, out of a total of 343,929 interactions, only 105,317 (30%) were in cases where blocks were adjacent. Out of these 105,317 adjacent + interactive connections, 58275 (55%) were chart → chart interactions, 37186 (35%) were filter --→ chart, and the remaining 9856 (10%) were legend --→ chart interactions. This distribution suggests two broad genres or patterns of interaction design: one was cliques or tightly connected subgroups of charts mutually interact and another "light switch" style where a control panel with filters and legends interacts with many if not all charts on a dashboard.

C. Characterizing Clusters of Dashboard Design Patterns

Finally, to explore the design patterns across our corpus we conduct a clustering analysis (see Section IV-D2 for methodological details and parameter sensitivity analysis). We analyzed the features and topological relationships from the adjacency and interaction graphs and applied HDBSCAN to derive clusters. As a reminder, unlike k-means (a commonly used clustering method), HDBSCAN does not force each dashboard into a cluster, which can mean that the resulting cluster has more consistent design patterns; unclustered items may be outliers, have too few examples, or might exist resemble two clusters. Our analysis identified 16 clusters

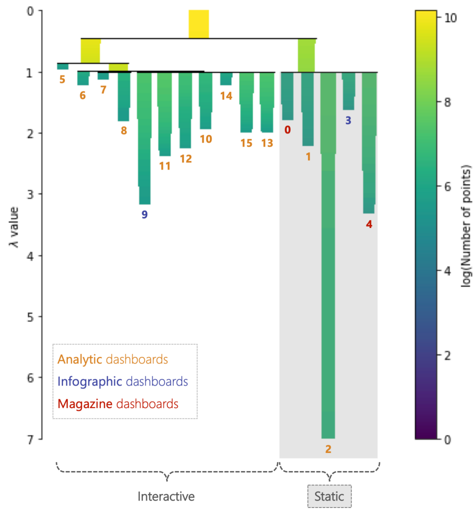


Fig. 7. Dendrogram generated by the HDBSCAN algorithm summarizing cluster hierarchies. Cluster IDs are colored to match the design patterns they were mapped to. The distinct split between clusters 0-4 on the right and clusters 5-15 on the left illustrates that the algorithm picked up on the presence (or absence) of interactions as a salient feature for clustering.

covering 15,013 out of 25,620 (59%) dashboards, while the remainder was flagged as noise. In Figure 8A, we summarize the characteristics of each cluster according to their total number of blocks, the prevalence of non-visualization blocks, and their connection types. As a reminder, our inclusion criteria requires that dashboards contain at least two or more chart blocks. To convey the cohesiveness of each cluster, we compute the silhouette score. A score of one indicates that dashboards all have exactly the same design (which would be a concerning finding). The scores in Figure 8, however, show that, while sharing common elements, there is diversity *within* each cluster, but this is less than the diversity *between* clusters.

1) Delineating Dashboard Design Patterns: Similar to Bach *et. al.* [8] and Sarikaya *et. al.* [7] we used the composition of the dashboards, via automated clusters and then qualitative coding, to identify design patterns (genres) of dashboards. In Figure 7 we show the results of the clustering analysis and in Figure 8 we provide an overview of their characteristics according to our census dimensions. The first major distinction between patterns is static (18% of dashboards in our corpus) and interactive (82%); these numbers have a similar distribution to those reported by [8]. We then examine clusters and based on the summary statistics of each (Figure 8) we attempted to apply the classifications from either Bach *et. al.* [8] and Sarikaya *et. al.* [7]. In general, we found that without additional user interviews, it was too difficult to ascribe a specific intent of the dashboard (e.g., dashboards for motivation and learning, dashboards for decision making) that were described by Sarikaya *et. al.* [7]; this could be a fruitful avenue for future work. Our findings aligned more composition and layout patterns described by Bach *et. al.* [8], specifically, *Analytic*, *Magazine*, and *Infographic* styles. There was not a precise linear relationship between the definitions described by Bach *et al.* [8] and our clusters. For example, we use multimedia blocks to align with the pictograms and gauges that characterize the infographic style in [8].

Magazine dashboards (n=1,913; 7.4%) predominately static and typically include multiple text blocks that complement the charts and provide additional commentary about the data and key takeaways. While, ***infographic dashboards*** (n=1,125;4.4%) generally include a richer mix of block types, including at least one multimedia and text block in addition to charts. One important difference between our characterization of infographic and multimedia is the prevalence of text, which is not discussed in either [8] or [7]. ***Analytic dashboards*** (n=22,582; 88%) were much more interactive but did not use multimedia elements as regularly (except cluster 6) and had text elements that were less verbose compared to the magazine and infographic patterns. Analytic dashboards exhibited more cross-filtering between charts (see Connection types in Figure 8), conforming to the definitions of this genre from [8]. One exception is cluster 2, which contains only visualizations and has no interactions, but still constitutes approximately 6% of dashboards in our corpus; these may be proto-analytic dashboards, reflecting either initial design explorations or be indicators of difficulties adding other elements or interactions.

2) Content prevalence and design patterns: Our cluster analysis shows that we can use our graph schema to apply genres and patterns from prior research via automated methods. Our analysis can also go further and reveal variations within the broader genres, represented as subclusters in Figure 8, which adds context to the prevalence of elements in dashboards – something that prior work does not do at scale. This is an example of how a design census adds perspective to qualitative surveys of dashboard “zoos”. One way that we interpret our results is they suggest that, in practice, dashboard designs are simpler than some of the examples covered in prior work [8], [7]. We arrive at this conclusion by considering the prevalence of content elements across dashboards (Sections V-A1 and V-B1 and their composition into higher-level patterns, which suggest that many dashboards are quite simple because they contain limited interactions and have relatively few dashboard elements including visualizations. While neither Bach et al. [8] nor Sarikaya et al. [7] make explicit claims about the variability or complexity of dashboards within each genre, the absence of such information and the use of more “charismatic” examples, could produce a false impression of what users actually do. While Tableau does impose a learning curve and constraints on the authoring experience, it may be that simplicity is desired. Alternatively, users do not know how to compose more complex dashboards that could eventually meet their needs, and our census captures this issue. While there exist power users that can create rich and complex dashboards—the kinds likely to be studied in prior research—our findings suggest users are the exception and not the norm. Authoring experiences catering to these power users neglect the majority of the population.

VI. Applications for a Dashboard Census

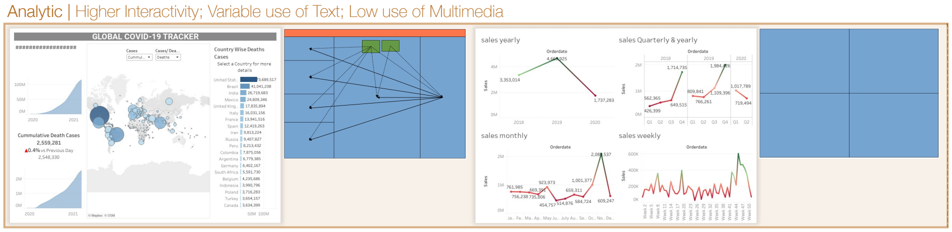
We return to the original premise we articulated in Section IV-A that a census of dashboard design patterns helps

A) Cluster Characteristics

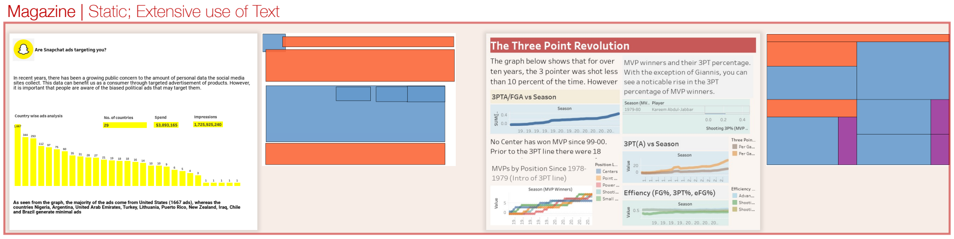
Design Pattern	Cluster ID	Cluster Overview: Cluster Size	Silhouette Score	Median # **Blocks**	Median # **Interactions**	Non-Chart Block Types: **Text** %	**Legend** %	**Filter** %	**Multimedia** %	Connection Types (Median #): legend → chart	filter → chart	chart → chart
Analytic	[S] 1	520	0.24	4	-	0%	100%	0%	0%	-	-	-
	[S] 2	1638	0.26	3	-	0%	0%	0%	0%	-	-	-
	5	344	0.17	6	4	19%	100%	66%	0%	2	2	0
	6	500	0.17	11	6	100%	100%	100%	100%	1	4	0
	7	321	0.17	5	3	100%	0%	0%	0%	0	0	3
	8	633	0.11	3	2	0%	0%	0%	0%	0	0	2
	10	1467	0.11	8	6	100%	100%	100%	0%	1	4	0
	11	865	0.18	6	2	100%	100%	0%	0%	2	0	0
	12	1377	0.10	7	4	100%	0%	100%	0%	0	3	0
	13	1124	0.19	4	4	0%	0%	100%	0%	0	3	0
	14	1409	0.22	4	2	0%	100%	0%	0%	2	0	0
	15	1777	0.11	6	6	0%	100%	100%	0%	1	3	0
Magazine	[S] 0	452	0.22	6	-	100%	100%	0%	0%	-	-	-
	[S] 4	1461	0.14	5	-	100%	0%	0%	0%	-	-	-
Infographic	[S] 3	638	0.18	10	-	100%	0%	0%	100%	-	-	-
	9	487	0.16	10	4	100%	0%	100%	100%	0	4	0

B) Summary and Examples

Analytic | Higher Interactivity; Variable use of Text; Low use of Multimedia



Magazine | Static; Extensive use of Text



Infographic | Some Interaction; Extensive use of Multimedia

Fig. 8. A) Characteristics of Analytic, Magazine, and Infographic design patterns distributed amongst sub-clusters. N is the total number of dashboards in the cluster, % values refer to the percentage of dashboards containing an element (e.g., in cluster 6, 100% means all dashboards contain have text) B) Examples and summary characteristics of each design pattern with our graph schema representation.

visualization researchers understand user practices and support gaps via further investigation. To lay the groundwork and help foster ideas for future research and development, we describe some exemplary use cases and applications of the blocks and connections graph schema in addition to the dataset.

A. Application of a Census Dataset and Findings

The census dataset itself and its results can be used for different downstream research objectives. Our primary interest is in the use of this dataset to inform the development of dashboard authoring tools. However, our findings can be used to highlight gaps in need of further investigation, such as user onboarding and authoring support, in addition to providing more insight into existing practices, like intent inference, recommendation, and guidance. While the utility of this data ultimately depends on richness and appropriateness for the task at hand, having a census remains an important precursor for making such a determination.

Developing Better Onboarding Techniques to Support Interaction. Our census shows that 75% of the dashboards in our corpus were interactive and supported using at least one block to visually update one or more other blocks. We also found that interactions are configured between different pairs of blocks (filter --→ chart, chart → chart, legend --→ chart) and are not always consistently used in a dashboard (e.g., filters may only update a subset of charts). While this nuanced configuration of interactions suggests that current tools provide a rich set of features to author interactivity, it raises questions about the discoverability and usability of such dashboards from a viewer standpoint. One important direction for future systems is to incorporate built-in strategies to improve the dashboard onboarding process [54] and orient viewers to a dashboard and its use (e.g., through overlaid walkthroughs or using assistive tooltips and guiding text). Notably, the graph schema can also be valuable for designing features such as the underlying links that can be used to identify the flow of actions between blocks.

Prioritizing support for customization of simple charts over authoring bespoke charts. As stated in Section V-A1, our analysis showed that basic visualizations, including bar charts, maps, line charts, tables, pie charts, etc., are more prevalent in dashboards with only a minor subset of dashboards containing bespoke charts. However, while inspecting examples during the cluster analysis, we observed that the absence of bespoke charts did not dampen the "richness" of dashboard designs and that authors either heavily formatted basic charts or combined basic charts such as maps and pie charts in innovative ways to create visually compelling designs. Unfortunately, creating such highly stylized and custom designs with tools like Tableau can require substantial expertise, forcing new or novice authors, in particular, to resort to default visuals or integrate artifacts across design and visualization tools. As the user base for dashboard design tools broadens, it is more important to provide more expressive and flexible authoring interfaces for formatting (including using images and icons as marks [55]) and integrating basic charts than to focus on allowing dashboard authors to create and incorporate more bespoke visualizations.

Enhancing dashboard authoring through AI/ML supported intent inference, recommendations, and guidance. A large corpus of dashboards is a useful dataset for training or fine-tuning AI/ML models. We examine several useful applications based on the content of our corpora. Prior work has shown that dashboards are generated for a variety of purposes and that intents play an integral role in dashboard design [7], [56]. However, the process of inferring dashboard intent in their work has been largely qualitative and performed at a small scale. For instance, Pandey et al. [56] derive dashboard intents such as "change analysis" and "category analysis" by manually inspecting the views, filtering widgets, and textual content of 200 dashboards. Our schematic representation and dataset present an opportunity to investigate this idea at scale and explore how a combination of information from blocks (charts, text, and filters) and their connections can be used to programmatically infer dashboard intent. Intent is also to guide recommendations and guidance that is tailored to the user needs at the present moment of their analysis [57].

However, our census points to both optimism and caution when using large dashboard corpora, ours, or others that may emerge in the future. Given that the distribution of dashboard designs skews toward simplicity, the signals for user intent can be washed out. Instead, it may be more appropriate to use a census to guide a principled selection of dashboards to form a training or fine-tuning dataset. Further, the left-out, simpler dashboards still serve a critical purpose: they become useful boundary markers delineating the minimum viable content and composition necessary to derive useful recommendations.

B. Application of Blocks and Connections Graph Schema

The blocks and connections graph schema is simple and extensible and can easily incorporate features from prior work, such as by making their properties of blocks or new features that others wish to study, or that emerge as dashboard authoring practices change. Graphs, more generally, are flexible data structures that have an attendant analytic toolbox that can enable practical and informative assessments of user practices.

Improving Support for Non-Chart Blocks. Dashboards have conventionally been considered as visual analytic artifacts composed predominantly of multiple coordinated views [6]. However, our analysis shows that, in practice, non-chart blocks such as text and multimedia (e.g., images) play an integral role in dashboard design. An important consideration for future dashboard tools is to provide ample support for authoring and incorporating such content in flexible ways. The prevalence of text, in particular (Figure 5A), also hints at potential synergies with recent research on interactively linking text and charts [58], [50], [59], and presents an opportunity to further explore this relationship.

Having better support for non-visualization blocks can also enable a richer analysis of dashboard content. One reason we examine a limited set of features is that visualization blocks tend to have more and richer features than, for example, text or images. The visualization blocks would have dominated the analysis. Knowing that users create other types of blocks can prime future dashboard authoring tools to prompt the user for richer, non-visualization data or to suggest options to users. Our census suggests that, at present, only power users are likely to make full use of these features.

Developing Dashboard Linters for Layout and Interaction designs. Poor choices in dashboard design or layout can produce designs that are confusing or even misleading. In particular, Qu & Hullman [20] suggest that keeping multiple views "consistent" is important for the legibility of dashboards. In our corpus, we observed occasional mistakes or violations in dashboard design that were visible through inspection of the *graph structure alone*. For instance, interactive filters that were centrally placed and only applied to some (but not all) of the charts in a dashboard, a commonly placed legend although two charts used a different color mapping, etc. This suggests the ability for our graph schemas to be used to automatically "lint" or otherwise "audit" [60], [61] dashboards and surface potential issues during dashboard authoring.

Designing accessible dashboards. Recent research has highlighted the importance of making dashboards accessible to people with disabilities [62], [63], [64]. To this end, effectively understanding a dashboard's composition and design can help both assess and improve its accessibility. For instance, in a recent co-design study with screen reader users, Srinivasan et al. [64] suggested that dashboards are more accessible via screen readers if they provide explicit filtering widgets that users can easily navigate to and adjust. *The authors used our proposed graph-based schema to model the layout and interaction design of dashboards and subsequently redesign them to be more accessible.* This application exemplifies how the proposed graph schema can be used to validate if dashboards meet different accessibility criteria. Furthermore, modeling accessible dashboard design practices as graph heuristics can also help automatically update dashboard designs and make them more accessible at scale (e.g., interaction graphs should be updated such that any chart→chart connections should have an equivalent filter-->chart connection).

Enhancing dashboard search. Dashboards are challenging artifacts to search over because both their content and composition can be relevant to end-users. However, dashboard search currently prioritizes content elements, and more specifically, dashboard metadata or, when present, text that is encapsulated in the dashboard. When metadata or text content is poor, the opportunity to search over dashboards can be limited. Using a graph schema creates richer search opportunities. For example, comparing a given dashboard's graph schema to a repository of dashboard graphs can enable searching for dashboards that exhibit a similar visual layout and/or interactivity. Alternatively, it is also possible to create an embedding representation from the dashboard graphs and then perform a search over these embeddings [65]. This approach is explored with knowledge graphs of single visualizations in KG4VIS [13], but can be extended to more general graphs that capture additional aspects of dashboard content and composition.

VII. Discussion and Conclusion

Collections of user-created artifacts are an important resource for understanding existing user practices and motivating research agendas and trajectories. Here, we consider a corpus of dashboards, a complex artifact that ties together different content elements, including inter-mixing visualizations with other media, composed in both interactive and static configurations. Aspects of dashboards have been examined individually, most notably the visualization elements [9], [13], and together to reveal composition patterns [8], [7], [10]. Although these attempts use a variety of approaches to capture different aspects of dashboard content and composition, no one approach fully covers the others, and none have been tested on a large corpus of dashboards.

Our research advances prior work by proposing a block and connections representation of dashboards. Our approach brings together and extends prior research through a simple graph schema. Blocks define content, while connections define their relationships. We show that this approach can be used to capture and examine dashboard design patterns *en masse* to derive a census of these user artifacts.

A. From a "Zoo" to a "Census" of Dashboards

Prior research that closely examines dashboards primarily uses 'close reading' approaches to annotate dashboard features and derive design patterns. However, this approach can only reasonably analyze a limited number of dashboards due to their intensive manual labor demands. As with Heer et al.'s [11] "visualization zoo", these hand-picked and hand-analyzed corpora may not reflect common practice or dashboards of everyday use: "After all, you don't go to the zoo to see chihuahuas and raccoons; you go to admire the majestic polar bear, the graceful zebra, and the terrifying Sumatran tiger." A research agenda emphasizing charismatic fauna while overlooking the more prevalent raccoon has limitations.

In our census, we show that dashboard quality varies considerably, with many being far simpler than prior research accounts for. While we cannot identify a singular mechanism behind this variability or this bias toward simplicity, potential rationales suggest missing areas in our current thinking and understanding of dashboard design and use. For instance, existing authoring tools may not offer sufficient support for the majority of dashboard authors to create richer dashboards. New authoring paradigms or tools could address this mismatch. Or alternatively, the analytical needs and data literacy of dashboard audiences may be fully met by simple and static collections of one or two simple charts; if so, then research that focuses on more "exotic" forms of visual presentation and interactivity may fail to meet users where they are and assist them with their everyday analytical goals [66].

With the growth of data-intensive AI/ML applications, our census also offers an opportunity for reflection. Prior uses of visualization and dashboard corpora do not comment on what the corpora contain or how it could impact the recommendations or guidance of AI/ML models. While at face value, large corpora may invite complex analyses, the distribution of patterns surfaced by a census can inform if the analyses are feasible and appropriate. The necessary data may not exist, or, for example, if analyzing text content, it may be too sparse and simple to be usable. Developing a "zoo" of exotic examples again risks misalignment with user needs. We present initial evidence for how a census could clarify appropriate AI/ML usage of dashboard corpora. Further, our graph schema could be used to generate data cards to inform how these corpora should be used by downstream models.

B. Limitations

The primary limitation of our research is that we only use the Tableau Public corpus. However, prior research has also made use of corpora obtained from a single source and defined using a common specification to simply their analytic workloads without loss of generality. For example, VizML [9] uses Plotly charts without consideration of visualizations made with other libraries. DMiner [10] considers a set of approximately 850 Tableau dashboards mined from Github. The majority of the corpus analyzed by Bach *et. al.* [8] is also reported on in Sarikaya *et. al.* [7]. While corpora are generated and defined differently, and so may highlight different aspects of dashboard design, we hypothesize that our finding of dashboard design patterns skewing toward simplicity will likely hold. A final limitation is that we analyze relationships between dashboard elements via their interactive relationships and positional placements. There are potentially other ways of defining relationships between these elements, such as shared data or semantic information that we do not explore. What we analyze here is the minimum when considering multiple elements. For example, a representative text image may not share data with a visual encoding (see Figure 1 for an example), but we are able to model their adjacency or overlaps. Future work may seek to go further and explore other edge types. We make our corpora publicly available so that others can build on our findings or compare them to prior research in ways we do not cover or anticipate.

Acknowledgements

This research was supported in part by the National Science Foundation (Award # 2141506 & 2402718).

REFERENCES

[1] S. Wexler, J. Shaffer, and A. Cotgreave, *The Big Book of Dashboards: visualizing your data using real-world business scenarios*. John Wiley & Sons, 2017.

[2] W. W. Eckerson, *Performance dashboards: measuring, monitoring, and managing your business*. John Wiley & Sons, 2010.

[3] B. A. Wilbanks and P. A. Langford, "A review of dashboards for data analytics in nursing," *CIN: Computers, Informatics, Nursing*, vol. 32, no. 11, pp. 545–549, 2014.

[4] L. D. Roberts, J. A. Howell, and K. Seaman, "Give me a customizable dashboard: Personalized learning analytics dashboards in higher education," *Technology, Knowledge and Learning*, vol. 22, no. 3, pp. 317–333, 2017.

[5] O. M. Yigitbasioglu and O. Velcu, "A review of dashboards in performance management: Implications for design and research," *International Journal of Accounting Information Systems*, vol. 13, no. 1, pp. 41–59, 2012.

[6] S. Few, *Information Dashboard Design: The Effective Visual Communication of Data*. O'Reilly Sebastopol, CA, 2006, vol. 2.

[7] A. Sarikaya, M. Correll, L. Bartram, M. Tory, and D. Fisher, "What Do We Talk About When We Talk About Dashboards?" *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 682–692, Jan. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8443395/

[8] B. Bach, E. Freeman, A. Abdul-Rahman, C. Turkay, S. Khan, Y. Fan, and M. Chen, "Dashboard design patterns," *IEEE Transactions on Visualization &; Computer Graphics*, vol. 29, no. 01, pp. 342–352, jan 2023.

[9] K. Hu, M. A. Bakker, S. Li, T. Kraska, and C. Hidalgo, "Vizml: A machine learning approach to visualization recommendation," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1–12. [Online]. Available: https://doi.org/10.1145/3290605.3300358

[10] Y. Lin, H. Li, A. Wu, Y. Wang, and H. Qu, "Dminer: Dashboard design mining and recommendation," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–15, 2023.

[11] J. Heer, M. Bostock, and V. Ogievetsky, "A tour through the visualization zoo: A survey of powerful visualization techniques, from the obvious to the obscure," *Queue*, vol. 8, no. 5, p. 20–30, may 2010. [Online]. Available: https://doi.org/10.1145/1794514.1805128

[12] F. Moretti, "Conjectures on world literature," *New Left Review*, vol. 1, p. 54, Jan 01 2000, last updated - 2013-02-24. [Online]. Available: http://login.ezproxy.lib.vt.edu/login?url=https://www.proquest.com/scholarly-journals/conjectures-on-world-literature/docview/1301903612/se-2

[13] H. Li, Y. Wang, S. Zhang, Y. Song, and H. Qu, "Kg4vis: A knowledge graph-based approach for visualization recommendation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 1, pp. 195–205, 2022.

[14] P. Dourish, *The Stuff of Bits: An Essay on the Materialities of Information*. MIT Press, 2017, ch. Spreadsheets and Spreadsheet Events in Organizational Life, pp. 81–104.

[15] M. Tory, L. Bartram, B. Fiore-Gartland, and A. Crisan, "Finding their data voice: Practices and challenges of dashboard users," *IEEE Computer Graphics and Applications*, 2021.

[16] H. M. Al-maneea and J. C. Roberts, "Towards Quantifying Multiple View Layouts in Visualisation as Seen from Research Publications," in *2019 IEEE Visualization Conference (VIS)*. Vancouver, BC, Canada: IEEE, Oct. 2019, pp. 121–121. [Online]. Available: https://ieeexplore.ieee.org/document/8933655/

[17] X. Chen, W. Zeng, Y. Lin, H. M. Ai-Maneea, J. Roberts, and R. Chang, "Composition and configuration patterns in multiple-view visualizations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1514–1524, 2020.

[18] J. C. Roberts, "State of the art: Coordinated & multiple views in exploratory visualization," in *Fifth international conference on coordinated and multiple views in exploratory visualization (CMV 2007)*. IEEE, 2007, pp. 61–71.

[19] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky, "Guidelines for using multiple views in information visualization," in *Proceedings of the Working Conference on Advanced Visual Interfaces*, ser. AVI '00. New York, NY, USA: Association for Computing Machinery, 2000, p. 110–119. [Online]. Available: https://doi.org/10.1145/345513.345271

[20] Z. Qu and J. Hullman, "Keeping multiple views consistent: Constraints, validations, and exceptions in visualization authoring," *IEEE transactions on visualization and computer graphics*, vol. 24, no. 1, pp. 468–477, 2017.

[21] V. Setlur, M. Correll, A. Satyanarayan, and M. Tory, "Heuristics for supporting cooperative dashboard design," *IEEE Transactions on Visualization and Computer Graphics*, 2023.

[22] Y. S. Kristiansen, L. Garrison, and S. Bruckner, "Semantic Snapping for Guided Multi-View Visualization Design," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 1, pp. 43–53, Jan. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9555491/

[23] R. Langner, U. Kister, and R. Dachselt, "Multiple Coordinated Views at Large Displays for Multiple Users: Empirical Findings on User Behavior, Movements, and Distances," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 608–618, Jan. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8440846/

[24] R. Ma, H. Mei, H. Guan, W. Huang, F. Zhang, C. Xin, W. Dai, X. Wen, and W. Chen, "Ladv: Deep learning assisted authoring of dashboard visualizations from images and sketches," *IEEE transactions on visualization and computer graphics*, vol. 27, no. 9, pp. 3717–3732, 2020.

[25] P. O'Donovan, A. Agarwala, and A. Hertzmann, "Learning Layouts for Single-Page Graphic Designs," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 8, pp. 1200–1213, Aug. 2014. [Online]. Available: https://ieeexplore.ieee.org/document/6777138/

[26] M. Oppermann, R. Kincaid, and T. Munzner, "Vizcommender: Computing text-based similarity in visualization repositories for content-based recommendations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 495–505, 2021.

[27] A. Wu, L. Xie, B. Lee, Y. Wang, W. Cui, and H. Qu, "Learning to Automate Chart Layout Configurations Using Crowdsourced Paired Comparison," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, ser. CHI '21. New York, NY, USA: Association for Computing Machinery, May 2021, pp. 1–13. [Online]. Available: https://doi.org/10.1145/3411764.3445179

[28] F. B. Viegas, M. Wattenberg, F. Van Ham, J. Kriss, and M. McKeon, "Manyeyes: a site for visualization at internet scale," *IEEE transactions on visualization and computer graphics*, vol. 13, no. 6, pp. 1121–1128, 2007.

[29] L. Battle, P. Duan, Z. Miranda, D. Mukusheva, R. Chang, and M. Stonebraker, "Beagle: Automated extraction and interpretation of visualizations from the web," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, ser. CHI '18. New York, NY, USA: ACM, 2018, pp. 594:1–594:8. [Online]. Available: http://doi.acm.org/10.1145/3173574.3174168

[30] K. Morton, M. Balazinska, D. Grossman, R. Kosara, and J. Mackinlay, "Public data and visualizations: How are many eyes and tableau public used for collaborative analytics?" *SIGMOD Rec.*, vol. 43, no. 2, p. 17–22, dec 2014. [Online]. Available: https://doi.org/10.1145/2694413.2694417

[31] K. Hu, S. N. S. Gaikwad, M. Hulsebos, M. A. Bakker, E. Zgraggen, C. Hidalgo, T. Kraska, G. Li, A. Satyanarayan, and c. Demiralp, "Viznet: Towards a large-scale visualization learning and benchmarking repository," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1–12. [Online]. Available: https://doi.org/10.1145/3290605.3300892

[32] A. Jena, U. Engelke, T. Dwyer, V. Raiamanickam, and C. Paris, "Uncertainty visualisation: An interactive visual survey," in *2020 IEEE Pacific Visualization Symposium (PacificVis)*. IEEE, 2020, pp. 201–205.

[33] H.-J. Schulz, "Treevis.net: A tree visualization reference," *IEEE Computer Graphics and Applications*, vol. 31, no. 6, pp. 11–15, 2011.

[34] C. Tominski and W. Aigner, "The timeviz browser. a visual survey of visualization techniques for time-oriented data," 2017.

[35] J. Chen, M. Ling, R. Li, P. Isenberg, T. Isenberg, M. Sedlmair, T. Möller, R. S. Laramee, H.-W. Shen, K. Wünsche, and Q. Wang, "Vis30k: A collection of figures and tables from ieee visualization conference publications," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 9, pp. 3826–3833, 2021.

[36] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. D. Stolper, M. Sedlmair, J. Chen, T. Möller, and J. Stasko, "Vispubdata.org: A metadata collection about ieee visualization (vis) publications," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 9, pp. 2199–2206, 2017.

[37] P.-S. Lee, J. D. West, and B. Howe, "Viziometrics: Analyzing visual information in the scientific literature," *IEEE Transactions on Big Data*, vol. 4, no. 1, pp. 117–129, 2018.

[38] E. Hoque and M. Agrawala, "Searching the visual style and structure of d3 visualizations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 1236–1245, 2020.

[39] K. Davila, S. Setlur, D. Doermann, B. U. Kota, and V. Govindaraju, "Chart mining: A survey of methods for automated chart analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 3799–3819, 2021.

[40] J. Poco and J. Heer, "Reverse-engineering visualizations: Recovering visual encodings from chart images," *Computer Graphics Forum*, vol. 36, no. 3, pp. 353–363, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13193

[41] J. Poco, A. Mayhua, and J. Heer, "Extracting and retargeting color mappings from bitmap images of visualizations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 637–646, 2018.

[42] V. S. N. Prasad, B. Siddiquie, J. Golbeck, and L. S. Davis, "Classifying computer generated charts," in *2007 International Workshop on Content-Based Multimedia Indexing*, 2007, pp. 85–92.

[43] M. Savva, N. Kong, A. Chhajta, L. Fei-Fei, M. Agrawala, and J. Heer, "Revision: Automated classification, analysis and redesign of chart images," in *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 393–402. [Online]. Available: https://doi.org/10.1145/2047196.2047247

[44] Z. Zeng, P. Moh, F. Du, J. Hoffswell, T. Y. Lee, S. Malik, E. Koh, and L. Battle, "An Evaluation-Focused Framework for Visualization Recommendation Algorithms," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 1, pp. 346–356, Jan. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9552925/

[45] D. Moritz, C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe, and J. Heer, "Formalizing Visualization Design Knowledge as Constraints: Actionable and Extensible Models in Draco," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 438–448, Jan. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8440847/

[46] A. Crisan, S. E. Fisher, J. L. Gardy, and T. Munzner, "GEViTRec: Data Reconnaissance Through Recommendation Using a Domain-Specific Visualization Prevalence Design Space," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9524484/

[47] P. Eichmann, E. Zgraggen, C. Binnig, and T. Kraska, "Idebench: A benchmark for interactive data exploration," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 1555–1569. [Online]. Available: https://doi.org/10.1145/3318464.3380574

[48] J. Purich, H. Mahmood, D. Chou, C. Udeze, and L. Battle, "An adaptive benchmark for modeling user exploration of large datasets," 2022.

[49] Y. S. Kristiansen, L. Garrison, and S. Bruckner, "Content-driven layout for visualization design," in *Proc VINCI'22*, ser. VINCI '22. New York, NY, USA: Association for Computing Machinery, 2022. [Online]. Available: https://doi.org/10.1145/3554944.3554950

[50] S. K. Badam, Z. Liu, and N. Elmqvist, "Elastic documents: Coupling text and tables through contextual visualizations for enhanced document reading," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 1, pp. 661–671, 2018.

[51] W. Tang, Z. Lu, and I. S. Dhillon, "Clustering with multiple graphs," in *2009 Ninth IEEE International Conference on Data Mining*, 2009, pp. 1016–1021.

[52] W. Huang, R. Liu, and C.-L. Tan, "Extraction of vectorized graphical information from scientific chart images," in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, vol. 1, 2007, pp. 521–525.

[53] R. J. G. B. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Advances in Knowledge Discovery and Data Mining*, J. Pei, V. S. Tseng, L. Cao, H. Motoda, and G. Xu, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 160–172.

[54] V. Dhanoa, C. Walchshofer, A. Hinterreiter, H. Stitz, E. Gröller, and M. Streit, "A Process Model for Dashboard Onboarding," Open Science Framework, preprint, Dec. 2021. [Online]. Available: https://osf.io/gux9w

[55] S. Haroz, R. Kosara, and S. L. Franconeri, "Isotype visualization: Working memory, performance, and engagement with pictographs," in *Proceedings of the 33rd annual ACM conference on human factors in computing systems*, 2015, pp. 1191–1200.

[56] A. Pandey, A. Srinivasan, and V. Setlur, "Medley: Intent-based recommendations to support dashboard composition," *arXiv preprint arXiv:2208.03175*, 2022.

[57] D. J.-L. Lee, V. Setlur, M. Tory, K. G. Karahalios, and A. Parameswaran, "Deconstructing Categorization in Visualization Recommendation: A Taxonomy and Comparative Study," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9444894/

[58] S. Latif, Z. Zhou, Y. Kim, F. Beck, and N. W. Kim, "Kori: Interactive synthesis of text and charts in data documents," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 1, pp. 184–194, 2021.

[59] Q. Zhi, A. Ottley, and R. Metoyer, "Linking and layout: Exploring the integration of text and visualization in storytelling," vol. 38, no. 3, pp. 675–685, 2019.

[60] Q. Chen, F. Sun, X. Xu, Z. Chen, J. Wang, and N. Cao, "Vizlinter: A linter and fixer framework for data visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 1, pp. 206–216, 2022.

[61] A. McNutt, G. Kindlmann, and M. Correll, "Surfacing Visualization Mirages," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, Apr. 2020, pp. 1–16. [Online]. Available: https://doi.org/10.1145/3313831.3376420

[62] N. W. Kim, S. C. Joyner, A. Riegelhuth, and Y. Kim, "Accessible visualization: Design space, opportunities, and challenges," in *Computer Graphics Forum*, vol. 40, no. 3. Wiley Online Library, 2021, pp. 173–188.

[63] A. F. Siu, D. Fan, G. S. Kim, H. V. Rao, X. Vazquez, S. O'Modhrain, and S. Follmer, "Covid-19 highlights the issues facing blind and visually impaired people in accessing data on the web," in *Proceedings of the 18th International Web for All Conference*, 2021, pp. 1–15.

[64] A. Srinivasan, T. Harshbarger, D. Hilliker, and J. Mankoff, "Azimuth: Designing accessible dashboards for screen reader users," in *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility*, 2023, pp. 1–16.

[65] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.

[66] L. Battle, D. Feng, and K. Webber, "Exploring d3 implementation challenges on stack overflow," in *2022 IEEE Visualization and Visual Analytics (VIS)*. IEEE, 2022, pp. 1–5.

Arjun Srinivasan is a member of the Research Staff at Tableau and holds a Ph.D. in Computer Science from the Georgia Institute of Technology. His research explores the design of intelligent and expressive visualization tools that combine multimodal input and mixed-initiative interface techniques for human-data interaction.

Joanna Purich is a PhD student at the University of Maryland.

Michael Correll is a Research Associate Professor at Northeastern University's Khoury College of Computer Sciences and a member of the Roux Institute. His research explores the visual communication of statistics to mass audiences, data ethics, and visualization rhetoric. He holds a PhD in Computer Sciences from the University of Wisconsin-Madison

Leilani Battle is an Associate Professor at the Paul G. Allen School for Computer Science & Engineering at the University of Washington. Her research focuses on developing interactive systems for exploring large, complex datasets. She earned a PhD degree in Computer Science from MIT.

Vidya Setlur is the Senior Director of Tableau Research. Her research lies at the intersection of natural language processing and computer graphics to better understand data semantics and user intent to inform the meaningful visual depiction of data. She earned a PhD degree in Computer Science from Northwestern University.

Anamaria Crisan is a Lead Research Scientist at Tableau. She holds a Ph.D. in Computer Science from the University of British Columbia. Her research explores the intersection of data science and data visualization; her earlier research has emphasized biomedical applications.