Graphicle: Exploring Units, Networks, and Context in a Blended Visualization Approach

Timothy Major, Student Member, IEEE, and Rahul C. Basole, Senior Member, IEEE

Abstract—Many real-world datasets are large, multivariate, and relational in nature and relevant associated decisions frequently require a simultaneous consideration of both attributes and connections. Existing visualization systems and approaches, however, often make an explicit trade-off between either affording rich exploration of individual data units and their attributes or exploration of the underlying network structure. In doing so, important analysis opportunities and insights are potentially missed. In this study, we aim to address this gap by (1) considering visualizations and interaction techniques that blend the spectrum between unit and network visualizations, (2) discussing the nature of different forms of contexts and the challenges in implementing them, and (3) demonstrating the value of our approach for visual exploration of multivariate, relational data for a real-world use case. Specifically, we demonstrate through a system called Graphicle how network structure can be layered on top of unit visualization techniques to create new opportunities for visual exploration of physician characteristics and referral data. We report on the design, implementation, and evaluation of the system and effectiveness of our blended approach.

Index Terms—Unit visualization, network visualization, context

1 INTRODUCTION

As the availability and complexity of data increases across many domains, the use of data visualization and visual analytics continues to gain in prominence [8, 22]. Many real-world datasets are often large, multivariate, and relational in nature and relevant associated decisions frequently require the consideration of both attributes and network structure. Consider data about physicians. There are more than 950,000 actively licensed physicians in the United States [45]. Each physician has a plethora of attributes, including medical specialty, practice location, hospital affiliations, education and training background, and many more. Relations between physicians can be explicit (e.g., referrals) or implicit (e.g., co-investigators on a grant, same residency). For such a dataset, both the network structure and data attributes can be important in the exploration process. An analyst looking for key influencers within this physician landscape, for example, would not only be interested in knowing who has a high number of publications on a specific topic (data attribute) or how many physicians they are connected to (structure), but also if their influence is limited to a few organizations or spans multiple organizations and states of interest (structure and...
data attributes). To answer these types of questions, analyses may commence from different starting points. An analyst may begin her investigation from an overview of the entire physician network first or start with what she knows — a specific physician — and then explore the data to identify relevant patterns.

Supporting such flexible analyses in an intuitive manner is challenging. It requires allowing both top-down [27] and bottom-up [35] data exploration approaches and the ability to inspect data attributes in the context of both network and faceted structure [9, 41, 46]. One existing approach to such problems is unit visualization. Unit visualizations encode each data case as a visual mark, maintaining their identity and allowing flexible multi-focus and multi-scale approaches [23]. Existing studies, however, have yet to investigate how to overlay network data into their approaches. In this study we aim to fill this important gap by characterizing the design space between unit visualization types, including direct, packed, and network unit visualizations, and showing how it can be defined as a spectrum between unit-based (i.e., data attributes and values) and structure-based (i.e., network connections) exploration. Moreover, when blending the two techniques, the notion of context needs to be revisited. In our study, we define the different forms of context found in unit visualizations and consider how they can be used as paths for exploration.

We instantiate these ideas in a visualization system that blends techniques from across the unit visualization spectrum to improve the exploration of large multivariate networks. Our core design is based on layering network structure onto several unit visualization approaches. By so doing, we soften the boundaries between unit-based and structure-based views and simplify the ways in which users can interact with the data. The system supports several layouts that can be used to position units on a canvas while providing clean, simple, and seamless transitions between them. This includes novel layouts that allow users to explore connections within and without groups of data at multiple scales. Direct manipulation can be used to select, filter, and restore subsets of interest based on data attributes or network structure. The use of dedicated space for encoding removed units allows users to explore filtered context and intelligently restore it to the canvas.

In summary, our main contributions are:

- A characterization of the design space between unit visualizations that facilitates a better understanding of the relationship between unit-based and structure-based exploration, embodied in a system that blends unit visualization techniques to allow flexible large-scale multivariate network exploration through layouts that support multi-focus and multi-scale inspection of data attributes and network structure simultaneously combined with interaction techniques that allow users to select context and restore filtered units intelligently.

The remainder of this study is organized as follows. Section 2 presents related work. The design space between unit visualization approaches is characterized in Section 3. Section 4 describes our visualization system, Graphicle. An illustrative real-world usage scenario is provided in Section 5. Section 6 describes the results of an expert study with pharmaceutical professionals. Section 7 discusses our findings. Section 8 concludes the study with implications and opportunities for future work.

2 RELATED WORK

Despite commonalities, the fields of unit visualization and network visualization have largely been addressed independently from one another. In this section we review prior work in each of these fields and discuss how the strengths of each can be combined for exploration of multivariate, relational data.

2.1 Unit Visualization

Park et al. [23] define unit visualization as a visualization that maintains the identity property of each visual mark. In other words, each data case is represented in the visualization by a mark. This definition encompasses many well-known visualization techniques including scatterplots and dotplots [42]. In contrast, aggregate visualizations seek to merge data cases together into a single visual encoding [13]. Some visualizations exist that blur the distinction between the two approaches, however. Hieraxes [29] aggregate units within a grid to allow users to explore and organize large data collections. This sort of faceted unit exploration is also supported by techniques and tools such as GatherPlots [24] and SandDance [11]. Pixel bar charts [21] use the individual pixels within the bars of a bar chart to capture both aggregate and detailed information about multivariate data. Circle packing [40] uses nested circles to depict both hierarchy and individual nodes within a tree. The advantages of these methods is that information about both data cases and data distributions are visualized simultaneously.

In their work, Park et al. [23] seek to classify the unit visualization family. They propose a grammar for describing and generating unit visualizations. However, they do not address the subject of network visualization despite node-link diagrams falling under their definition of unit visualization. This may be due to the fact that their grammar is focused on producing layouts where unit position is determined by data attributes and values, rather than network topology as is common with unit-based approaches. In general, work in unit visualization has remained separate from network visualization. Two notable exceptions are GraphDice [4], which overlays edges on top of a scatterplot and plot matrix, and the FlowVizMenu [38] which extends the work through more complex plot matrix layouts and a contextual menu for dynamic selection of data. These methods, however, only blend a simple case of unit visualization with network structure when there are many other approaches that can be considered.

2.2 Network Visualization

Extensive work has been done in the field of network visualization (see e.g., [18, 39]). A substantial portion of it is devoted to topological structure and layout [3]. One of the common ways network visualizations are modified for better understanding is by grouping nodes based on topology or attributes [37]. Methods exist that treat groups as nodes themselves. GrouseFlocks [2], for example, creates metanodes based on hierarchy in order to simplify the network representation. The motif simplification technique [12] also simplifies the visual representation of complex graphs by replacing graph motifs with representative glyphs. In both approaches, edges become combined into thicker metaedges. Our system utilizes the concepts of aggregate nodes and edges, but they are based on data attributes rather than topology. In addition, rather than replacing multiple nodes with a single metanode glyph, all nodes remain visible but can be interacted with as a whole.

Another significant research area in network visualization is focused on integrating multivariate data properties into the exploration process. Several techniques have been presented. Pretorius and Van Wijk [25] demonstrate an interface that allows users to partition or aggregate the graph according to the attributes of nodes or edges. PivotGraph [41] provides an aggregate view on the network by rolling up nodes and edges and positioning them on a grid. Unit-based structural exploration is lost in both these methods, however. Other approaches, such as the Group-in-a-Box layout [26] divide networks into different regions based on data characteristics without aggregation. Semantic substrates [28] separate nodes into non-overlapping regions based on a categorical attribute. Nodes within a region can be positioned either by that attribute or using force-directed methods. PivotSlice [46] builds on this method by allowing users to subdivide the available space into regions using a visual query language. Our system allows faceting by attributes and intra-group layout options, though our layout options are more flexible and support organization based on other unit attributes.

As outlined above, current methods are still focused primarily on either structural exploration or multivariate exploration, but not both. The DOSA process proposed by Van Den Elzen and Van Wijk [34] is a notable exception by using selections of interest to simultaneously show details and high-level overviews of large multivariate networks. This approach, however, has several drawbacks. First, it requires two views, decreasing the available rendering space for details and requir-
ing users to map the relationship between the two. Second, breaking up the data into subparts requires direct manipulation for each selection of interest. This can be tedious. Third, like many methods for multivariate network exploration, DOSA does not support bottom-up paths. g-Miner [5] supports interactive group mining using structural and multivariate exploration. Users can identify groups using top-down, bottom-up, and middle-out [44] approaches by searching via hierarchy, ego-networks, and template matching respectively. However, their approach is designed to support a specific user task rather than general visual analysis.

3 Characterizing the Design Space

This study aims to fill the aforementioned gaps by exploring the design space between unit visualizations and network visualizations. We begin by classifying node-link diagrams as unit visualizations according to the definition proposed by Park et al. [23]. Doing so allows us to characterize the commonalities and distinctions between the two groups in a more nuanced manner. The design space includes three types of unit visualizations:

- **Direct unit visualizations.** These approaches position each unit according to data attributes or values. Scatterplots are a common form of direct unit visualization.

- **Packed unit visualizations.** These approaches blur the lines between unit and aggregate visualizations. Units are packed within containers [23] to show information about both data cases and distributions. Each container represents a facet based on values for data attributes. Dotplots [42], pixel charts [21], or Gather-Plots [24] are examples of packed unit visualizations.

- **Network unit visualizations.** These approaches emphasize the structure of a graph. Units are positioned according to the connections they have with other nodes. Node-link diagrams are a common form of network unit visualization.

In order to support rendering all units simultaneously, each type of unit visualization requires as much rendering space as possible. Given the growing size and complexity of datasets, it is therefore particularly pertinent to address issues of scaling and occlusion.

3.1 Multi-focus, Multi-scale, and Multi-path Exploration

One of the strengths of graphically depicting each unit is that visual exploration can occur across multiple parts of the dataset (multi-focus exploration) and at multiple levels (multi-scale exploration). This is prominent with packed and network unit visualizations. In packed approaches the positioning of a unit within a larger container reveals aggregated counts or distributions while retaining the ability to find within-container patterns and unit outliers. Similarly, network visualizations often employ layout algorithms chosen to group related nodes together. This allows users to see clusters and other features of the graph at a topographical level, but individual nodes and their connections are still available to be explored.

Unit visualization methods also facilitate multiple paths for exploration. A top-down approach begins with an overview of the entire dataset. Interaction is then used to constrain the data to specific subsets of interest. Following the visual information seeking mantra “overview first, zoom and filter, then details-on-demand” [27] supports user tasks that seek to identify interesting patterns or units within the overall dataset. A bottom-up [35] approach starts with a unit of interest and expands the area of exploration outward by navigating structure (in the case of networks) or through a degree-of-interest function. This allows users to begin exploration with previous knowledge or ask questions about a specific datum.

3.2 Defining Context

Context plays an important role in the visual exploration process. Generally, context is information that helps orient a user within the available data, showing what other parts of the data exist as well as aiding with the interpretation of the current focus [6, 14]. Context is found in many forms within unit visualizations. Common across existing approaches, filtered context can be defined as those units that are being hidden or removed due to a selection criteria. In addition to dealing with filtered context, packed unit visualizations place units within the context of their containers. Thus, the multi-scale exploration afforded by packed visualizations is really an affordance of context. We will refer to this context as container context. Likewise, network unit visualizations have another form of context in addition to filtered context. Structural context refers to the nodes and edges that are directly or indirectly linked to the current focus. It can include both units on the screen as well as those that have been hidden or removed. In the latter case, we refer to it as filtered structural context.

3.3 The Continuum Between Direct, Packed, and Network

The three types of unit visualizations show two separating characteristics. The first characteristic is the innate support for multi-focus and multi-scale exploration. Packed and network unit visualizations have this feature, while direct unit visualizations do not. The second separating characteristic is the use of topography/structure within the visualization. Network visualizations use topography as the source of meaning for unit position as well as a source of context, while direct and packed unit visualizations do not. Dividing the visualization types along the topographical/non-topographical line also highlights a difference between the two groups on matters of exploration. For network visualizations, the focus is often on exploring and navigating the structure rather than data attributes and values. Direct and packed unit visualizations, on the other hand, rarely integrate network structure for exploration, though some hybrid methods do exist [4].

Considering these distinctions, we can place all three types along a continuum from unit driven, single-focus, single-scale exploration to structure-driven, multi-focus, multi-scale exploration, as shown in Fig. 2. Transitioning from direct unit visualization to packed unit visualization is straightforward: the addition of container context also creates opportunities for multi-focus and multi-scale exploration. Movement from packed to network, however, is more complex. Some questions to ask about this transition are:

- Q1. What are the relationships between unit-driven and structure-driven exploration?
- Q2. How can structural characteristics be integrated into packed (and direct) unit visualizations?
- Q3. Conversely, how can unit-focused characteristics be integrated into network unit visualizations?
- Q4. Are there multiple degrees to which structural and unit-focused characteristics can be combined? Or, in other words, is there a spectrum of design choices that can be made?
- Q5. What roles does context play within blended unit/structural visualization?
We answer these questions by examining how functionality from direct, packed, and network unit visualizations can be layered on top of each other. In particular, our approach is to combine network features with packed and direct unit visualizations. Features supporting unit-driven exploration serve as the foundational layer. Structural information (unit network connections) is added as the second. Third, intra-container network structure is exposed. Fourth, inter-container network features are added. We will elaborate by describing how this blended approach has been implemented in our visualization system.

4 The Graphicle System

We have designed and implemented Graphicle, a web-based application that allows users to explore large multivariate networks using approaches that span the continuum from direct to packed to network unit visualization. The application is written in JavaScript and uses the HTML5 Canvas API to support visualizing tens of thousands of nodes and edges. To illustrate the features of our system, we use a physician referral dataset as a running example. Edges are undirected, though directed edges are supported.

4.1 Unit Exploration

The basic layer of Graphicle is the unit canvas, shown in Fig. 1a. Individual units are encoded by circles, also referred to as particles. At startup, all units are drawn to the canvas in order to allow users to begin with an overview of the dataset. From there, they can begin top-down exploration or use the searchable (Fig. 1d) to find a specific unit from which to start. The filter panel (Fig. 1b) contains selected widgets [43] that are cross-filtered together. Histograms and filled bars display the distributions of values for the corresponding data attribute and are updated as units are filtered or restored. Zooming and panning is supported through the mouse rather than fixing the viewport. This helps the system better support exploring very large datasets. We also believe this “particles-on-a-canvas” metaphor helps users understand the visual representation and how units are positioned by leveraging the constructivism and physicality of unit visualization [19, 23].

Units can be colored and/or sized according to data attributes. For quantitative attributes, the system can be configured to apply interpolation to the values in order to better discriminate among the attribute domain. For example, high value outliers can be encoded with very similar sizes in order to “spread out” the other much more common values. As users filter and remove units from the canvas, coloring and sizing adjust to include only the attribute values of the data that remains. In other words, the data domain used for mapping to color and size adjusts to what is seen. This allows users to continue leveraging coloring and sizing for discrimination.

4.1.1 Unit Views

Two views are available to position nodes according to data attributes and values. The scatterplot and grid views correspond to direct and packed unit visualizations, respectively. Smooth transitions between views help users maintain awareness of how units are being repositioned [16, 30]. We will now describe each view in more detail.

Scatterplot. The scatterplot is a direct unit visualization that allows users to position units along the x- and y-axes according to data attributes. Both categorical and quantitative attributes can be selected for each axis. When a categorical attribute is selected, units are positioned at equally spaced intervals along the corresponding axis. When zooming and panning about the canvas, axes ticks and labels are dynamically updated to reflect the current transform for quantitative attributes, but remain fixed for categorical attributes. The effect maintains the “particles-on-a-canvas” metaphor.

Grid. The default view is the grid view. It is a packed unit visualization. Users can subdivide the units into different containers, referred to as clusters, by selecting a categorical attribute in the “Cluster By” menu. Units are packed into clusters in one of two different ways, shown in Fig. 3. If units are unsorted, they are packed into a circle; otherwise they are packed into a square sorted from left to right, top to bottom. Varying the shape of the packing helps users quickly identify if sorting is taking place and if x/y position within a cluster carries meaning.

4.1.2 Unit Interaction

The system supports direct interaction with units on the canvas. Brushing a unit highlights it and displays a tooltip (e.g., the physician’s name). Units can be selected by clicking on them; a card is displayed with information about the corresponding datum. Other units can be added to the current selection by use of the cmd/ctrl key or multiple units can be selected at once through “rubber band” selection. The information card updates to show the properties of the whole selection. Interacting with the label of a cluster will highlight all the nodes in that cluster. Clicking on the label will select them. This allows users to quickly select groups of interest. In addition, units can be filtered based on the current selection. The “Remove” action removes all units in the current selection, while “Isolate” retains the units in the current selection and removes all others.

The ability to directly select units on the canvas enables users to rapidly explore and filter the dataset based on characteristics of the visual representation rather than by forming queries through filter widgets (e.g., [43]) or a query language (e.g., [46]). This reinforces and leverages the “particles-on-a-canvas” metaphor. Direct selection in Graphicle also enables multi-scale filtering: both individual units and groups can be selected and removed from the canvas. This method for quick, expressive filtering is also useful for many network-related sorting can occur under two conditions. First, users are able to select an attribute by which to sort. Second, an implicit sort is created when units are colored or sized by attribute. The purpose of this is twofold. First, adding an implicit sort allows users to more easily discriminate between the attribute values used for color/size and see distributions (strata) of nodes because of their position. This approach supports the aims of unit visualization, which seeks to show both a macro and a micro view. Second, sorting by multiple attributes brings out two visual insights: anomalies within strata “pop out” and uneven distributions within strata are shown through a “speckling” effect. If multiple conditions for sorting are met, sorting occurs by the user-specified attribute first, then size, then color. In our system, the available variance in size is much smaller than the variance in color, so we prioritize sorting by size first to increase its discriminatory power. Users are informed of an implicit sort through the “Sort By” menu in the toolbar.

Fitting non-uniform circles to a square through packing has challenges. First, computing the packing can be computationally expensive for large numbers of circles. Second, existing approaches [20, 23] either place the units in a random order or build the packing diagonally. This does not support good visual ordering and scanning; the distribution of the labels in the container becomes difficult to see and interpret. To overcome these challenges, we place units into a grid spaced according to the size of the largest circle across all units in all groups (a global “shared” sizing [23]). We refer to this approach as the clustered grid layout. A small amount of overlap is allowed between circles in order to compress the clusters. In addition, the circles are slightly transparent. This creates a visual border around overlapping circles in order to reduce occlusion. Smaller circles become spaced with gaps between them, which reduces their visual weight and draws interest to the larger ones. With this approach, the position of a unit can be calculated algebraically. It is scalable, supports sorting, and supports non-uniform circles.

The clusters are placed into a grid matching the aspect ratio of the canvas (using the same layout algorithm as the clusters), ranked from most to fewest units. Typically, the layout of clusters and units in packed unit visualizations are dictated by a top-down subdivision of the available space [23]; the view is divided uniformly or proportionally into containers depending on the desired layout, then units are packed within them. With our clustered grid layout, however, the size of a unit dictates the amount of space needed for the packing. This bottom-up approach allows for zooming and panning the canvas while avoiding problems that would occur if units have a scaling effect that interacts with the canvas transform. When a new layout occurs, a smooth zoom and pan [36] adjusts the canvas to fit the new layout boundaries. After a small delay, the units then transition to their new positions.
We now describe the ways Graphicle further explores the design space. For example, when organizing units into clusters in the grid view, as units are brushed, connections to other units are shown by high-lighting edges and linked units. The network view uses a force-directed layout to position the units in a node-link diagram. Edges are interpolated using a gradient when nodes are colored by an attribute. Unit and edge transparency as well as a dark canvas background help with occlusion. Transparency makes the boundaries between overlapping elements more distinguishable and subtle variations in transparency are more noticeable with the dark background. Like the scatterplot and grid views, the network view supports direct interaction with the units as well as zooming and panning around the canvas.

4.2 Network Exploration

Graphicle facilitates network exploration through (1) a network view and (2) integrating network features into the scatterplot and grid views. The network view uses a force-directed layout to position the units in a node-link diagram. Edges are interpolated using a gradient when nodes are colored by an attribute. Unit and edge transparency as well as a dark canvas background help with occlusion. Transparency makes the boundaries between overlapping elements more distinguishable and subtle variations in transparency are more noticeable with the dark background. Like the scatterplot and grid views, the network view supports direct interaction with the units as well as zooming and panning around the canvas.

4.2.1 Network Interaction

As units are brushed, connections to other units are shown by high-lighting edges and linked units. Elements that are outside the structural context fade into the background. Users can select connected units by holding down the mouse to open a context menu with options for the target’s 1-step network (direct connections), 2-step network (direct connections and their connections), or full network (all units reachable through connections starting at the target). Fig. 4 shows examples of each. The units can then be filtered using the “Remove” and “Isolate” options previously mentioned. This provides users with a technique to filter the dataset based on topological structure.

The ability to explore network structure is also available in the scatterplot and grid views. Integrating network information into the direct and packed unit views reveals information about the design space between unit visualization types. As connections are layered across views, each view is no longer a strongly distinct “mode” of exploration. Instead, the views begin to differ only in the way that they position units. This simplifies the user’s mental model; what a unit represents and how to interact with one is consistent. Users are free to focus on positioning units in a way that best supports their current tasks without losing structural context. Certain patterns become easier to find as well. For example, when organizing units into clusters in the grid view, cross-cluster connections are easily seen. In the scatterplot, cross-strata connections are revealed.

4.3 Blended Exploration

We now describe the ways Graphicle further explores the design space between packed and network unit visualizations by considering how the addition of containers and container context in packed unit visualizations opens up opportunities for intra- and inter-cluster network structure. We blend force-directed network features with packed unit visualization methods to give users flexibility in how they position units and explore the data. These choices fall into a spectrum as shown in Fig. 5.

4.3.1 Clustered Networks

In addition to packing clusters into circles and grids, Graphicle allows clusters to be expanded into individual force-directed networks by clicking on their labels while holding the “alt” key. Examples are shown in Fig. 1 and Fig. 5 (b,d). This layout is called a clustered network—a form of superimposed graph partitioning [37] like semantic substrates [28] or group-in-the-box layouts [26]. Combining clustered networks with clustered circles and grids allows for multi-focus exploration. Clusters of interest can be expanded into networks while keeping others unchanged. Thus, both structural context and container context are available for exploration. Users are able to focus on the relationships within clusters while maintaining an organized view of the context without. Connections within the cluster can be viewed with specificity and connections without are shown in distributions. This approach scales better than a single network unit visualization, too. Network layouts are only computed on demand for a subset of the units, increasing responsiveness. In addition, because networks remain organized in clusters rather than combined in a single layout, overall readability is improved.

4.3.2 Networked Clusters

When clusters are expanded into clustered networks, only edges connecting units within the cluster are made explicitly visible. However, inter-cluster connections exist. Fig. 6a shows how inter-cluster connection information is displayed when interacting with cluster labels in the grid view. Brushing a cluster label highlights all units within the cluster and all units connected to them. In addition, an edge is shown connecting the brushed cluster to the other clusters that have linked units. Edge thickness indicates the sum total weight of the connections between them.

This approach is designed to support multi-scale exploration. Users are able to examine connections in an aggregate form. All units within the brushed cluster become highlighted to indicate that further interaction with the label will deal with the cluster as a whole. Drawing a single edge between the brushed cluster and clusters with connected units also reinforces the idea of a “cluster-as-unit”. However, because one of the benefits of a packed unit visualization is the ability to see distributions within containers, we only highlight individual nodes that are connected to the brushed cluster, not entire clusters with connected nodes. Admittedly, this means that the distribution of units with connections within the brushed cluster is not shown. Coloring, sizing, or sorting unit by degree can show this distribution.

Drawing aggregate edges extends previous work, such as Grouse-Flocks [2], NodeTrix [17], and motif simplification [12]. Graphicle further extends the idea of “cluster-as-unit” in two ways. First, long-press on a cluster label selects the network of a cluster. Second, clusters...
Fig. 4. Users can utilize long press on a node (or a cluster label (not shown)) to open up a context menu where they can select the target’s (i.e., node or cluster) (a) 1-step , (b) 2-step, or (c) full network. The selected structure is highlighted and enclosed by a blue selection frame.

Fig. 5. The spectrum between packed and network unit visualizations. The addition of intra- and inter-cluster forces creates new layout types. (a) shows the basic grid cluster layout with no forces applied. Individual clusters can be expanded into force-directed graphs, creating clustered networks as shown in (b). The networks remain in place within the overall grid layout. Applying inter-cluster force ((c) and (d)) treats clusters as supernodes and creates a force-clustered layout. Aggregate edges between the clusters are then displayed.

4.4 Context Exploration

The previous section discussed the ways in which structural and container contexts can interact. It is also important to consider how structural and filtered context overlap. In a top-down exploration approach, structural context may be removed through filtering (referred to as filtered structural context). In bottom-up exploration, structural context begins filtered and is brought in on-demand [15,35]. Graphicle supports flexible exploration by allowing users to view and selectively restore exploration as a strength of unit visualizations.
units that have been removed via two context bars. We now describe a context bar and show how it enables exploration wherein zoom and focus can be controlled intelligently by the user.

4.4.1 Context Interaction

As units are removed from or returned to the canvas through filtering, they are placed into a context bar on the right of the interface. (see Fig. 1e and Fig. 6). The units are animated as if they are flowing into or flowing out of the bar. The context bar is an aggregate visualization that displays the total number of units in the filtered context in proportion to the overall total number of units. Units are grouped into pixel “blocks”, with many units represented by a single block in order to save screen space. If there are fewer units in the context than are required to fill a block, a single block is still displayed in order to keep a visual indicator that units are in context.

Filtered structural context is highlighted as users interact with the canvas. As users brush units or cluster labels, a number of blocks in the context bar proportional to the number of filtered units connected to the brushed target are highlighted. (See Fig. 6a). In addition, selecting structural context via long press (Fig. 4) selects units in the filtered context if it exists, as shown in Fig. 6b. When units in the filtered context are selected, a card is shown that displays the distribution of data values for the selected units. Users can choose to restore the the units to the canvas through the “Bring in” action. Individual groups in the shown distribution can be selected for more fine-tuned control over what is restored, shown in Fig. 6c. Clicking on a context bar will select the entire filtered context. Selected units in the filtered context are not included in the main data card in order to keep focus on the visible units, avoid confusion, and avoid a situation where information becomes inaccessible to the user. (in this case, the user would be unable to separate information about the unfiltered units from the filtered units.) However, users can use the “Bring in” action to quickly add the filtered units back to the canvas and the current selection, which then adds their information to the data card. This action is easily reversed through undo to restore the focus to the original subset of data.

4.4.2 The Nuances of Filtered Context

The addition of a context bar allows users to quickly refine the exploration space. After zooming and filtering to a subset of interest, they can “back up” and readjust the subset. However, restoring structural context to the canvas interacts with filtering widgets in an interesting way. When selected units are brought back, they may still be filtered out by the current filters. To overcome this issue, the filters are modified to allow all the desired units to be restored to the canvas. If the current domain of a filter needs to be changed to include the data values in the restored units, the filter is modified and changes are reflected in the corresponding scented widget.

Modifying the filters creates a second issue. Expanding filter domains may also restore unwanted units to the canvas. In response, a second step is added to the “Bring in” action: units that would — but should not — be restored as part of the filter modifications are removed and placed into a second context bar to the left of the first. Thus, the second context bar contains units that are removed by interaction outside the filter widgets. When filtered structural context is selected, it is selected within both context bars, if applicable, and presented in a single distribution card.

5 USAGE SCENARIO

Consider the following actual real-world usage scenario for Graphicle. A major biopharmaceutical company is interested in gaining a comprehensive understanding of the neurologist landscape in the United States and identifying thought leaders and their network of influence. Existing analysis approaches involve querying a customer relationship management database with relevant search criteria (e.g., physicians field of specialty, geographic location, hospital affiliation, etc.) and receiving a list-based spreadsheet of physicians. Pending the specificity of the search query, the resulting list may not capture relevant physicians, result in an empty set with no feedback of why, and typically does not include any relational information to identify connected physicians.

To address this problem, and in collaboration with a global medical team from a biopharmaceutical company, we curated a comprehensive dataset of neurologists in the U.S. using multiple publicly available data sources (e.g., NPI Registry [31], Doximity [10], PubMed [32], American Board of Psychiatry and Neurology [1], US News Hospital Rankings [33], and Medicare Referrals [7]). Our final dataset consists of comprehensive demographic, educational, publication, and organizational affiliation data attributes of 23,207 neurologists and 44,256 referrals among them. Following expert feedback, we also computed additional metrics from existing data attributes (e.g., a neurologists influence score) and referral network-centric attributes (e.g., direct connections, full network, etc.). All of these attributes are provided as filters and options for clustering and visual encoding.

Suppose that Walter, a pharmaceutical sales representative, is tasked...
to find physician leads for a new drug that is being launched. He begins his inquiry by focusing on his assigned geographic region (Minnesota, Maryland, Massachusetts, Ohio, and Pennsylvania) and non-pediatric neurologists, reducing the total number of physicians to 4,116. Next he clusters the units by state and he observes that both Pennsylvania (1,079) and Massachusetts (1,027) have roughly the same number of neurologists and Minnesota the least (498). Since he is interested in physicians that have a high impact ranking, he colors the units by “Impact Ranking” using a single-hue scale. He notices that MA has proportionally more high-impact ranking neurologists. Rubberbanding over the corresponding rows it reveals that the majority of neurologists come from Massachusetts General Hospital and Brigham and Women’s Hospital, confirming his knowledge of the MA ecosystem. Yet, he also knows that these hospitals are not open to collaboration.

Realizing that MA-based physicians may not be a good starting point, Walter continues his exploration by focusing on physicians who are highly referred to using the “Indegree” slider (>5) and then applying the force-clustering layout to identify cross-state referrals. He immediately notices that, within this filtered dataset, MA- and PA-based physicians have referral connections to many states in his geographic region, but that there are fewer referrals for MN and OH-based neurologists. Moreover, the MA-based neurologists are highly ranked when clustering by state, whereas the PA-based neurologists are not.

He colors the nodes by “Collaboration Advocacy” and notices that MN does not have any physicians aligned for partnership, but OH has several. Wanting to understand the referral network structure within OH, he activates the cluster network view and sizes the nodes by “Impact Ranking”. He observes several connected network clusters and identifies one physician (Dr. Bell) who is aligned for partnership and relatively well-connected to many unknown physicians. He selects Dr. Bell’s subgraph and finds that there are 74 physicians in his influence network, including 49 that have been filtered out. He focuses on this subgraph and notices states outside his assigned region restored to the canvas, including CA, RI, and others. He also finds that the most common hospital among these physicians is OSU Wexner Medical Center. He takes note of the promising lead as well as any other bridging nodes in the hopes that his colleagues may be able to help.

6 Expert User Study

We obtained feedback on Graphicle through formal and informal user studies conducted with pharmaceutical experts using the neurologist dataset. Throughout the design and development process, we received continuous feedback on the system’s features and functionalities. The final design reflects significant iterative improvements. To understand its value in the field, we recruited two regional coordinators for pharmaceutical field teams as well as a global medical director overseeing the development and deployment of a neurological drug. Their time spent in industry varied from 5 to 30 years. By engaging with domain experts, we were able to assess the general effectiveness of our design approach. Each participant was already familiar with specific physicians within the neurologist dataset, which encouraged both top-down and bottom-up exploration. All users held directorial or managerial roles within their organization with responsibilities for training others on their duties and were familiar with a wide variety of user tasks.

Each study session was one hour long and began with a 20-minute tutorial of the system and its features. The explanation was lengthy due to users’ unfamiliarity with visualization tools and approaches. Following the tutorial, we gave participants 30 minutes to use the system for free-form exploration. We encouraged them to use the tool in a way that would best support their own tasks or tasks that would be common for the domain. A think aloud protocol was followed during the studies. Following the exploration phase, a short semi-structured interview was conducted to obtain more feedback. The interview focused on soliciting information about user roles, tasks, and the ability of Graphicle to support them. The feedback was then transcribed, annotated, and discussed among the research team. Of special interest were instances where users expressed that they had learned something new in the dataset, the related action or feature that helped reveal that information, and instances where users expressed confusion at the behavior of the system.

Regional coordinator 1 (RC1) focused on bottom-up exploration. He used the search bar to locate a known physician within his region and began selecting related physicians using the long press action. Within a few steps, he was able to find another neurologist with whom he was unfamiliar. Examining the data card for that physician, RC1 realized that the unknown neurologist completed their postdoc training in the same location as himself. He then searched for other neurologists that had completed residencies in the same location. RC1 noted that the particular usefulness of being able to select networks for an individual. Describing the process of scheduling meetings with a known contact’s “circle of trust”, he observed: “[the circle of trust] is something that is difficult to figure out...but I can see here that they are comfortable [referring to] these other doctors.” The other regional coordinator (RC2) used the tool from a broader strategic role. During part of her exploration, she clustered all physicians by state and began to look at the distributions of pediatricians within those states. She commented on this functionality during the semi-structured interview, noting that it was easy to use the tool for such “big picture” planning and articulated how useful this would be for her organization in understanding the overall landscape. While discussing the value of Graphicle, she stated: “if I wanted to think about strategies for training purposes or helping people at the field level understand their territory [Graphicle] provides significant value to that.” The global medical director (GMD) explored the functionalities of Graphicle more broadly. His investigation was largely driven by identifying an individual physician and then expanding the scope, emphasizing the importance of search, elaborating along a path, and capabilities of bringing back context. His analysis focused on referrals within and across cities, which Graphicle facilitated through the specification of user-specified clusters. Moreover, the GMD commented, “What we need a tool like this for is identifying the upcoming people — upcoming [key opinion leaders]”, suggesting the potential for strategic decision making.

All participants primarily expressed concerns about the limitations of the dataset. All wanted to have more connection types available, both implicit and explicit. For example, being able to show connections between physicians who shared the same residency or co-authored publications would be valuable. Displaying two separate context bars was confusing to users. Both RC1 and GMD used the “bring in” functionality heavily during their exploration, but expressed frustration as units would shift between bars because of this interaction. While technically needed to distinguish contexts, it was evident that it was less useful to users. Lastly, while Graphicle provides functionality to undo actions (all the way to the beginning of the analysis), the GMD voiced his preference for providing provenance details through breadcrumbs.

Overall, the expert feedback we received provides evidence that combining network structure with unit visualization facilitates flexible exploration of real-world datasets. Participants engaged in both top-down and bottom-up exploration, were able to find patterns of interest across unit distributions as well as individual units of interest, and used network structure and filtered context to readjust the exploration space. It should be noted that all participants stayed within the grid view, where our blended approach is most prominent, despite training on the scatterplot and network views. However, some features within the grid view — particularly inter-cluster connections — were unused by participants. This could be due to inexperience with the tool or a lack of tasks to support. A deeper investigation into this issue is needed.

7 Discussion

In the beginning of this paper, we put forth several questions about the design space between unit visualization types. We now return to them and show how they relate to the tasks required for efficient multi-scale, multi-focus, and multi-path exploration of large multivariate graphs. Graphicle embodies an effective approach to answering these questions and supporting these tasks.

The relationship between unit-driven and structure-driven exploration (Q1) is distinguished by (1) the meaning and use of unit position and (2) the use of structure as a navigation tool. In unit-driven exploration, position is defined by data attributes and values. This allows groupings and orderings that define relationships between units based on data characteristics. Users can explore these relationships to com-
The range of values being selected along each axis are clear. When units are clustered in a packed unit visualization, rubber band selection can also be well defined if the packing has careful ordering. Graphicle orders units left-to-right, top-to-bottom within clusters to allow ranges of the ordering data attribute to be selected. Combined with the ability to remove or isolate selections, this enables rapid and flexible filtering.

In structure-based exploration, unit position is influenced by the other units. Proximity, rather than explicit position, is the basis for comparison and exploration. In a force-directed node-link diagram like the network view, proximity implies that units share common connections. However, because the meaning of position is implicit, exploration is better facilitated through the navigation of network structure encoded by edges. Selection also becomes focused on structure. Graphicle supports structure-based exploration through force-directed layout options and the ability to select and explore 1-step, 2-step, and full networks from units of interest. Once again, combining selection with filtering allows users to rapidly explore the data. The ability to remove or isolate networks through direct selection is powerful and supports tasks focused on comparing subgraphs or looking for bridges between them.

By integrating structural characteristics into packed and direct unit visualizations (Q2) and unit-focused characteristics into network unit visualizations (Q3), the boundaries between unit-based exploration and structure-based exploration are softened. Our approach focuses on layering connection information onto units through interaction. This enables consistent structural navigation between views and allows users to examine connections within and between clusters. Furthermore, the attributes of connected nodes can be explored in a flexible manner. Consider the task of examining the relationships among highly connected nodes. In Graphicle, units may be sorted by degree to quickly identify the nodes that are highly connected. Their connections to and from other highly connected nodes are easily seen within their own clusters and across others while maintaining awareness of the nodes that are unconnected. In large node-link graphs, this task would likely require filtering the dataset by degree, removing the context in which the highly connected nodes are found.

Structural characteristics can be further combined with unit-focused characteristics (Q4) after examining the relationships between structural context and containers (as embodied by clusters in our approach) and container context. Both intra- and inter-cluster relationships need to be considered. This opens up a spectrum of possibilities as to how the features are combined (as shown in Fig. 5). By allowing users to flexibly expand and contract unit groupings while remaining organized in context, Graphicle provides a multi-focus, multi-scale way to explore large scale graphs. The power of this sort of exploration is supported by previous work [41,46]. Unlike previous approaches, however, our groupings are maintained as packed unit containers rather than aggregations or abstract boundaries, allowing for more detailed exploration of the distributions and units contained therein.

The combination of structural context and filtered context also opens up several novel design possibilities (Q5). In order to save screen space and prevent occlusion, large scale unit visualizations must remove filtered units from the view. Graphicle introduces a context bar that aggregates removed units into blocks and allows users to selectively restore filtered context. Filtering in, then filtering back out to find units of interest is a powerful exploration technique that allows iterative and bidirectional analysis paths. This technique is especially important when potentially relevant structural context has been removed. Our approach presents one possible design alternative, but based on feedback it is evident that additional improvement opportunities for context exploration and interaction exist. It also opens up renewed considerations into the intricacies of combining filter widgets with the direct removal and restoration of units. Our intuition suggests that a tight coupling between these two modes of filtering can provide users with better navigation of the exploration space but it is difficult to reflect its free-form nature in traditional widgets or action histories.

We have successfully shown that the scatterplot and grid view can handle tens of thousands of nodes and hundreds of thousands of edges (the network view is severely limited by the performance of force-directed layout algorithms). Several design decisions allow for a large-scale approach. First, zooming and panning allows navigation around the large dataset. Second, the grid layout can be rapidly calculated using simple algebraic expressions. Third, edges are shown on demand and the related network properties are calculated as needed for subsets of the data. Fourth, the clustered network layout presents a novel approach to distributing the calculation of force-directed subgraphs. However, we acknowledge that the cardinality of qualitative attributes has an impact on scalability. As the cardinality increases, fragmentation increases when grouping into clusters. If there are many small groups, units on the canvas become highly scattered. Zooming and panning mitigates the issue and the user can quickly select and remove small groups through direct selection to clean up the canvas.

There is a need for future technique work in several areas addressed by this paper. We need to consider how multiple connection types change the meaning of structural context and what opportunities there are for designing systems that allow multiple types to be explored simultaneously or sequentially. Changes in data or network structure over time could also reveal important information about units rising to prominence within a dataset. Moreover, many-to-many relationships between units and unit containers (i.e., set-based unit membership) is unaddressed by our work and other unit visualization approaches. Graphicle also places limitations on the types of data supported by grouping (qualitative only) and sizing (quantitative only). Lastly, the distinction between position and proximity in graph exploration points to opportunities for using proximity to encode information in direct and packed unit visualizations. Dimensionality reduction techniques or a grid view that positions units radially from a point of interest might be considered, for example.

8 Conclusion

In this paper, we have examined the challenges and opportunities of combining unit visualization approaches in order to support large scale, multivariate network exploration. We characterize the design space between direct, packed, and network unit visualizations and discuss the meaning of context and its role in each. By taking this approach, opportunities for designing new layouts and interactions are revealed. These features not only expand the exploratory power of unit visualizations, but create consistency in how exploration can occur across different views. Our findings are embedded in a web-based system, Graphicle. We show how users are able to combine top-down and bottom-up exploration in a multi-focus, multi-scale, and multi-path approach in order to view patterns within and across data groups, identify units of interest, and find meaningful relationships between units. Though our current dataset was situated in the healthcare domain, the designed system and techniques are domain-agnostic. Large scale, relational data is pervasive in many areas, including those seeking to understand the composition and collaboration of large research organizations. Each of these datasets may reveal exciting opportunities to expand our work and further explore the blended unit visualization design space.

Acknowledgments

The authors would like to thank Bruce Lavin MD, MPH and his team for feedback throughout the project, the three participants of the expert study, Richard Sturr and Matt Sanders for data support, and Sakshi Prapat and Dhruv Chand for contributions in earlier stages of the project. The research was supported in part by UCB Inc. (Award #RH175).

References
