Accelerating Data Analysis with Visualization Recommendation

Kanit Wongsuphasawat and Dominik Moritz

Areas of Interests: User Interfaces, Visualization, Recommender Systems, Data Analysis

As business and academia become increasingly data-driven, more and more data analysts may be novices in statistics and data visualization. Accordingly, it becomes important for tools to better guide users towards productive analytic processes and effectively designed visualizations.

For example, consider an analyst examining Broadband Internet Subscription per Population (*Subscription*) over time in South American countries. As a line chart is a well-suited representation for temporal data, our system will recommend a line chart showing average *Subscription* over *Time* and another line chart showing *Subscription* over *Time* across different *Countries*. If the analyst becomes interested in the *Subscription* in 2013, our system will subsequently present 2013 data as a sorted bar chart (to facilitate comparison) and also suggest a map (for investigating spatial relationships). In addition, as *Subscription* is correlated with each countrys *GDP per capita*, our system will recommend a scatterplot to show the relationship between *Subscription* and *GDP per capita*. With these suggestions (Figure 1), the analyst can quickly examine relationships and generate more hypotheses without having to manually instruct software to plot each chart one-by-one.

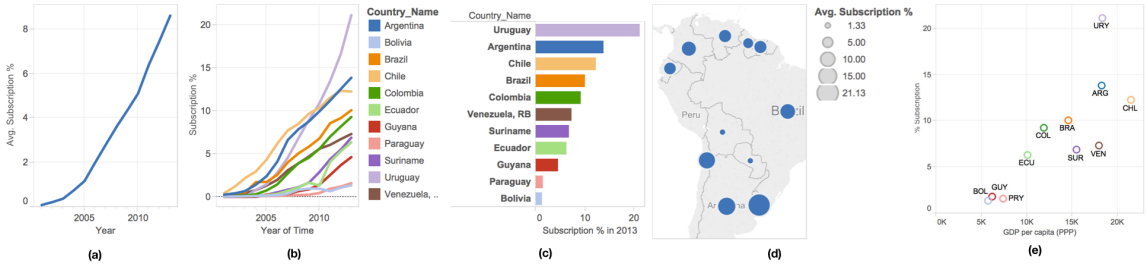

Figure 1: (a,b) line charts showing Broadband Internet Subscription per Population over time in South American countries (on average, and grouped by country respectively) (c,d) a bar chart and a map showing the Subscription in 2013 (e) a scatterplot showing relationship between Subscription and GDP per capita

To augment analysts ability to perform data analysis, we propose to build a recommender system for visualizing relational data. The system will consist of the following components.

1. Interface for Rapid Exploration. The interface will include expressive user controls for searching visualizations, enabling analysts to focus on iterative exploration and refinement rather than lower-level design details.

2. Comprehensive Recommendation Algorithm. What makes a recommendation good depends both on the data and task at hand. Accordingly, we will develop a recommendation algorithm that balances user input, best practices (e.g., perceptual effectiveness rankings) for visualization design, data properties and diversity.

3. Scalable System. With larger data, current visualization systems become less responsive or even fail to run at all, preventing rapid exploration. Furthermore, our recommender system will require expensive querying and modeling operations. We will develop a scalable client-server architecture with the goal of supporting interactive response rates.

Background

It is often argued that the most effective visual analytics tools should support analysis at the rate of thought [10]. However, existing tools [11] have not yet achieved this vision. *Chart Typology* interfaces, such as a palette of chart templates found in spreadsheet software, are easy for view creation but difficult for view refinement. *Visualization Toolkits* [2, 26] enable intricate designs but require coding, and therefore hinder rapid data exploration. Tableau [22], a state of the art visual analysis tool, enables rapid view exploration refinement and supports expressive visualizations creation. However, creating an effective visualization in Tableau still requires both tool and design expertise. To lower

this barrier, Tableau's Show Me [20] feature automatically suggests suitable chart types for selected data attributes based on design practices [5, 24]. Nonetheless, it can suggest only a single visualization at a time although there are usually many appropriate visualizations.

A few research prototypes [7, 13, 14] recommend a list of visualizations based on statistical properties of the data. While these systems may recommend relevant subsets of data, they produce only a limited range of visualizations and do not incorporate best practices of design. Moreover, they lack interfaces for users to express their intention, leading to irrelevant suggestions. An effective visualization recommender must address these limitations to successfully support users analysis.

User Interface

Our system will help users focus on data exploration rather than design details by letting users search for visualizations. Figure 2 shows a mock-up of our interface. The interface will highlight data attributes and chart types as two main search facets [28]. Prior work indicates that people usually begin describing visualizations by specifying these two properties [8]. Users can also specify data transformations and mappings between data attributes and visual variables. The browser view (Fig. 2, middle) will show recommended visualizations based on user input. Users can then select and interact with a visualization of interest on the right side. The interface will also support the iterative nature of visual analysis. Analysts can keep refining their queries with more specific intents. Our interface will also provide annotation and provenance tools to support the analysis flow.

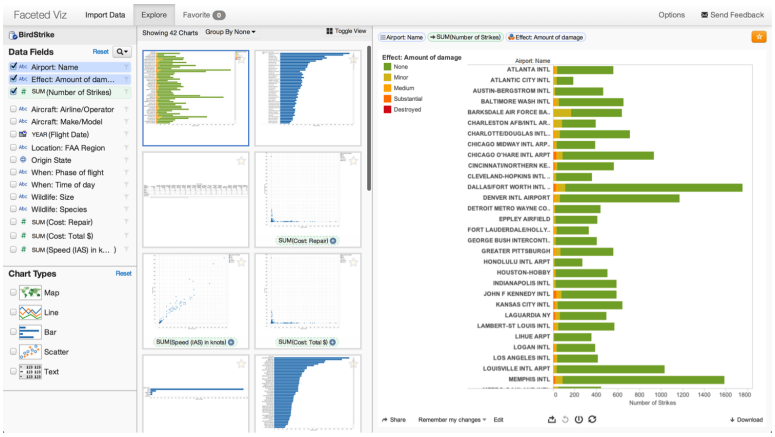

Figure 2: Mock-up of our interface. Users can specify their query using controls on the left side. The browser in the middle displays recommended visualizations. The interactive view on the right shows the selected visualization.

We will build an open-source, scalable interactive visualization module for rendering visualizations, taking specifications generated by the recommendation algorithm as inputs. The module will incorporate design practices such as determining the optimal aspect ratio for the chart type and the data [23]. Output views will support user interactions [11] for data exploration such as highlighting, brushing and linking, zooming, and filtering. The views will also allow the recommender to automatically highlight inferred anomalies and trends.

Recommendation Algorithm

The recommendation process will involve *generation*, *ranking* and *pruning*. The generation algorithm will create *relevant* visualization specifications, which includes permutations of possible mappings between data attributes and visual variables, and combinations of marks types and data transformations that match user queries. The system will then rank visualizations using a quality metric that we will develop and evaluate as part of this research. Finally, as top-ranked visualizations may contain redundant information, we will use a similarity metric to prune the results to ensure diversity.

Our quality metric will combine multiple criteria including *design quality* and *interestingness*. The *design quality* score will integrate design guidelines based on graphical perception studies [4]. For example, in perceptual effectiveness rankings of visual encodings of quantitative data [19], length is more effective than angle. Therefore, our algorithm will a priori prefer bar charts to pie charts. To determine an *interestingness* score, we will apply statistical measures such as mutual information [25] as well as anomaly and trend detection [3]. We will explore the use of interpretable machine

learning techniques [15] to combine our metrics, enabling our system to explain why a particular view is recommended. In the long term, we will extend the algorithm to learn from user interactions.

Scalable System Architecture

Our visualization system will use a client-server infrastructure with a web-based client. The backend server will use database techniques to efficiently store and query data for the visualizations. Data can reside in the client, the server's memory or in persistent storage. We will develop a query optimizer that automatically determines whether a particular computation should be computed on the client or on the server based on data size, data location and latency. Spare server resources will be used to find and profile anomalies and trends, expediting *interestingness* score calculations. Finally, we will explore big data techniques including index precomputation and data cubes to support fast brushing and linking [16, 18], data reduction methods such as sampling to simplify complex data, and online aggregation [1, 6, 12] to enhance responsiveness for larger data. We believe that coupling database systems and interactive visualizations will be performant and enable new functionalities [27].

One Year Horizon

This research is a multi-year project. We expect to deliver a working system for general datasets that fit in memory in the first year.

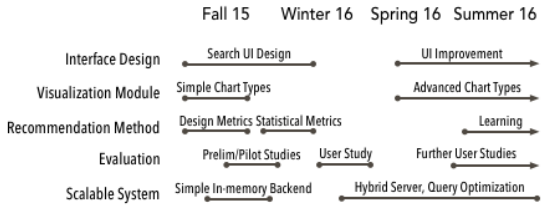

Figure 3: Timeline during the 2015-2016 Academic Year

Figure 3 shows our project timeline during the 2015-2016 Academic Year. in the beginning, we will focus on developing and evaluating user interfaces and recommendation algorithms. The initial system will support commonly used chart types including histogram, bar chart, line chart, scatter plot and map. We will perform a lab study comparing our recommender system with existing tools such as Tableau. We will measure the rate of events such as participants observations, generalizations and questions during the analysis sessions [17].

During the first half of the year, we will also start the implementation of the backend server. In the latter half of the year, we will focus on scaling the system to support higher volume of data and advanced visualization types. We will also improve our user interface and recommender algorithm based on user study results, and run further user studies.

Team

Our work will be a major step towards visualization systems that support analysis at the rate of thought. For this kind of work to succeed, it is critical to have an interdisciplinary team with expertise in both Human-Computer Interaction and Data Management Systems. Our experience across these areas makes our team a strong candidate to tackle these challenges. In addition, at the University of Washington we have strong collaborators in the Interactive Data Lab, the Database Group, and the eScience Institute.

Kanit Wongsuphasawat is a Ph.D. student in Computer Science & Engineering and a member of the Interactive Data Lab at the University of Washington. Kanit has expertise in Human-Computer Interaction and Visualization. He is a co-developer of a declarative model for interactive visualization design [21]. Kanit has been awarded a Fulbright Fellowship and H.M. the King of Thailand scholarship. He also has professional experience working in leading data-driven companies including Google, Tableau Software and Trifacta.

Dominik Moritz is a Ph.D. student in Computer Science & Engineering at the University of Washington. As a member of both the Database group and the Interactive Data Lab, Dominik has a strong background in data management systems and visualization. He co-develops Myria, a distributed database system [9] and was a core developer of CKAN and creator of a number of libraries for data management and sharing. He has been awarded a Fulbright Fellowship and a scholarship from the German National Academic Foundation.

References

[1] Agarwal, S., Mozafari, B., Panda, A., Milner, H., Madden, S., and Stoica, I. BlinkDB: queries with bounded errors and bounded response times on very large data. In *Proceedings of the 8th ACM European Conference on Computer Systems*, ACM (2013), 29–42.

[2] Bostock, M., Ogievetsky, V., and Heer, J. D: Data-Driven Documents. *IEEE transactions on visualization and computer graphics 17*, 12 (Dec. 2011), 2301–9.

[3] Chandola, V., Banerjee, A., and Kumar, V. Anomaly detection. *ACM Computing Surveys 41*, 3 (July 2009), 1–58.

[4] Cleveland, W. S., and McGill, R. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association 79*, 387 (1984), 531–554.

[5] Few, S. *Now you see it: simple visualization techniques for quantitative analysis*. Analytics Press, 2009.

[6] Fisher, D., Popov, I., Drucker, S., et al. Trust me, I'm partially right: incremental visualization lets analysts explore large datasets faster. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2012), 1673–1682.

[7] Gotz, D., and Wen, Z. Behavior-driven visualization recommendation. In *Proceedings of the 14th international conference on Intelligent user interfaces* (2009), 315–324.

[8] Grammel, L., Tory, M., and Storey, M. How information visualization novices construct visualizations. *Visualization and Computer Graphics, IEEE Transactions on 16*, 6 (2010), 943–952.

[9] Halperin, D., de Almeida, V. T., Choo, L. L., Chu, S., Koutris, P., Moritz, D., Ortiz, J., Ruamviboonsuk, V., Wang, J., Whitaker, A., et al. Demonstration of the Myria Big Data Management Service.

[10] Hanrahan, P. Analytic database technologies for a new kind of user: the data enthusiast. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, ACM (2012), 577–578.

[11] Heer, J., Shneiderman, B., and Park, C. Interactive Dynamics for Visual Analysis A taxonomy of tools that support the fluent and flexible use of visualizations. *Queue 10*, 2 (2012), 1–26.

[12] Hellerstein, J. M., Haas, P. J., and Wang, H. J. Online aggregation. *ACM SIGMOD Record 26*, 2 (1997), 171–182.

[13] Kandel, S., Parikh, R., Paepcke, A., Hellerstein, J. M., and Heer, J. Profiler: Integrated statistical analysis and visualization for data quality assessment. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, ACM (2012), 547–554.

[14] Key, A., Howe, B., Perry, D., and Aragon, C. VizDeck: self-organizing dashboards for visual analytics. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data* (2012), 681–684.

[15] Letham, B., Rudin, C., McCormick, T. H., and Madigan, D. An interpretable stroke prediction model using rules and bayesian analysis. In *AAAI (Late-Breaking Developments)* (2013).

[16] Lins, L., Klosowski, J. T., and Scheidegger, C. Nanocubes for real-time exploration of spatiotemporal datasets. *Visualization and Computer Graphics, IEEE Transactions on 19*, 12 (2013), 2456–2465.

[17] Liu, Z., and Heer, J. The effects of interactive latency on exploratory visual analysis. *Visualization and Computer Graphics, IEEE Transactions on* (2014).

[18] Liu, Z., Jiang, B., and Heer, J. imMens: Real-time Visual Querying of Big Data. In *Computer Graphics Forum*, vol. 32, Wiley Online Library (2013), 421–430.

[19] Mackinlay, J. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics 5*, 2 (1986), 110–141.

[20] Mackinlay, J., Hanrahan, P., and Stolte, C. Show me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1137–1144.

[21] Satyanarayan, A., Wongsuphasawat, K., and Heer, J. Declarative interaction design for data visualization. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, ACM (2014), 669–678.

[22] Stolte, C., Tang, D., and Hanrahan, P. Polaris: A System for Query, Analysis, and Visualization of Multidimensional Relational Databases. *IEEE Trans. Vis. Comput. Graph. 8*, 1 (2002), 52–65.

[23] Talbot, J., Gerth, J., and Hanrahan, P. Arc length-based aspect ratio selection. *Visualization and Computer Graphics, IEEE Transactions on 17*, 12 (2011), 2276–2282.

[24] Tufte, E. R., and Graves-Morris, P. *The visual display of quantitative information*, vol. 2. Graphics press Cheshire, CT, 1983.

[25] Wang, J. J.-Y., Wang, Y., Zhao, S., and Gao, X. Maximum mutual information regularized classification. *Engineering Applications of Artificial Intelligence 37* (2015), 1–8.

[26] Wilkinson, L. *The Grammar of Graphics*. Springer, 2005.

[27] Wu, E., and Madden, S. R. The Case for Data Visualization Management Systems [Vision Paper]. 903–906.

[28] Yee, K.-P., Swearingen, K., Li, K., and Hearst, M. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM (2003), 401–408.