Research Statement and Agenda

Eugene Wu
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
eugenewu@mit.edu

I build systems that make it easier and faster for data analysts to transform, track changes in, visualize, and interact with complex datasets. Specifically, my thesis has focused on how data provenance systems[1] can be used in combination with interface design to make analysis more humane, and with machine learning to augment analyst decision making. I developed Scorpion [8] to explore a user interface and algorithms to let users visually identify and explain the source of aggregate outliers (e.g., "this month's sales were surprisingly high") in a dataset. Scorpion needs to efficiently retrieve the source data of any user-selected output, so I built Subzero [9], a workflow system that tracks record-level provenance that can tune the indexing, materialization, and encoding behavior of the provenance information based on user-specified runtime and storage constraints. I am currently building Smoke [6], a low-latency provenance system that simplifies the way developers build user interactions in data visualizations by modeling interactions as simple provenance queries.

Background and Research Approach

My previous research projects fit into my over-arching interest in systems and algorithms that simplify the way end-users interact with and express complex analyses over data. This is critical because a broad audience is quickly becoming attuned to the importance of data-driven decision making. Thus, tools cannot assume the users are experts with deep understanding of data management techniques, but instead are domain experts that need simple ways to express and manage complex analyses. Thus a full solution must tackle challenges at each step of the data analysis pipeline – from simplifying the process of loading data into management tools [3], improving core query performance through relational optimization principles such as partitioning and indexing [4, 7], extending database expressiveness to new forms of data [5, 1], helping the user understand outliers in their query results [8], and extending database optimization techniques to visualization interaction [6].

My research approach emphasizes builting end-to-end systems that are workload driven. In each research project, I strive to identify unique properties of the workload that can be leveraged to both improve query performance and directy how the queries can be expressed through language extensions or visual interfaces.

Projects

Data analysis extends beyond simply manipulating and extracting patterns from our data, but also the ability to reason and explain the consclusions that we reach. While (existing techniques) can make analyses faster, data provenance is a key component of the latter.

The bulk of my thesis work has focused on practical data provenance systems and their application to data analysis. An ideal data provenance system will automatically and efficiently track the relationships of data as it is transformed within a workflow and provides an efficient query interface over the provenance information. Such a system has broad applications in many domains (e.g., scientific computing, data analysis, entity resolution, security). However, tracking these input-output relationships at a fine-grained (per record) level in a data-intensive environment is non-trivial – a single transformation can generate $O(N^2)$ pair-wise relationships for a dataset of size N. In addition, it is unclear how data analysts and systems can effectively *use* these provenance capabilities to enhance their analyses. The SubZero, Scorpion, and Smoke projects investigate the performance, applications and usability of fine-grained provenance systems.

A Low Overhead Provenance System

Scientific applications such as LSST are now demanding provenance at the record or pixel level (e.g., "what pixels generated this star?") for debugging and data validation purposes. However scientific workflows also have tight constraints on the amount of storage and runtime overhead they are willing to incur – LSST processes 2GB/sec each night, but can only allocate 20% of their storage to provenance information, and must process each pair of images within 15 seconds. Many operators are vectorized, so even the act of generating provenance information can cause operators to slow down by orders of magnitude. Finally, science applications regularly use custom operators, and the provenance system must provide an efficient means to expose the operator's provenance information.

[1] Data provenance systems track the input and output relationships of data as it is aggregated and transformed in a workflow. An example query is "What input records generated this output record?"

I proposed and built *SubZero* [9], a provenance system that separates the mechanisms of *how* provenance is specified and stored from decisions of *what* provenance to generate. To efficiently expose operator provenance, we differentiate classes of operators by the amount of provenance they need to store (constant, linear, or polynomial with the output dataset size) and develop efficient APIs for each class. This allows operators to incur the cost of generating and writing provenance in proportion to their complexity. When SubZero executes a provenance query, it can make use of previously stored provenance information, or re-run previous operators to generate provenance. This lets the optimizer make policy decisions that trade off between provenance query performance and the amount of runtime and storage overhead by deciding which operators should generate provenance and what their encoding and indexing strategies should be.

Using Provenance to Explain Outliers

As datasets become larger and analysis pipelines grow in complexity, even making sense of query results becomes very difficult. Consider a simple analytic query that computes a company's total expenses by month, and shows that last month's expenses was unexpectedly high. The analyst will naturally want to understand why – perhaps the company has put more resources into a new customer demographic, or a department is overspending. Currently, the analyst must manually split the input data along different dimensions (e.g., dept, customer age), and hope that re-running the query will cause changes. If there is more than one outlier, or many dimensions in the dataset, this ad-hoc process quickly becomes untenable. As a real-life example, a local medical researcher spent six months manually performing a similar process to investigate why a small number of lung cancer patients cost tens of millions of dollars. (It turns out that two doctors that over-perscribed chemo treatment were responsible a significante amount of the costs).

Scorpion is an example of an interactive system that uses provenance capabilities to answer these "why" questions. The analysts simply selects outlier and normal results and the system constructs predicates that most influenced the outliers. Our work formalized this notion of predicate influence in terms of sensitivity analysis, provided a framework to search for influential predicates, and identified aggregation operator properties that can be leveraged for faster search heuristics.

Low Latency Provenance for Interactive Visualizations

Increasingly, people are publishing data as interactive visualizations, and while tools for creating static visualizations and animations are prevalent, there are limited tools for creating rich interactions – many visualization authors manually implement and optimize interactions such as brushing and linking. Our main contribution is to establish the parallel between many forms of visual interaction and data provenance. By modeling a visualization as a workflow from raw data to visual elements, many user interactions can be expressed as provenance queries and potentially leverage an expressive query interface and performance optimizations. I proposed and am building a provenance system that can execute provenance queries at interactive (<100ms) speeds based on two key insights. First, when designing a publishable visualization, the author can specify the exact set of provenance queries apriori for the system to optimize. This is impossible to do in existing systems designed for ad-hoc queries. Second, many interactions do not need all of the source data specified by a provenance query, and a sample of the data, their ids, or a summary statistic is often sufficient. Thus we are defining classes of provenance queries with reduced expressivity that can be used for more optimizations.

Improving Data Analysis Along Other Dimensions

The above three projects are designed to move the state of data analysis to one where an end-user can effectively use the expressive power of data provenance in a visual environment. In addition to provenance-related projects, I have been broadly interested in, and worked on other projects that simplify the analysis process.

Complex Event Processing

Sensor devices generate streams of unreliable, raw sensor readings (e.g., "RFID tag 123 was read"), however sensor-based applications expect high level event notifications (e.g., "shoplifting at register 3"). Diao and I built one of the first high-performance complex-event detection systems that lets users declaratively specify high level events from patterns of raw or other high level events. Our main insight was to compile sequence patterns into an NFA and introduce relational optimizations such as filter push-down and early pruning.

Human Computation

Marcus and I developed one of the earliest data management systems that introduces large scale human computation as a new class of query operators. Our work described the trade-offs between query latency, cost, and result quality, and implemented an asynchronous query engine that takes into account wildly unpredictable response times and the three

trade-offs. Furthermore, our results identified the tight coupling between how the task interface is designed and the result latency and quality. We recently proposed [2] a system that uses a synergistic relationship between data intergation in data spaces and news consumers that want more context about data in the news articles they read.

Future Research

Large scale interactive data analysis is still a burgeoning field and there are a huge number of fascinating research directions in this space.

In the short term, there are many valuable extensions to my thesis. Scorpion only scratched the surface of explanatory analysis by focusing on simple statistical functions. However, the general framework could be extended to explain more complex aggregations as well as data constraint violations (e.g., functional dependency violations), which are crucial features when cleaning and integrating datasets. Smoke is built into a full-featured visualization system, which I plan to release and gather usage information about how the visualization and provenance system. I hope this information can be developed into a fine-grained provenance benchmark that is currently missing in the field.

(I am not happy with this) There are also opportunities to apply database optimization and recommendation techniques by leveraging user-level cues as they interact with a visualization. For example, a selection box has a limited number of directions it can be extended, which can be used as prefetching hints when the user hovers over the edge of the selection. Similar hints such as known perceptual inaccuracies, interaction history and output resolution are promising directions for optimization techniques.

In general, the intersection of databases, usability and interaction is an area ripe for future research. Data analysis is ultimately driven by a human being, and while reducing the cost of query execution is certainly important, the bottleneck is more often centered aronud the analyst. How can analyst tasks such as picking the questions to ask, understanding query results, and testing hypotheses be offloaded to the system so the analyst can fully utilize her domain expertise and shift her role from *implementer* to *decision maker*?

References

[1] A. Marcus, E. Wu, D. R. Karger, S. R. Madden, R. C. Miller, et al. Crowdsourced databases: Query processing with people. In *CIDR*, 2011.

[2] A. Marcus, E. Wu, and S. Madden. Data in context: Aiding news consumers while taming dataspaces. *DBCrowd*, 2013.

[3] E. Wu. Dbtruck: Humane data import, October 2012.

[4] E. Wu, C. A. Curino, S. R. Madden, et al. No bits left behind. In *CIDR*, 2011.

[5] E. Wu, Y. Diao, and S. Rizvi. High-performance complex event processing over streams. In *SIGMOD*. ACM, 2006.

[6] E. Wu and S. Madden. Smoke: Visualization interactions using provenance (in preparation).

[7] E. Wu and S. Madden. Partitioning techniques for fine-grained indexing. In *ICDE*, 2011.

[8] E. Wu and S. Madden. Scorpion: Explaining away outliers in aggregate queries. In *VLDB*, 2013.

[9] E. Wu, S. Madden, and M. Stonebraker. Subzero: a fine-grained lineage system for scientific databases. *ICDE*, 2013.