Tessera: Discretizing Data Analysis Workflows on a Task Level

Jing Nathan Yan
Cornell University
Ithaca, New York, United States
jy858@cornell.edu

Ziwei Gu
Cornell University
Ithaca, New York, United States
zg48@cornell.edu

Jeffrey. M Rzeszotarski
Cornell University
Ithaca, New York, United States
jeffrz@cornell.edu

ABSTRACT

Researchers have investigated a number of strategies for capturing and analyzing data analyst event logs in order to design better tools, identify failure points, and guide users. However, this remains challenging because individual- and session-level behavioral differences lead to an explosion of complexity and there are few guarantees that log observations map to user cognition. In this paper we introduce a technique for segmenting sequential analyst event logs which combines data, interaction, and user features in order to create discrete blocks of goal-directed activity. Using measures of inter-dependency and comparisons between analysis states, these blocks identify patterns in interaction logs coupled with the current view that users are examining. Through an analysis of publicly available data and data from a lab study across a variety of analysis tasks, we validate that our segmentation approach aligns with users' changing goals and tasks. Finally, we identify several downstream applications for our approach.

CCS CONCEPTS

• **Human-centered computing → Visual analytics**; • **Information systems → Users and interactive retrieval**.

KEYWORDS

Data Analytics, Interaction Log Analysis, Visualization

ACM Reference Format:
Jing Nathan Yan, Ziwei Gu, and Jeffrey. M Rzeszotarski. 2021. Tessera: Discretizing Data Analysis Workflows on a Task Level. In *CHI Conference on Human Factors in Computing Systems (CHI '21), May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3411764.3445728

1 INTRODUCTION

Interactive visual analytics tools for exploratory visual analysis (EVA) help to support analysts as they explore large scale, complex datasets by augmenting human capacities for information storage and processing. With computational support, analysts can explore many more points and data attributes in a session than they could manually inspect. By providing visual representations which more efficiently present data and affording interactions which aide in encoding requests such as filtering into computational actions, EVA tools help analysts develop an understanding of their data. Interaction, in particular, is key because an analyst's understanding of the data is not developed instantaneously, but rather over a period of time. Analysts speculate, gather evidence, (dis)prove hypotheses, and make presentations or summary findings over many iterations. Effective EVA tools help to hasten this iteration by providing analysts a means to quickly inspect and manipulate data.

However, the same features which augment analysts' exploration of data can also make it difficult for them to track their progress [12]. Further, tools may unwittingly introduce biases into the process which may be hidden by rapid iterations [60]. Once an analysis is complete, it may be hard for an analyst to recall all of the steps that lead to their findings, obscuring data provenance as well as slowing integration of new knowledge into practice [9]. As a result, there is a growing community of researchers investigating approaches for making sense of the stream of interactions an analyst conducts through an EVA tool. Event logs are difficult to manage - a successful analysis may take take minutes or hours, during which the practitioner completes many operations per minute. Further, activities such as an analyst pausing and thinking or turning to external resources are not indicated in the data. Additionally, these log data are not necessarily comparable across users, datasets, and tasks.

A number of approaches exist for processing behavioral traces of EVA sessions in order to deliver actionable results. These approaches range from identifying cognitive bias [60] to automatically constructing provenance data for an analysis [9]. Some tools employ hierarchical or graph-based structures to make inferences about higher level features. Hierarchies, while human-interpretable, require customization for tasks and may not reach sufficient levels of granularity on a sub-task (where a user is focusing on a specific, small goal as part of a larger exploration) or session-level (where the focus is on an entire analysis session, composed of many goals) basis. On the other hand, graph-based models can abstract well across classes and deliver both retrospective data from their structure and prospective predictions for future sessions. However, their structure may be hard to interpret and can be swamped by large scale data since they are better suited to small-scale analysis.

We introduce a parallel approach for creating higher level abstractions from behavioral log data aimed at segmenting logs into blocks of goal-directed or task-oriented user behavior by approximating users' current intentions and data coverage. Our technique integrates traditional signals from event logs, inferences about user exposure to data, and analysts' data transformations as reflected by their interface state in order to model the EVA process. Additionally, we analyse behavior over windows of time, detecting the revisiting of past states which mark looping behaviors during an EVA session that can indicate hypothesis testing or iterative improvement. Because we focus on identifying segments of activity,

CHI '21, May 8–13, 2021, Yokohama, Japan

ACM ISBN 978-1-4503-8096-6/21/05…$15.00
https://doi.org/10.1145/3411764.3445728

we can provide simple abstract representations of one "block" of activity which explain the actions that occurred during a section of goal-directed user behavior (e.g. investigating the relationship between attributes α and β). While our approach lacks the predictive ability of graph-based models and does not make inferences based on pre-defined patterns, it more efficiently and accurately splits a streaming log of an analyst's activities into segments of self-reported goal-directed behavior that might be used to power classifiers to identify sensemaking process stages, recommend new avenues of exploration, or provide reflective visualizations of an analyst's EVA history.

In this paper we describe our approach, Tessera, and its implementation. As a baseline, we compare its performance at segmenting workflows against two benchmark algorithms (hierarchical- and graph-based techniques) using a publicly available dataset of goal-directed analysis. In order to further understand the connection between segments of EVA activity and users' self-reported intentions, we assemble a new dataset of open EVA performed by both novices and skilled users using lab think-aloud studies and compare model performance using this new dataset. We find that Tessera outperformed benchmark algorithms both in terms of recognizing when analysts were shifting from one goal to the next and in terms of compressing log data into a more manageable scope. Finally, we outline potential extensions of our approach and downstream applications. We provide the following contributions:

- We introduce Tessera, a framework for post-processing EVA event logs which considers data, cognitive, and temporal features to abstract user interactions into goal-directed activity segments.
- We assemble a dataset combining think-aloud reports by analysts of varying skill levels with raw interaction event log data.
- We conduct comprehensive experiments to demonstrate the effectiveness of Tessera compared to benchmarks.

In the following sections we begin by outlining related literature in the EVA and databases communities. We then describe theoretical and technical details for Tessera. Through a public dataset and our own independently-gathered think-aloud data, we describe an evaluation of Tessera. Finally, we identify potential limitations and extensions.

2 RELATED WORK

In this section we outline several different threads of work that are related to exploratory visual analytics (EVA), provenance, and behavioral modeling. Our goal is not to exhaustively survey every attempt. Rather, we aim to identify some of the key threads that distinguish different techniques and connect them to our broader understanding of analytics.

2.1 Exploratory Visual Analysis (EVA)

EVA involves the active exploration of data using interactive, visual interfaces in order to derive insights or achieve specific goals [61]. One aim for developers and researchers of EVA software is understanding the different goals, phases of exploration, and activities that individuals engage in while they conduct EVA. Understanding these features is of immense value when tracking the provenance of analysis results [51]. While the research community has examined how individuals direct their explorations, there is no dominant definition of the task structure and goal-direction of EVA [9]. Recent work pursues two major streams. On one hand, EVA can be considered generally as a practice of gaining insights from data [21, 31, 37], which does not necessitate that users come into the process with clearly defined starting goals and states.

On the other hand, EVA can be characterized as a process of evolving from a starting state with vague goals towards more and more sophisticated representations [26, 62]. Gotz and Zhou [27] suggest that users switch between discovering insights and recording insights. However, [28, 32, 49] argue that in EVA, users are engaging in iterative improvement which is flexible and may involve mixing different activities. Moreover, based on the nature of the exploration (i.e. open-end tasks vs. focused analysis) the goal and task structure could differ substantially [3, 24, 59]. Additional work has focused on how these factors influence the pace of EVA, going beyond the scope of goal orientation [33, 39].

Rather than debating between explicitly defined goals and evolving goals, some recent advances have changed focus towards task-level behavior and user behavior at different levels of granularity. If an overarching goal is hard to explicitly define, then it might be easier to decompose the analysis into a series of smaller, more tractable units. This is the track that we take in the development of Tessera. We don't intend to produce a full hierarchy of the EVA's goal or motivation structure. Rather, we hope to identify task-level units which fit together to describe a priori what happened in a session in more tractable and generalizable form.

2.2 Mining the EVA Process

In psychology and cognitive science a number of papers have explored how individuals reason about data and develop findings or insights based on their investigations [34, 50]. Often, this work discusses task- and session-level features such as goal orientation and iteration by individuals. Following along these lines, researchers in the interactive visualization community have proposed parallel frameworks describing how individuals use tools to reason about data [9]. One focal point of this work is abstraction. While low-level interactions are often the atomic elements of observation, the meaningful factors that describe how a user is reasoning about their data lay at a much higher level of abstraction, necessitating techniques for bridging the gap. For example, a semi-automated framework [27] captures low-level interactions in visual systems and abstracts them into sub-task units. Researchers have made a case that many general goals can be decomposed into sub-task elements [37] (which later map to atomic interactive or cognitive units). There are also strategies for modeling EVA interactions not in terms of goal-specific units but in terms of generalizable patterns across all investigations. Heer et al. [32] propose a task model for Tableau which connects interactions with five categories of visual exploration steps. In the domain of image editing, [40] also proposed data-driven approaches to identify a user's breaking points to reveal their intentions at various parts of the process.

Approaches for abstracting analysis steps into tasks or generalizable units is a growing area of focus in HCI, data management/mining, and machine learning. The potential benefits of these approaches are clear - they may contribute general knowledge about

how individuals reason about data and digital systems [50]; represent opportunities for new, beneficial cognitive interventions[60]; encode data that can guide automated systems [17, 22, 23]; and offer new ways to help users disseminate or track their analyses [9]. The general intuition is to segment or connect the series of small interactions logged in the visual system to understand the paths or patterns that a user employs while doing EVA [18, 43, 56].

2.3 Modeling Behavioral Data

The logged steps of an EVA session using an interactive system are essentially a time series stream of interaction events [31]. Sequences in the time series potentially encode higher level semantic meaning (e.g. goal-directed behavior) and patterns highlight higher-level structures (e.g. iteration) [11, 26, 31]. Recent research has proposed techniques for identifying common patterns in event sequences [26] and sub-sequences [11]. [62] leverages interaction events to generate visualization recommendations. Similarly, [10] computes a common sub-sequence from a sequence of interactions to predict the following interactions. However, prior research suggests that relying solely on interaction sequences may not capture more complex, global features of an analysis [20, 51]. To capture more complex structures, recent approaches [17, 22, 26, 27] turn towards more complex representations of user behavior which attempt to characterize the deeper semantic meanings underpinning segments of user activity. For example, such approaches construct state-based representations which may be easier to encode/decode and reveal complex structural information through their own organization (e.g. connectivity in a graph).

One common technique makes use of a hierarchical model [5, 55]. A taxonomy hierarchically defines the relationship between tasks, sub-tasks, actions and events [27]. The underlying idea is to organize the low-level sub-goals into higher level ones. However this approach requires manual labeling and taxonomy adaptation for specific applications. It also may be the case that behavioral sequence data cannot be cleanly binned into some hierarchical levels [18]. However, lower-level tasks in the hierarchy may be sufficient in order to derive beneficial findings [13]. The importance of interactions between goals and low-level interactions is also highlighted in [8]. While the high granularity of these approaches may be beneficial, it does come at a cost of reduced global- or session-level information.

Probabilistic approaches are another common strategy for modeling behavioral log data. Such techniques model transitions between interaction states [6, 42]. Often, these take the form of a Markov model or one of its siblings. By asserting that each lower-level event is the state symbol, a finite Markov chain models the probabilistic transitions between states. In this way, the model provides an ability to evaluate common patterns and loops in an analysis. However, the focus on transitions does come with risks [42]. On one hand, the underlying assumption that future states are conditioned only on current states may not be consistent with prior work on sensemaking and information diffusion [30, 65, 66]. On the other hand, when applying a probabilistic model for prediction, it is generally assumed that the users' activities are independent which again might break with our understanding of analyst cognition. A similar markov-expression automata approach [18] was introduced to merge similar states and eventually output aggregated paths. However, this approach requires advanced partitioning of the logs in order to function.

Machine learning models have also been employed to infer user activities from discrete log data. However, explainability of results and the potential feature explosion of a behavioral log state space make incorporating models challenging at present [23]. Additionally, these models pose a greater risk that bias may be unwittingly encoded into the model due to the opacity of many machine learning algorithms [7].

In our work, we seek to offer a complementary strategy for segmenting and modeling logs. We explicitly focus on sub-task and task segmentation, as these seem to be both the most commonly employed levels of granularity in analyzing/employing behavioral log data and have not been fully "solved" by the research community. While our approach may suffer similar issues on a global level as in hierarchical modeling, we have designed our method with an aim of enabling recursive application of the segmenting. In the future, we propose integrating multiple rounds of segmentation to create a hierarchical structure or connecting segments together to build aggregated probabilistic models. In this way, Tessera might benefit from some of the prior work done on segmentation.

2.4 Connections to Other Research Communities

Research advances in data management and data mining research provide further opportunities for improving behavioral analytics. Frequent pattern mining, a common approach for deriving frequent patterns in time series data and temporal data, has been adapted to web logs which share similar issues to EVA in terms of size, complexity, and goal-directed behavior [4, 35]. EVA techniques have also been adapted and used in the database community for use cases such as improving data quality [58], generating meaningful intermediate visualizations [8, 38, 41], and human-in-the-loop interactive analysis [1, 45]. To support those applications on large scale data, database researchers have focused intensely on scalability. One research thread is approximate query processing [1, 2, 16, 48] which carefully constructs queries on strategic data regions instead of entire databases in order to give rapid results. Database learning predicts users' incoming queries [25, 47] by leveraging and reusing previous query results, much like some applications of behavioral models in EVA. These approaches have primarily focused on prior query histories rather than per-session features. Grid-object discovery in Explore-by-Example[19] proposed data operators that can be used for recommendation queries in the interactive data exploration. We take from this body of literature some of the concepts from mining similar queries in order to improve the overall segmentation of Tessera. By examining data-level and behavioral time-series features simultaneously, we hope to better estimate the task-directed behavior of users.

3 MODELING USER BEHAVIOR

In this section, we describe the design of Tessera and outline the data features that we employ. In prior work, interaction logs have been modeled in order to gain insight into the analytics process. However, algorithms for processing logs have focused solely on data

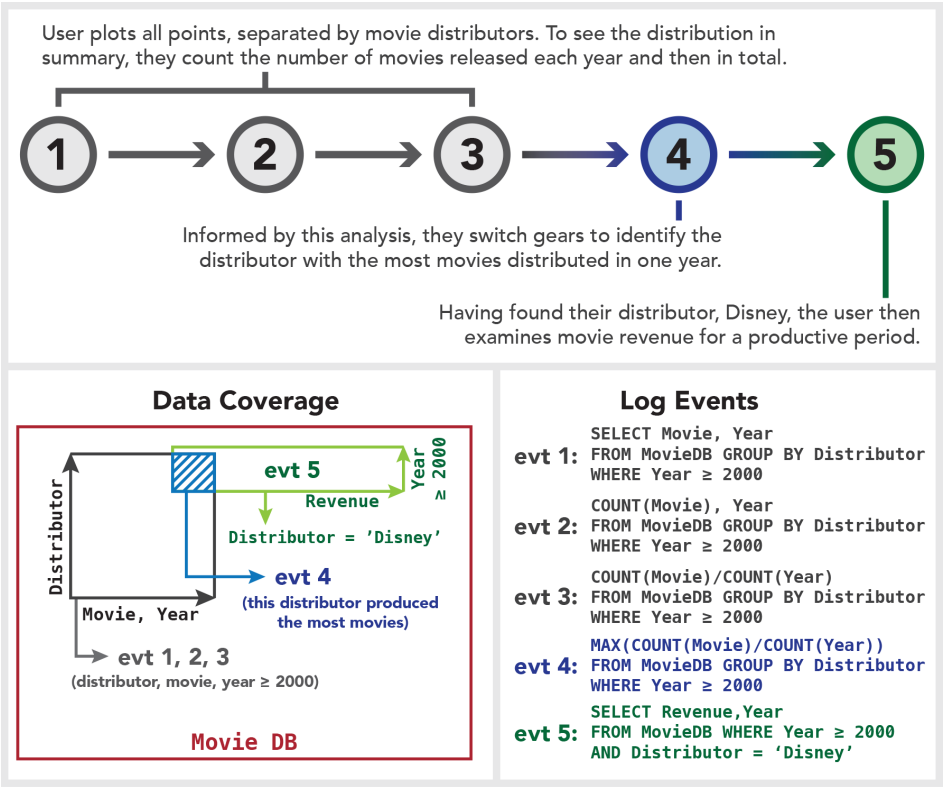


Figure 1: User activities, recorded interaction log events in reduced SQL form, and data coverage for the analyst use case outlined in Section 3. The events describe 3 periods of activity in which a user explores distributions, identifies a target, and then examines that target with respect to new data attributes.

transformations in the process or on modeling user activity. Less attention has been paid towards mapping relationships between user activity and data transformations during an analysis.

In developing Tessera, we categorized user activities during EVA into different levels of granularity. *Individual-level features* refer to singular log events or short segments of events through which an analyst might identify usability issues or find patterns of use. However, given the number of logs could be very large, it is hard to evaluate individual-level, *local* features in isolation. On the other hand, *session-level features*, referring to a set of events for an entire analysis session, help to reveal *global* information about an EVA session such as bias or data coverage. While there are fewer sessions in an event log dataset, they are much more complex than individual events and may represent multiple separate patterns of activity. In Tessera we aim between these two foci, focusing on smaller segments of activity which indicate sections of goal-directed behavior within a session. We refer to these goal-directed segments of log activity as *task-* or *sub-task level units* of activity, depending on whether they are the few macro-scale blocks of goal-directed behavior during a session (e.g. identifying outliers) or the plethora of smaller blocks of behavior used to accomplish those goals (sub-tasks, e.g. identifying outliers by examining the distribution of attribute A).

In Tessera we consider both visual elements (as mapped to data by visual channels) as well as the interactions themselves as representations and manipulations of the underlying queries made to the data by the user. Through this formalism we attempt to remain agnostic across different visualization systems and visual metaphors, with the caveat that our approach may neglect to consider the cognitive impact of a particular choice of visualization (such as a scatterplot drawing attention to outliers more effectively than a pie chart). We make this compromise not only because it is hard to estimate these factors by inference from interactions without a set of hard-coded heuristics, but also because a query-based formalism allows us to draw an advantageous connection between interaction events and the data themselves. One benefit of this approach is that in cases where one signal is weaker (e.g. the user is making small changes in an interface), the other signal may make up the difference (e.g. small interactions such as reconfiguring data leading to large shifts in data coverage in the visual interface). This is especially useful when connecting user behavior across events in cases where adjustments in terms of visual metaphor (e.g. bar chart, scatterplot) do not actually affect the results of the underlying query of the data and coverage. However, pairing up these two features is non-trivial, and there will often be cases where one can be swamped with an explosion of measurable factors. In the following subsections we will outline how we construct query

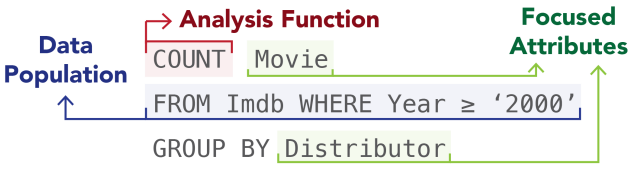


Figure 2: Extracted components from a user query during an EVA session

representations from interface log activities. Figure 1 shows user interactions, Tessera query representations, and a visualization of the space of data covered during several blocks of analyst activity, color-coded based on the Tessera segmentation.

3.1 Constructing Data Transformations

Imagine that an analyst is investigating a dataset of movie statistics. She is trying to find relationships between distribution companies, directors, genres, and the overall distribution of points in the dataset using an interactive exploratory visual analysis tool. The tool provides affordances for displaying points using a number of different chart designs as well as hooks for selecting, filtering, annotating, grouping, and styling points through interactive widgets. As system designers, we want to be able to examine her interactions with the tool in order to make inferences about her intentions. In our case, we might want to automatically generate a summary of her exploration after the session completes or provide a real-time streaming display of her exploration process.

Turning towards the event log data that we can easily gather from interactions with the tool, we can observe a few different factors. Figure 1 shows a few samples from this analyst's exploration session. Note that in this case the user is interacting with widgets, but we are making use of the underlying data queries performed by the system, as represented by Tessera (their extraction will be described later in this section). Based on think-aloud feedback from this session, we know that in this particular moment the analyst was trying to observe some basic statistics. She investigated which distributor produced the most movies for each year after 2000 in the dataset (log events 1-3). After finding that in this case Disney was the most prolific (log event 4), she followed up by investigating their specific revenue (log event 5). What inferences can we make about these interactions and their associated data?

In order to make sense of the analyst's coverage of data and interactions, we visualized all of the data points of the movie dataset using a rectangular projection in the figure. We can immediately observe that in the log-1 to log-3, the analyst was exploring the same information from an overview to average statistics. At step log-4, the target data region changed to the smaller blue region as the user tried to find the most prolific distributor (Disney in this example). Armed with that new information at log-5, the analyst turned towards looking at Disney's revenue across different years. The data transformation from black region, to blue, and finally to green shows how the changing focus of the user corresponded to changes in data coverage. The dramatic shift in projected area between steps 3, 4, and 5 indicates that this is a moment of transition from one period of focused activity to another. By leveraging this information, it is possible for us to infer when users are making small and large changes in their area of focus. However, it is challenging for us to abstract this level of information from the data transformation log, and moreover, to map it to a possible set of user activities (which can be informative in cases where the user is interacting but not changing data coverage across events). In order to merge information from this data coverage approach with specific activities carried out in the analysis, we make use of a two-step approach: first, we abstract user intention data from the log to understand users' target data, focused attributes, and analysis stages; and second, we map this high-level information to a set of possible intended analysis tasks/activities derived from the literature.

Figure 2 shows one example of a query from the analyst's activities. The structure of this query reveals several important pieces of information about their investigation. The element, *FROM Movie WHERE Year* $\geq$ 2000, identifies the specific data points that the analyst is inspecting. Further, the GROUP BY attributes, *Distributor* and *Movie*, gives us information about each data point on which the user might be focusing. One common task in EVA, filtering, is evident through the inequality observed earlier. One other factor at play in this query is the element, *COUNT*. This provides evidence of the specific task that the analyst is conducting, and often relates to the particular stage of an analysis (e.g. overview first, then drill-down) because these functions are used to compress and extract specific statistical information for datapoints over selected attributes. By identifying three components in a logged data transformation: *selected data points*, *focused attributes* and *analysis functions*, we can make inferences about the analyst's EVA workflow. This rough signal, connected over time, helps to reveal changes in focus and coverage as illustrated in Figure 1.

3.2 Selected Data and Focused Attributes

As mentioned in the previous section, we can decompose user data exploration actions, as reflected by their event log data, into multiple kinds of information. Examining query representations of analysts' selection, filter, and aggregation actions, we can identify the kinds of data points that they are selecting and the attributes on which they are focusing. Practically speaking, this often simply takes the form of sub-setting rows and columns from a relational table, where data selection picks out rows and attribute focus picks out columns. In the first stage of Tessera, we compare these subsets across multiple event log steps in order to identify differences and commonalities. Later on, we can use these comparisons in order to estimate when an individual has inflected from one segment of the analysis to the next, and identify the most essential events in a segment of activity.

One intuitive way for us to achieve this comparison is to simply examine the *exact* data points and attributes involved at each log step. Given two pieces of log data, log_1, log_2, with associated data points DP_{log_1}, DP_{log_2} (e.g. picking out movies released after 2000) and attributes $Attributes_{log_1}$, $Attributes_{log_2}$ (e.g. viewing *revenue* by *year*), we can define a distance function to estimate the similarity between the pieces of log data:

$$\textbf{DirectDistance}(log_1, log_2) = \alpha \frac{DP_{log_1} \cap DP_{log_2}}{DP_{log_1} \cup DP_{log_2}} + \beta \frac{Attributes_{log_1} \cap Attributes_{log_2}}{Attributes_{log_1} \cup Attributes_{log_2}}$$

Where α and β are weight parameters (s.t. $\alpha + \beta = 1$) which can be set by users to bias the similarity score towards data- or attribute-level similarity. We observe that often the focused attributes deliver more information about a user's intentions compared to the coverage of datapoints. One can argue that in many cases EVA tool affordances are oriented more towards selecting and aggregating across attributes rather than specifying data subsets.

Moving beyond comparing individual data points using similarity scores (a potentially naive approach in light of how EVA is actually conducted), the statistical meaning of a point distribution is critical in understanding connections between steps. If the user is looking at focused attributes (whether or not the selected data points have also been changed), we can use statistical distance comparison techniques to measure the similarity between two selected data points within these attributes as **StatDistance**($Distribution_{log_1}$, $Distribution_{log_2}$).

For example, we might want to use a Z-test[15, 44, 54] for numerical data values for an attribute. For categorical data values and mixed data values, we can use frequency-based approaches such as a chi^2[15, 44, 54] test to measure distance based on statistical differences.

In comparing event log steps on a data- and attribute-centric level, we can first apply $DirectDistance$ to measure the data transformation similarity. When two log steps are focusing on the exact same attributes, we can further apply the statistical distance measure $StatDistance$. We will revisit these metrics when we employ them as components of Tessera's final segmentation algorithm.

3.3 Analysis Functions and User Activities

Imagine that the following data transformations occurred during an EVA session:

Q_1 : AVG(budget) FROM movie GROUP BY Distributor
Q_2 : VAR(budget) FROM movie GROUP BY Director
Q_3 : Max(Budget) FROM movie GROUP BY Director

A key difference between Q_1, Q_2, Q_3 is the function employed to aggregate or summarize data values. Such functions are common in EVA, especially at large scales, as they allow an analyst to make sense of groups of data in aggregate. Most of these functions are directly supported in mainstream database systems (other functions can also be implemented through user-defined function programming interface). An *avg* function may indicate an intention of exploring the expectation of data, while a *var* function might imply that the user is estimating the stability of the selected data. While these two functions both focus on aspects of distribution, analysts may also use functions such as *max* or *top-n* which select specific points of importance for the analyst within the chosen data. Incorporating the statistical meaning of a query can help to trace how an analyst's intentions towards the data change and gauge their overall understanding of the data as it evolves [62, 68]. The wrong choice of function may lead to potential misinterpretation and false discoveries [68–70]. On the other hand, we can observe changes in these functions across events to make inferences about when and how analyst intentions are changing, potentially signaling a shift to a new segment of investigation. In order to accomplish this task, we consider these functional operations both in terms of their mathematical implications and their cognitive/goal-directed factors.

Functions	Interface Interactions	Common Visualization Elements	Common User Intentions
plot(col)	plot	Histogram, bar chart, scatterplot	Overview of dataset
Missing(col)	plotting, filtering	Histogram, bar chart, pie chart, stacked bar chart, heatmap, binned box plot	Overview of dataset; count datapoints that are missing; data preprocessing; data quality examination
Sum(col)	plotting, filtering, aggregating points	Histogram, bar chart, scatterplot	Overview of dataset; summarize information of groups
Avg(col)	plotting, filtering, aggregating points	Histogram, bar chart, scatterplot	Overview of dataset; summarize information of groups
Var(col)	plotting, filtering, aggregating points	Histogram, bar chat, scatterplot	Overview of dataset; examine stability of a data region; statistical inference
Count(col)	plotting, filtering, aggregating points	Histogram, bar chart, pie chart, stacked bar chart, heatmap, binned box plot	Overview of dataset; counting/comparing datapoints by condition or criteria
Max(col) Min(col) Top-k(col)	plotting, filtering, aggregating points, sorting points	Histogram, bar chart, pie chart, line chart	Find the maximal/minimal/top-k value of selected data region; refine a search with endpoints; refine groups; finding particular instances; overviews
MaximalCount(col) MinimalCount(col) MostFrequent-k(col)	plotting, filtering, aggregating points, sorting points	Histogram, bar chart, pie chart, stacked bar chart, heatmap, binned box plot	Find the maximal/minimal/top-k value of selected data region; refine a search with endpoints; refine groups; get an overview of a dataset
Correlation(col1, 2)	plotting, filtering, aggregating points, sorting points	Correlation matrix, QQ-norma plot, box plot, histogram with regression line	Understand the correlation of focused attributes in selected data regions; refine understanding of data relationships; get an overview of a dataset

Figure 3: Mapping of analysis functions and common intentions, derived from prior research [58] and adapted for EVA.

In order to generate a signal for Tessera to use, we first consider the mathematical foundations of functions commonly used in data exploration programs. Specifically, we focus on functions used for exploring distributions, selecting points, and aggregating information. These functions are employed in a wide variety of contexts (see Figure 3 for functions we focused on in this study).

In order to identify how similar two log events are in terms of the functions they use, we derive mathematical connections between statistical functions. We define COL to be the entire data series of a column selected by the analyst (e.g. every movie box office return), and col to be an individual entry. For example, given two analysis functions, $a(COL)$ and $b(COL)$, we can derive $b(COL)$ given the results of $a(COL)$. Making use of the mathematical representations of the functions used in common EVA platforms, we constructed derivations. For example, we know that $Sum(COL)$ can be rewritten as the product of the number of data points selected and the their average value:

$$\mathbf{SUM}(COL) \equiv N \times \mathbf{AVG}(col = x)$$

and variance can be rewritten as operations between expectation or average values:

$$\mathbf{var}(COL) \equiv N^2 \times \mathbf{var}(col = x) \equiv N^2 \times (\mathbf{E}(col = x^2) - \mathbf{E}(col = x)^2).$$

For Pearson's relationship calculation, we can rewrite it as the fraction of variance and expectation:

$$\rho(COL_1, COL_2) \equiv \frac{\mathbf{E}[COL_1\ COL_2] - \mathbf{E}[COL_1]\mathbf{E}[COL_2]}{\sqrt{\mathbf{var}(COL_1)}\sqrt{\mathbf{var}(COL_2)}}$$

In order to estimate the mathematical similarity between different analysis functions employed by an analyst and reflected in their event log, we estimate the number of steps necessary to derive one from the other in terms of functions and values. For example, if we are comparing $\rho(COL_1, COL_2)$ and $var(COL_1)$, we know in addition to $var(COL_1)$, we need expectation, so the functional similarity term is $\frac{1}{2}$. We also calculate three values, $E[COL_1\ COL_2]$, $\sqrt{var(COL_1)}$ and $\sqrt{var(COL_2)}$, out of four, so the value similarity term result is $\frac{1}{4}$. In sum we can express the similarity between these

two events as $MathSim(\rho(COL_1, COL_2), var(COL_1)) = \frac{1}{2}(\frac{1}{4} + \frac{1}{2})$ in terms of analysis functions. In addition to this linear combination, we can also use $log_a\ b$ form or entropy. This similarity metric allows us to capture the functional equivalency of different steps in an analysis. However, it neglects to consider the cognitive and behavioral aspects of each operation (i.e. the analyst's *intentions*). We compute an additional score to account for these factors.

In order to identify similarities between log events in terms of their human factors, we make use of prior work in EVA on analysis functions [58]. In this work, a team of researchers used semi-structured interviews to associate data analysis functions with specific, self-reported analyst intentions. They then negotiated and discussed with users to come up with a final list. In our work we adapt this framework in order to evaluate similarity between event intentions (see Figure 3 for example functions and intentions).

Using the framework, we constructed a rule-based model using intuitions from Jaccard Similarity [36, 57]. If, in two event log steps, analysts are using functions with similar labeled intentions, we assume that their functions have high similarity. In order to compute a numeric value, we intersect the list of the union of intentions of all of the functions an analyst employed in a step with the intentions of another step, as listed in the prior work and adapted here.

$$CogSim(fun_1, fun_2) = \frac{Intention_{fun_1} \cap Intention_{fun_2}}{Intention_{fun_1} \cup Intention_{fun_2}}$$

Putting the two components together, we compute the functional similarity between two log events using:

$$\begin{aligned}\textbf{FunSim}(fun_1, fun_2) = \alpha(MathSim(fun_1, fun_2)) \\ + \beta(CogSim(fun_1, fun_2))\end{aligned}$$

Where α and β are weight parameters (s.t. $\alpha + \beta = 1$) which can be set by users to bias the similarity score towards mathematic- or intention-level similarity.

3.4 Identifying Segments of Activity

In the previous two sections we constructed several pairwise similarity metrics for events. In this subsection we use these metrics in order to identify segments of activity within an event log. First, we compare log events and normalize results:

$$\textbf{Sim}(log_1, log_2) = \underbrace{\{\textbf{DirectDistance}(log_1, log_2) + \textbf{StatDistance}(log_1, log_2)\}}_{Data\ points\ and\ selected\ attributes} + \underbrace{\textbf{FunSim}(fun_1, fun_2)}_{Analysis\ Functions}.$$

(Larger similarity values imply that two event steps are more likely to be similar.)

We noticed that in a real world scenario analysts tended to focus on one segmented task within a window of time. Considering human factors, analysts are more likely to focus on recent steps in an analysis as opposed to distant ones when directing their behavior on a step-wise level [14, 52, 63] (this, of course, differs on a session level where planning exhibits more complex integration [50]). This recency effect has an advantageous property for the application of our similarity metric – it is not necessary to compare across all of the states in an analyst event log. By setting the length of look-back time window K, we can reduce computational costs while maintaining reasonable segmentation results. The notion of *recency* also implies that the connection between event log states also might decay over time. In addition to summing pairwise similarities in a window, we employed a temporal decaying function $\textbf{Decay}(\Delta|i-j|)$ (from [63]) to re-weight each pairwise similarity result where $\Delta|i-j|$ is the temporal difference between two log events. Hence, given the length of time window K and log_i, we calculate the normalized sum of decayed similarity at most $\frac{K}{2}$ steps before log_i and at most $\frac{K}{2}$ steps after log_i. Thus, the final similarity score of each state S_{log_i} can be expressed as:

$$S_{log_i} = \frac{1}{K}(\sum_{i}^{i\pm\frac{K}{2}} \textbf{Decay}(i \pm \frac{K}{2}) \cdot \textbf{Sim}(log_i, log_{i\pm\frac{K}{2}}))$$

Finally, in order to determine the bounds of a segment we use a simple heuristic. Given the threshold and within the time window, if the difference between pair-wise S_{log_i}, S_{log_j} is smaller or equal to the threshold m, Tessera considers that the analyst is focusing on a similar segment. We discuss the selection of K in our experimental evaluation.

Additionally, analysts are often able to store or save the states they find interesting for the future revisitation [33]. However, revisitation doesn't necessarily reflect that one is changing their segmented goal – often it is only evidence of a comparison between present state and past state. In order to accommodate these affordances in log analysis (which side-trips analysts might make more generally), we use a continuous equality. Given one sequence of temporal log events $\{log_1, log_2, log_3, log4, log_5\}$, if we find that $S_{log_1} \equiv S_{log_2} \equiv S_{log_4} \equiv S_{log_5}$, (whether S_{log_3} is large or small with respect to other logs' similarity scores), we can accept that those steps are focusing on one sub-goal. For example, at log_3 the analyst might have revisited the previous stored results to compare with the current results they just received. Figure 4 illustrates Tessera's entire workflow.

4 EXPERIMENTAL EVALUATION

In order to estimate the overall effectiveness of Tessera with respect to different analysis tasks and other similar tools, we conducted a series of evaluations. Our core motivation is to determine whether Tessera and alternative algorithms can accurately identify when users are engaging in different segments of analysis activity on a granular level. While one might evaluate in terms of whether the model can *predict the next event* that a user will trigger (e.g. a probabilistic Markov model [42]), our interest is in higher level, task-directed behavior. We instead investigate whether a model can accurately *predict breakpoints that indicate task-switching* during a participant session. This indicates that the model is effectively segmenting the workflow. To do so, we make use of event log data that is paired with observational data.

We evaluate the effectiveness of Tessera with respect to other existing approaches using two different datasets. First, we employ publicly available logs of analyst behavior, as recorded in interface event logs, paired with think-aloud observations. These data

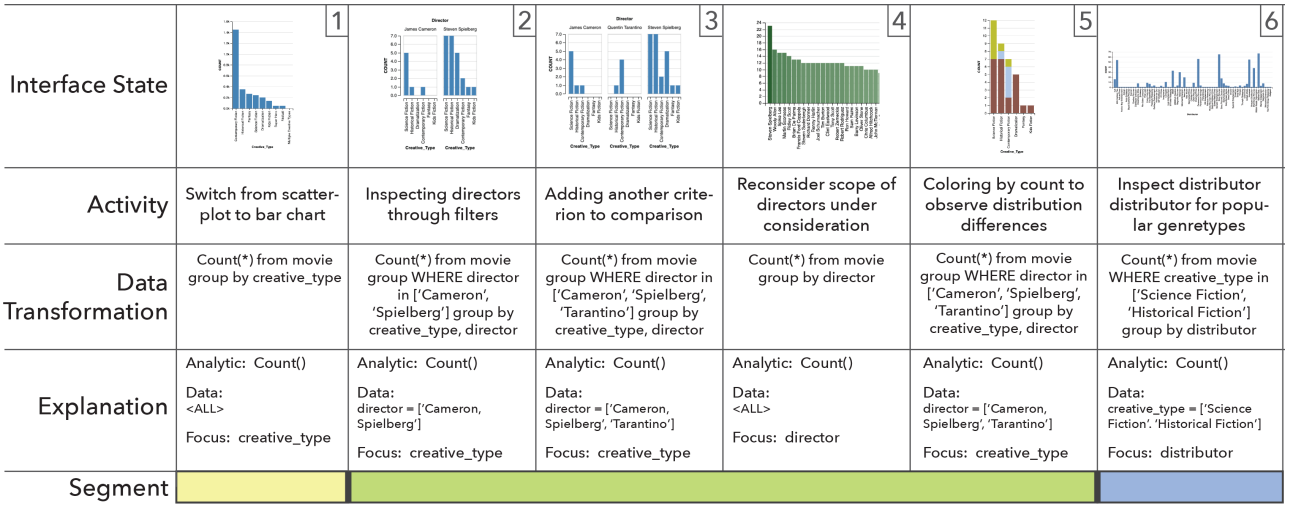


Figure 4: Individual exploration steps in EVA with paired user intentions, SQL representation, data coverage, and segmentation (color blocks) produced by Tessera. In this case, an analyst is investigating how different types of movies and directors connect to the distributors in a movie dataset. Beginning with a summary view (step 1), they use filters to call out specific directors with whom they are familiar (steps 2, 3). As they explore, they reference an earlier view (step 4) to recall their scope. Tessera correctly identifies that this is a momentary switch and not a redirection. They return to their view in step 5, examining distribution. Armed with some candidate movie types, they switch to investigating distributors (step 6).

contain flags that indicate that a user is switching between different segments of goal-directed activity. Second, we gathered a new dataset of analyst interface logs paired with think-aloud observations. While the first dataset is primarily *goal-oriented*, as participants were directed to accomplish a specific task, our second dataset focuses on *open exploration* which may be more heterogeneous and unpredictable [9]. In both cases, we compare the performance of Tessera against two benchmarks - a hierarchical model and a graph-based model. We evaluate the performance of the models at predicting segments of activity using a variety of time windows in order to reduce the rate of false reflections and expose a variety of different use cases for models (e.g. providing real-time feedback versus post hoc classification of sessions).

Through our evaluation, we seek to answer the these questions:

Q1: How performant is Tessera in real-world situations, and how does its performance compare to benchmarks?

Q2: How is performance affected by model time windows, and where is the best accuracy/efficiency compromise?

Q3: Will Tessera effectively reduce the amount of logs by recognizing redundant segments of events?

Q4: How is segmentation accuracy positively or negatively affected by data features, tool features, or task features?

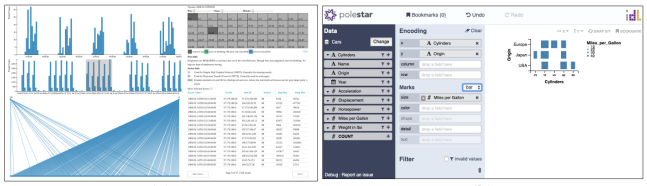
4.1 Prior Data and Benchmarks

4.1.1 Cyber threat analysis dataset. Researchers from Texas A&M University gathered a series of data focused on a cyber threat analysis task. Not only did they gather event log data of user interactions with a web-based analysis tool, but they also collected think-aloud participant responses in the form of transcripts, videos, and eye tracking data [43]. The dataset is available online [1]. The participant reports indicate when participants switched between segments of goal-directed behavior. As the think-aloud data are paired with video timestamps, we were able to pair these indications with the event log data to create a dataset for evaluating Tessera. In the dataset, 8 participants were recruited for a 90-minute session study where they analyzed data from 2009 VAST Mini Challenge[29] using a visualization tool specifically built for the study analysis (Fig 5 (a)). The dataset has approximately 250,000 events and 13 think-aloud reflections per participant. While some participants met more success, all engaged in goal-directed data analysis.

In assembling our dataset pairing think-aloud observations with logged interface events, we observed that participant self-reports may reflect some degree of latency between the logged system times and the times when participants verbalized a response. For example, a participant may be so immersed in the task that they forget to verbalize their thought process until prompted by the study proctor. This delay between recorded events induced by the need for the proctor to probe the participant may affect the quality of the ground truth data – even if a model accurately identifies task-switching behavior, it may not yet be reported. Further, there are also latencies between event logs and participants introduced by interface latency, delays in recording by the researcher, and potential offsets in time-stamping after the session (e.g. due to differences in system and researcher timekeeping).

In order to characterize this potential latency on a participant-by-participant basis, we matched video recordings with key moments in the event logs (e.g. a filter being triggered and reported). We use T_r to denote the time when the participant clearly expressed an

[1] https://research.arch.tamu.edu/analytic-provenance



(a) (b)

Figure 5: (a) Analysis platforms used for the prior cyber threat analysis dataset (adapted from [43]), (b) Analysis platforms for our national census income analysis dataset (modified PoleStar[53])

activity in recorded logs, and T_s to denote the time stamp in the log when the activity was logged as an interaction. Latency error can be expressed as $|T_r - T_s| \leq \Delta t$. We found that latency was an issue in this dataset both due to the complexity of the task and the 90 minute duration. In our own lab we also observed latencies, though, informed by these results, we sought to minimize them when generating our own dataset. In both cases we compensate for latency by establishing a Δt value for the ground truth test datasets.

4.1.2 Benchmark approaches. So that it might be easier to interpret the final results of our evaluation, we created benchmarks. Our goal was to enable comparisons between predominant strategies for post-processing event logs in order to extract higher level features. We implemented two different approaches:

Hierarchical approach: One common strategy for modeling workflows makes use of a hierarchy of behaviors. Encoded into the HARVEST system, [27] developed a semi-automatic framework to characterize user behaviors. A key element of this strategy is the formalization $Action =< Type, Intent, Parameters >$. *Type* refers to visual analytic system interaction activities. *Intents* are pre-defined to capture the kinds of intentions that might occur when engaging in interactions during exploration. *Parameters* provide additional information about the previous two elements. The entire framework taxonomy and example can be found in Table 1 of [27]. To construct a comparable framework for the two visual analytics systems employed in our evaluation, we adjusted the elements provided in the original work. We excluded annotate, brush create, change metaphor, pan and zoom types as they are not supported in either of the two visual systems used to collect data. To implement the approach, our team processed all annotations manually.

Graph-based approach: Another strategy for modeling workflows centers around constructing network or graph-based structures out of sequences of events. Particular focus is placed on merging similar states to prevent an explosion of nodes. By employing the K-reversible algorithm to merge states, [18] create a simplified graph structure reflecting changing interface states. States that convey a similar semantic action are merged together. Since [18] can set a K value to aggregate states that are similar, we adapt this to see whether this approach can identify sub-task structure across an entire analysis session. We adapt this approach, asserting that merged states represent a segment of directed exploration. Since K is a parameter here, we reported the best scoring results across variations of K in our re-implementation.

4.2 Gathering Data for Open-ended Exploratory Analytics

Though the cyber threat dataset reflects one potential data analytics scenario, the dataset itself does not contain a rich variety of attribute- and record-level information which might be challenging for analysts to uncover. In this case, the analyst is challenged more with inferring connections between atoms of data than inspecting highly multidimensional points to identify trends. Additionally, the dataset concerns a relatively niche task which may not generalize to tasks common in broader exploratory data analytics literature, and focuses on goal-directed analysis which may not adequately extend to open-ended exploration of data [9, 21].

To better understand the effectiveness of Tessera in comparison to other approaches across all of exploratory data analytics, we gathered additional data. Using PoleStar, a well-studied interface for basic multivariate data analysis[2][53], we gathered think-aloud, video, and interface event log data for 21 participants completing exploratory data analysis tasks in a laboratory environment. Participants engaged in open-ended exploration using a modified version of PoleStar followed by semi-structured interviews. As in the previous dataset, think-aloud reports were analysed for signals of task-switching and paired with event log data (adjusted for latency) in order to generate testable ground truth data for our evaluation. In the following subsections we outline the details of our laboratory study.

4.2.1 Study Design. All participants were recruited through a university research participant pool and screened on prior exposure and experience with data analysis. We recruited individuals using a pre-screen so that we could have balanced groups of two categories of participants: 1) novices who do not have experience with working with data analytics beyond introductory university coursework, and 2) skilled participants who at least have some practice conducting data analysis but have never analyzed our study dataset. Participants were binned into these groups based on threshold rules. We later made use of the pre-screen demographic responses as a comparison point in our post-survey analysis.

In total, 21 individuals participated in our study. Of those participants, 20 completed the entire protocol and submitted usable survey responses. 1 participant chose to leave the session early and did not finish the task. 8 participants identified as male and 12 as female. All participants reported to be current university undergraduate students. 11 participants ultimately fit into our Skilled category and another 9 fit our Novice category. Participants had a 1/2 chance of being randomly selected for a post-session semi-structured interview. Ultimately, 6 participants in each skill group participated in an interview.

During our user study participants used PoleStar to complete an *open-ended* exploration task. They were instructed to explore a dataset of salaries in order to understand the factors that influence income. The salary dataset, commonly known as *Adult*[3], collects anonymous national census information into 13 attributes and 40,000 records [67]. Participants in both groups were familiar with the attributes information collected in the dataset, though the

[2]http://vega.github.io/polestar/
[3]https://archive.ics.uci.edu/ml/datasets/adult

proctor was on hand to explain any data attributes if the participant was unfamiliar. This open-sourced dataset has been widely studied in previous research (e.g. machine learning prediction, bias evaluation). No direct investigation goals were provided beyond that initial framing.

To explore the dataset, participants made use of a version of PoleStar (see Figure 5 (b)) [4] modified to gather additional event log data. PoleStar is a tableau-style data visualization system built upon a higher-level grammar Vega-Lite [53], which has been widely used in the research community. Its interface is relatively simple and straightforward, which minimized the training burden for participants. Further, its design focuses on assembling data attributes and adjusting filters, which map well to the kinds of signals Tessera and the benchmark models employ. While a richer, more fully featured tool would offer additional expressability (and potentially environmental validity for industry practice), we view PoleStar as a compromise between the needs of our evaluation, training costs, and similarity to other existing analytics systems.

Participants had 30 minutes to explore the data. Afterwards each participant was asked to answer two summary questions about their analysis experience. During the lab study session a researcher encouraged participants to verbalize their thought processes following a traditional human-computer interaction *think-aloud* protocol [46], including but not limited to: what they were doing at this step, why they were doing it, and whether they were shifting from one task/goal to others. Participants received specific instructions to be sensitive to changes in their goals or to moments when they were "shifting gears" to another part of their analysis. For example, one participant reported "I was choosing the sex attribute since I want to see how it affects people's salaries". All of PoleStar's interaction events as well as their verbalizations were recorded throughout the session. Additionally, the study proctor took timestamped notes to aide in later analysis of the think-aloud data. Participants who were randomly selected for a semi-structured post-interview were asked to reflect on how they analyzed the data and to explain how the tool helped (or hurt) their ability to explore the data. In follow-up questions the participants were probed for specific examples and evidence to support their reports.

During the lab study, one research team member placed a timestamped marker in their log any time that the participant verbalized that they were shifting from one sub-task to another. To verify these markers, another research team member viewed the think-aloud audio and video streams and compared timestamps. Similar to the prior cyber threat dataset [43], the moments of the topic/task change were encoded as an inflective boundary between segments of goal-directed activity. For example, one participant was spending time interacting with the system, reporting, "I'm going to start playing around with the system". This was marked as the beginning of one task segment. The moment when they said "I'm gonna start my analysis now", another marker was placed to indicate the switch to a new task segment. Another participant reported a task switch by thinking aloud, "I am now trying to understand how the increase of decrease of hours-per-week impact the salary level," and ended this task by saying "I am done with analyzing the relationship between hour-per-week and income label." Our PoleStar event logs contained approximately 500 events and participants averaged 18 inflection markers per session.

As in the cyber threat dataset, we evaluated the paired logs for latency introduced by participant reports, system features, and logging. In our observation of lab study sessions, the average Δt per individual was approximately 10 seconds, a degree of latency lower than that of the cyber threat observations. We believe that two factors are at play in this difference: (1) compared to the threat analysis sessions, our lab sessions duration was 30 minutes, reducing potential participant fatigue; (2) there may be differences in how the think-aloud protocol was applied between studies.

The feedback collected through our semi-structured interviews was used as evidence of possible sensemaking paths users were pursuing as they explored the data. As these qualitative recollections are not easily paired with the event logs and don't naturally align to segments of think-aloud recordings, we report these separately in summary form. Primarily, we made use of the semi-structured interviews as a way to check the quality of our think-aloud data, as deviations between the participant reflections and our think-aloud logs might indicate a lack of reliability creeping into the study methodology. We did not observe any such misalignment, and make use of the qualitative feedback later in the paper as we discuss potential applications of Tessera.

4.2.2 Final Dataset. We have made our final dataset publicly accessible in an open source repository [5]. This dataset reflects the behavioral traces of 20 participants, split by performance and aligned with think-aloud responses. Timestamped think-aloud reports allow for independent investigation of latency. When permitted by participants, we include qualitative response data from interviews.

4.3 Evaluating Tessera

4.3.1 Data processing. We conducted evaluations across our two datasets (Cyber Threat and Adult) and three tools (Tessera and two benchmark models). We make use of three different metrics in considering the overall performance of Tessera with respect to the other approaches. *Precision*, describing the fraction of task shifts returned by the model that match ground truth, helps us gauge whether the model is effective at recognizing task shifts while avoiding false positives. On the other hand, *Recall* describes the ratio of the total shifts in the ground truth and the amount of shifts that the model was able to identify. This highlights whether the model is sensitive to all kinds of task shifts, or if it misses a portion of them (which might lead to an interface that is blind to some kinds of behavior). Finally, we employ *F-score*, the harmonic mean of *precision* and *recall*, to gauge overall performance of the models. For our model setting, we set the threshold to be $m = 0.2$, and we also observed that the threshold did not influence the goal results significantly.

Earlier in this section we mentioned that there is an observed latency Δt in logged shifting time stamps and the actual time in which a task shift occurred. To better validate our results, we considered a model response, T', to be correct if it was within $T \pm \Delta t$ range of the ground truth time stamp, T. By varying this parameter, we can enforce more or less rigid adherence to the logged data. While

[4] https://github.com/vega/vega-lite

[5] https://github.com/NathanYanJing/Tessera

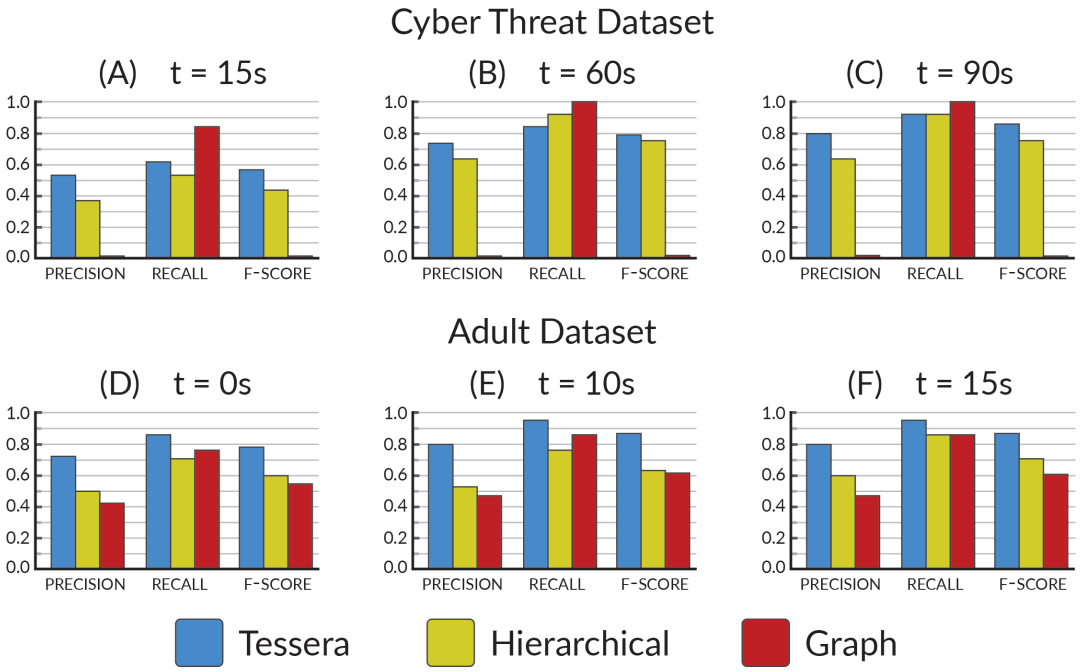


Figure 6: Precision, Recall and F-score results of Tessera, hierarchical model, and graph-based model

a large t might accommodate higher amounts of latency, it comes with the cost of potentially overstating the accuracy of a model. On the other hand, too low a t risks underestimating performance. As a result, we report across a variety of t values ranging from the latencies observed in the cyber threat dataset to zero latency (cyber threat has a larger range of windows as t was tuned for each dataset). Figure 6 illustrates the precision, recall and f-score of different approaches using different t. As anticipated, larger windows are associated with higher performance estimates.

In Figure 6 we note that the Tessera achieves both the highest precision and f-score among the three approaches across all t values for the cyber threat dataset. In general, precision and f-score for Tessera exceeds that of the hierarchical approach by 0.15 and 0.13. Though our graph-based benchmark achieves the highest recall, its precision value is $10x$ smaller than the other two approaches, implying that the graph-based approach is overwhelmed when presented with a very large log, returning many false positives. The Tessera recall value is comparable and achieves a better balance between precision and recall, as indicated by the f-score.

Figure 6 illustrates performance for our open-ended exploratory analytics dataset. We note that Tessera outperforms both other approaches in precision, recall and f-score. On average, the precision and f-score of Tessera outperformed the hierarchical benchmark by 0.27 and 0.23. Unlike that of the cyber threat dataset, for our dataset the graph-based approach didn't achieve the highest recall. Instead, it achieves better precision and f-score. This also implies that the graph-based approach scales better when given a smaller amount of log events. However, across different sizes of log events and tools, Tessera scales better. We will revisit this point in the compression efficiency section below. Another interesting discovery here is that the average observed error latency Δt is good enough for all three approaches to achieve reasonable performance. Thus, our findings suggest that performance in Tessera with respect to **Q1** is good.

4.3.2 Sensitivity of time window. In order to better understand how the amount that Tessera looks back into logs affects performance and efficiency (**Q2** in our evaluation plan), we varied its time window, K. The time window was motivated by the common need for users to revisit past interface states (e.g. in PoleStar user might bookmark previous interesting discoveries and revisited the results some time later in the following exploration). A window allows for comparisons between the current state, previous states, and later states. However, larger windows come with potential costs in terms of efficiency. Our goal was to identify if and when the model performance converges towards an asymptotic "sweet spot". Convergence for a smaller value of K would indicate that there is a "sweet spot" which balances between the risk of large time windows being overly inefficient and small time windows neglecting to detect revisiting and connected states. We examined Tessera's f-score under different K values when observed latency error $\Delta t = 10s$ for Adult dataset and $\Delta t = 60s$ for the cyber threat dataset. Figure 7 shows that for both datasets, increasing K results in better performance. Further, the performance appears to reach an asymptote relatively quickly, and at a size for K that is computationally tractable. Tessera might not need a very large K to obtain a good f-score, however it is possible that K is dependent on user- and task-level differences (e.g. some tasks require more revisiting than others) that we have not yet observed.

One goal of interaction sequence analysis and user behavior mining is to simplify event log content. Aggregating similar states can

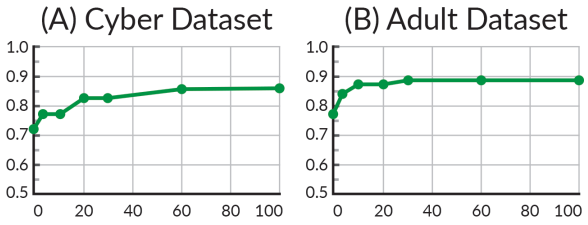


Figure 7: Sensitivity of K for pair-wise comparisons regarding Tessera's performance

make analyzing results easier and lead to better model generalizability [18]. In order to gauge how well Tessera and our benchmarks simplify data, we measure the compression efficiency of the resulting models, $\frac{\#derived\ segments}{\#original\ log\ events}$. The lower the compression ratio, the simpler the end representation. As a result we used a fixed $K = 20$ for Tessera in the rest of the experimental evaluation.

4.3.3 Measuring compression efficiency. In **Q3** of our evaluation agenda, we considered whether one approach would provide better compression – an indicator that it was successfully eliminating redundant events. Figure 8 shows the compression ratio for the three approaches. Tessera outperforms the benchmarks across both datasets, though the hierarchical model outputs a comparable compression ratio. Note that we introduce a scale break in Figure 8a, as the graph-based approach performed approximately 100x worse in terms of compression for the first dataset. This aligns with some of our earlier findings regarding the approach's inability to handle the large stream of events from a long session. One explanation for this is that the graph-based approach was over-fit for singular scenarios and not for an entire analysis log. Figure 8(B) shows that all three approaches achieve more comparable compression rates for our dataset which has fewer log events, however Tessera outperforms the other approaches by 0.23.

4.3.4 Qualitative reports. We collected qualitative data in the form of participant reports during think-alouds as well as in our semi-structured post-interviews in order to understand in depth how users conducted their evaluation (**Q4**). As our participant interview pool was limited, we did not code results or apply a formal qualitative research protocol. Instead, in this section we outline some of the general trends and themes we observed among participant responses which reflect on some of the quantitative findings reported earlier in this section.

Participants, in general, did show evidence of looping, sensemaking behavior [50], as reflected in schematic form in Figure 9. Though

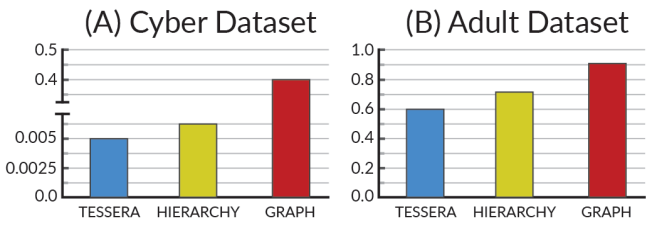


Figure 8: Compression rates for Tessera and benchmarks

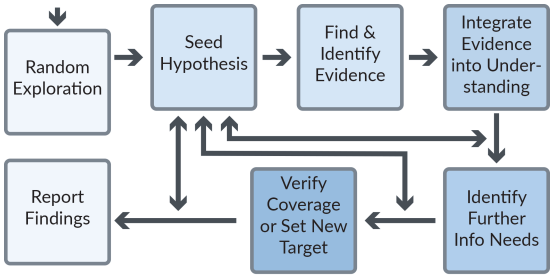


Figure 9: Schematic view of a common analysis path for participants examining the Adult dataset.

participants did not report that their analysis process involved a formal series of subtasks, they did report that they switched between evidence-seeking and hypothesis generation. "I was trying to find the relationship between income and gender, and moved to another pair after seeing something". Participants followed leads and iterated on findings. Examining event logs, participants revisited previous states frequently, providing further evidence of looping behavior. "I want to see whether the relationship between salary-level is consistent with their degree levels". This points to one potential application of our segmentation strategy - helping participants reflect on individual hypotheses or segments of activity as they loop through multiple iterations or tracking participant progress through the sensemaking process.

One other theme that emerged in our qualitative responses was a tension between open-ended exploration and goal-directed exploration. While participants reported that they felt free to follow whatever investigative leads they wished during our lab study sessions, practically speaking the data that we received were not all that different from that of the cyber threat database other than differences in session duration. One possible explanation for this is that, while participants had an open-ended tool, participants still felt that they had to achieve a specific goal and directed their activities accordingly in a lab study environment with an assigned task (no matter how openly phrased).

We also noted evidence of bias being encoded into analysts' exploration process. Aligning with a growing body of literature on cognitive bias [60], we found evidence that participants fell victim to confirmation bias as they explored. Revisiting of previous states and similarity between individual interface activities suggests that as participants iterated, they often followed the same paths as they had previously. On a high level, these activities risk leaking into the structure of a simplified representation of the analysis behavior. Revisiting and similar events can amplify parts of the simplified task structure, potentially boosting parts of the analysis that were subject to bias. If these models are then used for downstream applications, they might risk encoding the same bias into future analyses.

5 DISCUSSION

Through our evaluation we found that Tessera outperformed benchmark approaches at task-level shifting detection. This is a promising initial signal that Tessera could be used for mining analyst interaction event logs for higher order information about the analysis process and the analyst engaging in it. Because we compute similarity scores across segments of activity, we can potentially derive

additional signals from the scores themselves. For example, by sorting in terms of how much the scores contributed to the final similarity score, we can identify the particular stages of the analysis which contributed the most and the least information to the overall process. Similarly, if we notice a lack of divergence of scores, then this could indicate that the analyst is fixated on a smaller portion of the data and may potentially be experiencing bias. Beyond similarity scores, there is a possibility that the segments and our summary metrics might be of use for providing real-time feedback and retrospective provenance information.

However, there are several potential limitations both in the implementation of Tessera and in our evaluation. One theme throughout this paper has been the question of scalability across long sessions of analysis. As longer sessions are likely ones where provenance and real-time feedback would be of most value to the analyst, scale is an important issue to consider. When discussing sensitivity, we noted that the pairwise comparison of log points will lead to potentially $O(k^2)$ cost for comparisons where is k is the number of log records within the time window. This lead to the necessary compromise between look-back ability and efficiency. While applying efficient use of data structures may reduce the average cost to $O(k * log(k))$, it remains potentially intractable for longer data from longer sessions. On the other hand, k can be intentionally reduced as a trade-off between efficiency and accuracy if real-time, streaming feedback is necessary and extended for higher accuracy if more computation time is available. Additionally, other approaches suffer similar scale inefficiencies and trade-offs for system designers.

Tessera may also not be taking advantage of all possible signals available to it. For instance, one might compute more complex metrics about steps of an exploration (e.g. causal chaining) which deliver more nuanced splits between segments. We might also integrate Tessera into other approaches (for example the approaches proposed by [37] and [18]) to garner a more comprehensive task-level and sub-task understanding of an EVA session.

Additionally, there are potential threats to generalizability in our experimental method. We note that the two datasets used in our think-aloud study are not identical, varying in size and complexity. This lead to differences in events recorded for processing by Tessera. The size of the behavioral dataset we gathered for Adult is smaller than the Cyber Dataset. Besides differences in source data, one of the main reasons our new think-aloud Adult dataset is much smaller than cyber threat is because we did not create mouse movement events during our logging. The cyber dataset contains a large portion of mouse movement activities which was documented by its platform, though not used in our experiments. In addition to the feature collection difference, participants in our study (mimicking the time setup from existing work [64]) had less time to work than that of the cyber dataset. We think both datasets represent two different-but-commonly-encountered data analysis scenarios. While we believe these dataset differences did not result in any noticeable influence on our study outcomes, in the future we hope to explore how different logging and interaction affordances shape the overall efficacy of log processing tools like Tessera.

Our evaluation focused on two different datasets gathered using a custom interface and PoleStar. In both cases, the interfaces were relatively rudimentary and may not accurately reflect the interaction affordances of an industry tool such as Tableau. We accepted this compromise as it reduced the training burden on participants and made logs easier to pair with think-aloud data accurately. In the case of a more sophisticated tool, the variety of events being triggered and additional kinds of data manipulations may stretch the capabilities of Tessera. Much like in our hierarchical benchmark, there may be a need for adaptation to system- or task-specific features. These systems might also throw more events per minute of activity, which could lead to aforementioned scalability issues. In terms of the datasets used for testing Tessera, there is the possibility that both interfaces forced participants into patterns of use which naturally comport better with Tessera than the other techniques evaluated. There is a need to evaluate Tessera with a wider variety of systems, tasks, datasets, and analysts. In particular, the question of whether goal-direction in EVA affects our results remains somewhat unresolved. While we intentionally designed our dataset task to be more open-ended, we noted that qualitative responses may be indicating a degree of goal-direction as a result of the study design. Increasing the variety of data processing tasks may help to resolve this question.

As our think-aloud data encodes information about user intentions and their current place in a sensemaking loop, we plan to add additional labels to our dataset to determine whether Tessera features are predictive of cognitive factors. For instance, can we label segments that correspond to "hypothesis verification" or "drill-down" actions? If possible, this would enable better summarization of logs and could lead to a variety of feedback mechanisms ranging from automated presentations (i.e. activity labels and summaries of actions), progress visualizations (i.e. workflow dashboards), and real-time diagnostics for factors such as bias.

There are a number of potential applications of Tessera within the EVA ecosystem. Foremost, because Tessera extracts a number of features for each segment (e.g. similarity between segments), one might build classifiers that provide more detail about what is going on in an exploration. It is also possible that we can apply Tessera to other domains where users' behavioral features are available. For example, we could apply Tessera in detecting crowd workers' behavioral paths in solving tasks in aggregate, and finding patterns among image editors [40].

6 CONCLUSION

In this paper we described Tessera, a technique for identifying segments of goal-directed activity from interface event logs of data analysts engaging in EVA. We described an implementation of Tessera and conducted an initial evaluation against two benchmark algorithms (hierarchical- and graph-based techniques) through a public dataset of goal-directed analyst behavior. We then developed a methodology for gathering additional paired think-aloud and event log data for open-ended exploration, and conducted twenty participant sessions to assemble a dataset. Using this dataset, we compared model performance, finding that Tessera successfully identified task switching by participants and outperformed benchmarks. We hope to build on this work in the future by classifying the resulting segments to characterize patterns of activity and incorporating Tessera into existing EVA systems as an additional method of giving analysts feedback.

ACKNOWLEDGMENTS

This work was supported by NSF grant IIS-1850195. We would like to thank the associate chairs and anonymous reviewers for their invaluable feedback. We also would like to thank researchers and engineers of Dataprep library team from Database System Lab at Simon Fraser University for the valuable input of mapping of common functions and user intentions. All opinions, findings, and conclusions in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

[1] Swarup Acharya, Phillip B. Gibbons, Viswanath Poosala, and Sridhar Ramaswamy. 1999. The Aqua Approximate Query Answering System. In *SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadelphia, Pennsylvania, USA*, Alex Delis, Christos Faloutsos, and Shahram Ghandeharizadeh (Eds.). ACM Press, 574–576. https://doi.org/10.1145/304182.304581

[2] Sameer Agarwal, Barzan Mozafari, Aurojit Panda, Henry Milner, Samuel Madden, and Ion Stoica. 2013. BlinkDB: queries with bounded errors and bounded response times on very large data. In *Eighth Eurosys Conference 2013, EuroSys '13, Prague, Czech Republic, April 14-17, 2013*, Zdenek Hanzálek, Hermann Härtig, Miguel Castro, and M. Frans Kaashoek (Eds.). ACM, 29–42. https://doi.org/10.1145/2465351.2465355

[3] Sara Alspaugh, Nava Zokaei, Andrea Liu, Cindy Jin, and Marti A Hearst. 2018. Futzing and moseying: Interviews with professional data analysts on exploration practices. *IEEE transactions on visualization and computer graphics* 25, 1 (2018), 22–31.

[4] David C. Anastasiu, Jeremy Iverson, Shaden Smith, and George Karypis. 2014. Big Data Frequent Pattern Mining. In *Frequent Pattern Mining*, Charu C. Aggarwal and Jiawei Han (Eds.). Springer, 225–259. https://doi.org/10.1007/978-3-319-07821-2_10

[5] John Annett and Neville Anthony Stanton. 2000. *Task analysis*. CRC Press.

[6] Mamoun A Awad and Issa Khalil. 2012. Prediction of user's web-browsing behavior: Application of markov model. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 42, 4 (2012), 1131–1142.

[7] Solon Barocas and Andrew D Selbst. 2016. Big data's disparate impact. *Calif. L. Rev.* 104 (2016), 671.

[8] Leilani Battle, Remco Chang, and Michael Stonebraker. 2016. Dynamic Prefetching of Data Tiles for Interactive Visualization. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, Fatma Özcan, Georgia Koutrika, and Sam Madden (Eds.). ACM, 1363–1375. https://doi.org/10.1145/2882903.2882919

[9] Leilani Battle and Jeffrey Heer. 2019. Characterizing Exploratory Visual Analysis: A Literature Review and Evaluation of Analytic Provenance in Tableau. *Comput. Graph. Forum* 38, 3 (2019), 145–159. https://doi.org/10.1111/cgf.13678

[10] Tanja Blascheck, Markus John, Kuno Kurzhals, Steffen Koch, and Thomas Ertl. 2015. VA 2: a visual analytics approach for evaluating visual analytics applications. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 61–70.

[11] Tanja Blascheck, Kuno Kurzhals, Michael Raschke, Michael Burch, Daniel Weiskopf, and Thomas Ertl. 2017. Visualization of Eye Tracking Data: A Taxonomy and Survey. *Comput. Graph. Forum* 36, 8 (2017), 260–284. https://doi.org/10.1111/cgf.13079

[12] Christian Bors, John Wenskovitch, Michelle Dowling, Simon Attfield, Leilani Battle, Alex Endert, Olga Kulyk, and Robert S Laramee. 2019. A provenance task abstraction framework. *IEEE computer graphics and applications* 39, 6 (2019), 46–60.

[13] Eli T Brown, Alvitta Ottley, Helen Zhao, Quan Lin, Richard Souvenir, Alex Endert, and Remco Chang. 2014. Finding waldo: Learning about users from their interactions. *IEEE Transactions on visualization and computer graphics* 20, 12 (2014), 1663–1672.

[14] John Brown. 1958. Some tests of the decay theory of immediate memory. *Quarterly Journal of Experimental Psychology* 10, 1 (1958), 12–21.

[15] George Casella and Roger L Berger. 2002. *Statistical inference*. Vol. 2. Duxbury Pacific Grove, CA.

[16] Surajit Chaudhuri, Gautam Das, and Vivek Narasayya. 2007. Optimized stratified sampling for approximate query processing. *ACM Transactions on Database Systems (TODS)* 32, 2 (2007), 9–es.

[17] Kristin A. Cook, Nick Cramer, David J. Israel, Michael Wolverton, Joe Bruce, Russ Burtner, and Alex Endert. 2015. Mixed-initiative visual analytics using task-driven recommendations. In *10th IEEE Conference on Visual Analytics Science and Technology, IEEE VAST 2015, Chicago, IL, USA, October 25-30, 2015*, Min Chen and Gennady L. Andrienko (Eds.). IEEE Computer Society, 9–16. https://doi.org/10.1109/VAST.2015.7347625

[18] Filip Dabek and Jesus J Caban. 2016. A grammar-based approach for modeling user interactions and generating suggestions during the data exploration process. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 41–50.

[19] Kyriaki Dimitriadou, Olga Papaemmanouil, and Yanlei Diao. 2014. Explore-by-example: an automatic query steering framework for interactive data exploration. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, Curtis E. Dyreson, Feifei Li, and M. Tamer Özsu (Eds.). ACM, 517–528. https://doi.org/10.1145/2588555.2610523

[20] Cody Dunne, Nathalie Henry Riche, Bongshin Lee, Ronald A. Metoyer, and George G. Robertson. 2012. GraphTrail: analyzing large multivariate, heterogeneous networks while supporting exploration history. In *CHI Conference on Human Factors in Computing Systems, CHI '12, Austin, TX, USA - May 05 - 10, 2012*, Joseph A. Konstan, Ed H. Chi, and Kristina Höök (Eds.). ACM, 1663–1672. https://doi.org/10.1145/2207676.2208293

[21] Omar ElTayeby and Wenwen Dou. 2016. A Survey on Interaction Log Analysis for Evaluating Exploratory Visualizations. In *Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization, BELIV 2016, Baltimore, MD, USA, October 24, 2016*, Michael Sedlmair, Petra Isenberg, Tobias Isenberg, Narges Mahyar, and Heidi Lam (Eds.). ACM, 62–69. https://doi.org/10.1145/2993901.2993912

[22] Alex Endert, Patrick Fiaux, and Chris North. 2012. Semantic interaction for sensemaking: inferring analytical reasoning for model steering. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2879–2888.

[23] Alex Endert, W. Ribarsky, Cagatay Turkay, B. L. William Wong, Ian T. Nabney, Ignacio Díaz Blanco, and Fabrice Rossi. 2017. The State of the Art in Integrating Machine Learning into Visual Analytics. *Comput. Graph. Forum* 36, 8 (2017), 458–486. https://doi.org/10.1111/cgf.13092

[24] Danyel Fisher, Igor O. Popov, Steven Mark Drucker, and m. c. schraefel. 2012. Trust me, i'm partially right: incremental visualization lets analysts explore large datasets faster. In *CHI Conference on Human Factors in Computing Systems, CHI '12, Austin, TX, USA - May 05 - 10, 2012*, Joseph A. Konstan, Ed H. Chi, and Kristina Höök (Eds.). ACM, 1673–1682. https://doi.org/10.1145/2207676.2208294

[25] Alex Galakatos, Andrew Crotty, Emanuel Zgraggen, Carsten Binnig, and Tim Kraska. 2017. Revisiting reuse for approximate query processing. *Proceedings of the VLDB Endowment* 10, 10 (2017), 1142–1153.

[26] David Gotz and Zhen Wen. 2009. Behavior-driven visualization recommendation. In *Proceedings of the 14th International Conference on Intelligent User Interfaces, IUI 2009, Sanibel Island, Florida, USA, February 8-11, 2009*, Cristina Conati, Mathias Bauer, Nuria Oliver, and Daniel S. Weld (Eds.). ACM, 315–324. https://doi.org/10.1145/1502650.1502695

[27] David Gotz and Michelle X Zhou. 2009. Characterizing users' visual analytic activity for insight provenance. *Information Visualization* 8, 1 (2009), 42–55.

[28] Lars Grammel, Melanie Tory, and Margaret-Anne Storey. 2010. How information visualization novices construct visualizations. *IEEE transactions on visualization and computer graphics* 16, 6 (2010), 943–952.

[29] Georges G. Grinstein, Jean Scholtz, Mark A. Whiting, and Catherine Plaisant. 2009. VAST 2009 challenge: An insider threat. In *4th IEEE Symposium on Visual Analytics Science and Technology, IEEE VAST 2009, Atlantic City, NJ, USA, October 11-16, 2009, part of VisWeek 2009*. IEEE Computer Society, 243–244. https://doi.org/10.1109/VAST.2009.5334454

[30] Adrien Guille, Hakim Hacid, Cecile Favre, and Djamel A Zighed. 2013. Information diffusion in online social networks: A survey. *ACM Sigmod Record* 42, 2 (2013), 17–28.

[31] Hua Guo, Steven R Gomez, Caroline Ziemkiewicz, and David H Laidlaw. 2015. A case study using visualization interaction logs and insight metrics to understand how analysts arrive at insights. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 51–60.

[32] Jeffrey Heer, Jock Mackinlay, Chris Stolte, and Maneesh Agrawala. 2008. Graphical histories for visualization: Supporting analysis, communication, and evaluation. *IEEE transactions on visualization and computer graphics* 14, 6 (2008), 1189–1196.

[33] Jeffrey Heer and Ben Shneiderman. 2012. Interactive dynamics for visual analysis. *Queue* 10, 2 (2012), 30–55.

[34] Petra Isenberg, Anthony Tang, and Sheelagh Carpendale. 2008. An exploratory study of visual information analysis. In *Proceedings of the 2008 Conference on Human Factors in Computing Systems, CHI 2008, 2008, Florence, Italy, April 5-10, 2008*, Mary Czerwinski, Arnold M. Lund, and Desney S. Tan (Eds.). ACM, 1217–1226. https://doi.org/10.1145/1357054.1357245

[35] Renáta Iváncsy and István Vajk. 2006. Frequent pattern mining in web log data. *Acta Polytechnica Hungarica* 3, 1 (2006), 77–90.

[36] Paul Jaccard. 1912. The distribution of the flora in the alpine zone. 1. *New phytologist* 11, 2 (1912), 37–50.

[37] Heidi Lam, Melanie Tory, and Tamara Munzner. 2017. Bridging from goals to tasks with design study analysis reports. *IEEE transactions on visualization and computer graphics* 24, 1 (2017), 435–445.

[38] Juan Liu, Aaron Wilson, and David Gunning. 2014. Workflow-based human-in-the-loop data analytics. In *Proceedings of the 2014 Workshop on Human Centered Big Data Research*. book, 49–52.

[39] Zhicheng Liu and Jeffrey Heer. 2014. The effects of interactive latency on exploratory visual analysis. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 2122–2131.

[40] Zipeng Liu, Zhicheng Liu, and Tamara Munzner. 2020. Data-driven Multi-level Segmentation of Image Editing Logs. In *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*, Regina Bernhaupt, Florian 'Floyd' Mueller, David Verweij, Josh Andres, Joanna McGrenere, Andy Cockburn, Ignacio Avellino, Alix Goguey, Pernille Bjøn, Shengdong Zhao, Briane Paul Samson, and Rafal Kocielnik (Eds.). ACM, 1–12. https://doi.org/10.1145/3313831.3376152

[41] Yuyu Luo, Chengliang Chai, Xuedi Qin, Nan Tang, and Guoliang Li. 2020. Interactive Cleaning for Progressive Visualization through Composite Questions. In *36th IEEE International Conference on Data Engineering, ICDE 2020, Dallas, TX, USA, April 20-24, 2020*. IEEE, 733–744. https://doi.org/10.1109/ICDE48307.2020.00069

[42] Eren Manavoglu, Dmitry Pavlov, and C. Lee Giles. 2003. Probabilistic User Behavior Models. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003), 19-22 December 2003, Melbourne, Florida, USA*. IEEE Computer Society, 203–210. https://doi.org/10.1109/ICDM.2003.1250921

[43] Sina Mohseni, Andrew Pachuilo, Ehsanul Haque Nirjhar, Rhema Linder, Alyssa M. Pena, and Eric D. Ragan. 2018. Analytic Provenance Datasets: A Data Repository of Human Analysis Activity and Interaction Logs. *CoRR* abs/1801.05076 (2018). arXiv:1801.05076 http://arxiv.org/abs/1801.05076

[44] Douglas C Montgomery and George C Runger. 2014. *Applied statistics and probability for engineers*. John Wiley and Sons.

[45] Arnab Nandi, Alan Fekete, and Carsten Binnig. 2016. HILDA 2016 Workshop: A Report. *IEEE Data Eng. Bull.* 39, 4 (2016), 85–86.

[46] Janni Nielsen, Torkil Clemmensen, and Carsten Yssing. 2002. Getting access to what goes on in people's heads?: reflections on the think-aloud technique. In *Proceedings of the Second Nordic Conference on Human-Computer Interaction 2002, Aarhus, Denmark, October 19-23, 2002*, Olav W. Bertelsen (Ed.). ACM, 101–110. https://doi.org/10.1145/572020.572033

[47] Yongjoo Park, Ahmad Shahab Tajik, Michael J. Cafarella, and Barzan Mozafari. 2017. Database Learning: Toward a Database that Becomes Smarter Every Time. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*, Semih Salihoglu, Wenchao Zhou, Rada Chirkova, Jun Yang, and Dan Suciu (Eds.). ACM, 587–602. https://doi.org/10.1145/3035918.3064013

[48] Jinglin Peng, Dongxiang Zhang, Jiannan Wang, and Jian Pei. 2018. AQP++: Connecting Approximate Query Processing With Aggregate Precomputation for Interactive Analytics. In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, Gautam Das, Christopher M. Jermaine, and Philip A. Bernstein (Eds.). ACM, 1477–1492. https://doi.org/10.1145/3183713.3183747

[49] Adam Perer and Ben Shneiderman. 2008. Systematic yet flexible discovery: guiding domain experts through exploratory data analysis. In *Proceedings of the 13th International Conference on Intelligent User Interfaces, IUI 2008, Gran Canaria, Canary Islands, Spain, January 13-16, 2008*, Jeffrey M. Bradshaw, Henry Lieberman, and Steffen Staab (Eds.). ACM, 109–118. https://doi.org/10.1145/1378773.1378788

[50] Peter Pirolli and Stuart Card. 2005. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of international conference on intelligence analysis*, Vol. 5. McLean, VA, USA, 2–4.

[51] Eric D Ragan, Alex Endert, Jibonananda Sanyal, and Jian Chen. 2015. Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 31–40.

[52] Timothy J Ricker, Evie Vergauwe, and Nelson Cowan. 2016. Decay theory of immediate memory: From Brown (1958) to today (2014). *Quarterly Journal of Experimental Psychology* 69, 10 (2016), 1969–1995.

[53] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat, and Jeffrey Heer. 2017. Vega-Lite: A Grammar of Interactive Graphics. *IEEE Trans. Vis. Comput. Graph.* 23, 1 (2017), 341–350. https://doi.org/10.1109/TVCG.2016.2599030

[54] Richard C Sprinthall and Stephen T Fisk. 1990. *Basic statistical analysis*. Prentice Hall Englewood Cliffs, NJ.

[55] Neville A Stanton. 2006. Hierarchical task analysis: Developments, applications, and extensions. *Applied ergonomics* 37, 1 (2006), 55–79.

[56] Guo-Dao Sun, Ying-Cai Wu, Rong-Hua Liang, and Shi-Xia Liu. 2013. A survey of visual analytics techniques and applications: State-of-the-art research and future challenges. *Journal of Computer Science and Technology* 28, 5 (2013), 852–867.

[57] Taffee T Tanimoto. 1958. Elementary mathematical theory of classification and prediction. (1958).

[58] Dataprep Team. 2020. Dataprep: Data Preparation in Python. http://dataprep.ai.

[59] John W Tukey. 1977. *Exploratory data analysis*. Vol. 2. Reading, Mass.

[60] Emily Wall, Leslie M. Blaha, Lyndsey Franklin, and Alex Endert. 2017. Warning, Bias May Occur: A Proposed Approach to Detecting Cognitive Bias in Interactive Visual Analytics. In *12th IEEE Conference on Visual Analytics Science and Technology, IEEE VAST 2017, Phoenix, AZ, USA, October 3-6, 2017*, Brian Fisher, Shixia Liu, and Tobias Schreck (Eds.). IEEE Computer Society, 104–115. https://doi.org/10.1109/VAST.2017.8585669

[61] Kanit Wongsuphasawat, Yang Liu, and Jeffrey Heer. 2019. Goals, Process, and Challenges of Exploratory Data Analysis: An Interview Study. *arXiv preprint arXiv:1911.00568* (2019).

[62] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2016. Towards a general-purpose query language for visualization recommendation. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*. 1–6.

[63] Piotr A Woźniak, Edward J Gorzelańczyk, and Janusz A Murakowski. 1995. Two components of long-term memory. *Acta neurobiologiae experimentalis* 55, 4 (1995), 301–305.

[64] Jing Nathan Yan, Ziwei Gu, Hubert Lin, and Jeffrey M. Rzeszotarski. 2020. Silva: Interactively Assessing Machine Learning Fairness Using Causality. In *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*, Regina Bernhaupt, Florian 'Floyd' Mueller, David Verweij, Josh Andres, Joanna McGrenere, Andy Cockburn, Ignacio Avellino, Alix Goguey, Pernille Bjøn, Shengdong Zhao, Briane Paul Samson, and Rafal Kocielnik (Eds.). ACM, 1–13. https://doi.org/10.1145/3313831.3376447

[65] Jaewon Yang and Jure Leskovec. 2010. Modeling Information Diffusion in Implicit Networks. In *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, Geoffrey I. Webb, Bing Liu, Chengqi Zhang, Dimitrios Gunopulos, and Xindong Wu (Eds.). IEEE Computer Society, 599–608. https://doi.org/10.1109/ICDM.2010.22

[66] Reza Zafarani, Mohammad Ali Abbasi, and Huan Liu. 2014. *Social media mining: an introduction*. Cambridge University Press.

[67] Richard S. Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. 2013. Learning Fair Representations. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013 (JMLR Workshop and Conference Proceedings, Vol. 28)*. JMLR.org, 325–333. http://proceedings.mlr.press/v28/zemel13.html

[68] Emanuel Zgraggen, Alex Galakatos, Andrew Crotty, Jean-Daniel Fekete, and Tim Kraska. 2016. How progressive visualizations affect exploratory analysis. *IEEE transactions on visualization and computer graphics* 23, 8 (2016), 1977–1987.

[69] Emanuel Zgraggen, Zheguang Zhao, Robert C. Zeleznik, and Tim Kraska. 2018. Investigating the Effect of the Multiple Comparisons Problem in Visual Analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, Montreal, QC, Canada, April 21-26, 2018*, Regan L. Mandryk, Mark Hancock, Mark Perry, and Anna L. Cox (Eds.). ACM, 479. https://doi.org/10.1145/3173574.3174053

[70] Zheguang Zhao, Emanuel Zgraggen, Lorenzo De Stefani, Carsten Binnig, Eli Upfal, and Tim Kraska. 2017. Safe Visual Data Exploration. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*, Semih Salihoglu, Wenchao Zhou, Rada Chirkova, Jun Yang, and Dan Suciu (Eds.). ACM, 1671–1674. https://doi.org/10.1145/3035918.3058749