Mixed-Initiative Visual Analytics Using Task-Driven Recommendations

Kristin Cook, Nick Cramer, David Israel, Michael Wolverton, Joe Bruce, Russ Burtner, and Alex Endert

Abstract—Visual data analysis is composed of a collection of cognitive actions and tasks to decompose, internalize, and recombine data to produce knowledge and insight. Visual analytic tools provide interactive visual interfaces to data to support discovery and sensemaking tasks, including forming hypotheses, asking questions, and evaluating and organizing evidence. Myriad analytic models can be incorporated into visual analytic systems at the cost of increasing complexity in the analytic discourse between user and system. Techniques exist to increase the usability of interacting with analytic models, such as inferring data models from user interactions to steer the underlying models of the system via semantic interaction, shielding users from having to do so explicitly. Such approaches are often also referred to as mixed-initiative systems. Sensemaking researchers have called for development of tools that facilitate analytic sensemaking through a combination of human and automated activities. However, design guidelines do not exist for mixed-initiative visual analytic systems to support iterative sensemaking. In this paper, we present candidate design guidelines and introduce the Active Data Environment (ADE) prototype, a spatial workspace supporting the analytic process via task recommendations invoked by inferences about user interactions within the workspace. ADE recommends data and relationships based on a task model, enabling users to co-reason with the system about their data in a single, spatial workspace. This paper provides an illustrative use case, a technical description of ADE, and a discussion of the strengths and limitations of the approach.

Index Terms—mixed-initiative visual analytics, task modeling, recommender systems, sensemaking

1 Introduction

Visual analytics facilitates discovery and analytical reasoning via interactive data visualizations [1]. By coupling data analysis techniques with interactive visualizations, visual analytics enable humans to reason about complex datasets. These technologies complement human cognition with analytic techniques, with the goal of producing insights. Visual data exploration tasks are often performed via series of interactions with different visual representations and across datasets of different media, size, and structure. Further, each of these datasets may require a specific technique or approach.

The complex cognitive process of data analysis has been broadly described by a notional model of sensemaking [2], consisting of two primary activities: information foraging and synthesis. During information foraging, analysts seek relevant data through a series of broadening and narrowing searches. During synthesis, analysts develop and revise conceptual models or hypotheses and fit the data to these models. The sensemaking process is iterative, with new foraging-synthesis iterations initiated to start new subtasks, address gaps identified during current subtasks, reduce uncertainty, or investigate anomalies, for example. These foraging-synthesis iterations are more likely to be ad hoc and idiosyncratic than systematic, with much of the variability stemming from the transitions between foraging-synthesis iterations [3]. The planning, execution, and re-combination of these subtasks into a coherent whole is a critical part of the analytic discourse [4].

- *Kristin A. Cook is with Pacific Northwest National Laboratory (PNNL). E-mail: Kris.Cook@pnnl.gov.*
- *Nick O. Cramer is with PNNL. E-mail: nick.cramer@pnnl.gov.*
- *David Israel is with SRI International. E-mail: david.israel@sri.com.*
- *Michael Wolverton is with SRI International. E-mail: michael.wolverton@sri.com.*
- *Joe Bruce is with PNNL. E-mail: joseph.bruce@pnnl.gov.*
- *Russ Burtner is with PNNL. E-mail: russ.burtner@pnnl.gov.*
- *Alex Endert is with Georgia Tech. E-mail:endert@gatech.edu.*

- *Manuscript received 31 Mar. 2015*

Visual analytics have adopted concepts of mixed-initiative systems [5] to balance sensemaking efforts between the human and the system. Such systems aim to offload more computationally appropriate tasks or actions onto the system, enabling humans to reason about concepts at a higher level. A key idea is that these systems take some form of initiative on behalf of the user, i.e., to balance the effort between user and system. Visual analytic systems exist where initiative is taken on behalf of the user by steering data analytic models [6], modeling the intent and cognitive characteristics of users [7], [8], and adjusting the structural representation of the data by defining the features or dimensionality of the data [9]. These systems are often enabled using techniques for inferring analytical reasoning from user interactions and coupling these inferences with some form of model steering or selection.

Kang and Stasko [10] and Zhang and Soergel [11] identify the need for software-based assistants to support the iterative sensemaking process. Kang and Stasko describe a hypothetical tool that permits users to move flexibly among their conceptual model, foraging, and synthesis. We assert that a mixed-initiative system is appropriate to address this need.

However, we posit that an open challenge is in understanding how to design and implement such a mixed-initiative visual analytics system that computationally complements analysts in the iterative process of foraging and synthesis. To our knowledge, no design guidelines for such systems have been described.

This paper contributes a set of design guidelines and a corresponding prototype mixed-initiative system that aims to support sensemaking by automatically foraging and recommending relevant data for visual data exploration based on the current state of the sensemaking activity, in essence assisting with the transition between one foraging-synthesis iteration and the next. Our prototype, the Active Data Environment, or ADE, incorporates diverse data into a single spatial visualization where users can organize information. Similar in design to prior spatial workspaces (e.g., [12], [13]), such visualizations enable users to externalize their incremental analytic artifacts during sensemaking [14], [15]. ADE provides a dynamic "thinking space" [16] in which the user develops and refines her conceptual model, and in which relevant data can be recommended for her consideration.

ADE attempts to infer the analytic task from the user's interactions in this workspace and invoke task models when appropriate. These models recommend relevant information and relationships or connections. A recommendation is composed of the relevant data, a natural-language explanation for the recommendation, and a numeric relevance measure. ADE presents recommendations to the user in the context of her analytic thinking environment. We refer to this information as "active data." With no need for the user to formalize a query, the data finds the user.

In this paper, we outline related work and present a candidate set of design guidelines for mixed-initiative iterative sensemaking environments. We describe a usage scenario for ADE and discuss the major components of ADE in detail. Finally, we present a discussion of the strengths and limitations of our approach and conclude with a description of future work.

2 Related Work

We discuss related work in three areas: sensemaking with spatial workspaces, mixed-initiative visual analytics, and task modeling and recommender systems.

2.1 Supporting Sensemaking with Spatial Workspaces

The cognitive activity of gaining understanding about the world through the analysis of data, commonly called sensemaking, has been widely studied. Pirelli and Card depict the cognitive stages of sensemaking in the *sensemaking loop* [2], which emphasizes the importance of synthesizing as well as foraging for data and information to gain insight. This is a complex, iterative process, entailing generation and testing of hypotheses as well as lower-level data filtering and retrieval tasks. Sensemaking involves internalizing and understanding information in the context of the analyst's experiences and prior knowledge, ultimately constructing knowledge through the combination of new knowledge artifacts found in the data and existing knowledge structure [2]. The fluidity of these tasks was emphasized by Kang and Stasko [10].

Visual analytic systems have been developed to support sensemaking. An early example is the Entity Workspace [17], which used an evidence file metaphor. One common visual metaphor to support sensemaking is a spatial workspace, or canvas. Andrews et al. found that providing analysts with the ability to manually organize information in a large spatial workspace enabled them to extend their working memory for a sensemaking task [14]. They found that analysts created spatial constructs that represented knowledge artifacts (timelines, lists, piles, etc.) corresponding to intermediate findings throughout the process. Shipman and Marshall refer to this process of refining intermediate knowledge structures spatially over time as incremental formalism [15]. They discuss how the lack of formality involved in creating spatial constructs provides advantages for sensemaking. For example, analysts can create piles or lists without specifying the parameters used to create them. For text analysis in particular, these spatial constructs have been shown to encode significant semantic information about an analyst's process and insights [18]. Visual analytic applications have been built to enable users to manually create spatial data layouts that aid their analytical reasoning (e.g., [12], [13], [19]).

2.2 Mixed-Initiative Visual Analytics

The concept of mixed-initiative systems is integral to many visual analytic applications. The key feature of such systems is the ability to "take initiative" or perform some actions on behalf of the user [5]. Equally important is to ensure human control or action. Thus, the goal of mixed-initiative systems is to properly balance the action and work between the user and system. The need to explore the appropriateness of this balance has been explored by Endert et al., who argue for "human-is-the-loop" techniques [20]. Such techniques adhere closely to the sensemaking processes described earlier and contrast with more traditional *human-in-the-loop* guidelines. Specifically, the distinction is made that domain experts should not be used to explicitly tune parameters of collections of analytic models. Instead, their cognitive abilities can be better leveraged by systems that allow users to engage in higher-level analytical reasoning, while systems learn from their actions to steer lower-level computational models [6].

One example illustrating this concept is ForceSPIRE, a visual analytic system that enables both computational and user-driven spatial organization of text documents [9]. User interactions directly in the visual metaphor are interpreted to steer the underlying layout algorithm. Similarly, the ability for computational models and users to co-create such spatial metaphors has been shown in applications including Dis-function [8] and StarSPIRE [21].

These examples are grounded in the understanding that analytical reasoning is encoded in users' interactions during sensemaking and exploration, providing a means of capturing analytic provenance [6]. Examples of such semantic interaction interfaces build on this work by directly binding the inference of the user interaction with specific steering approaches for analytical models in the system [6].

The work presented in this paper explores how interactions can be interpreted for task recommendations, as distinguished from previous work, which is focused largely on user modeling and data modeling.

2.3 Task Modeling and Recommender Systems

Unlike information retrieval systems, which search collections of largely unstructured data to return information that meets criteria explicitly specified by the user [22], recommender systems suggest relevant information that may be useful [23]. Broadly, there are two kinds of recommender systems: collaborative filtering systems [24] and content-based systems [25]. Collaborative filtering systems are often used when the system has access to and agglomerates data from many users, while content-based systems are preferred when the system does not have such access.

Collaborative filtering systems, popularly exemplified by Google News, Amazon, and Netflix, rely on large quantities of data from users to build user preference models, on the basis of which the system predicts or infers either a new item or a preference ordering (most-preferred item) over new items. The user data can be either explicit (user rankings) or implicit (gleaned from users' behavior).

Content-based systems address cases such as text [26] in which it is possible to perform some level of content analysis. Text content analysis is often based on keywords, although more complex analyses are sometimes deployed (e.g., by building per-document word-frequency indices). Again, one builds a model of user preferences and recommends new items to the user on the basis of the similarity to past favorites.

For ADE, neither of these approaches was appropriate, as our goal is not to recommend information based on others' behavior or interests. Rather, we strive to recommend data relevant to this analyst's specific task. This relies on making informed guesses as to what she is trying to do at any point, and where it is given that the activity will comprise a number of steps (or subtasks). ADE's recommender is partially a content-based recommender system [27], for it is one in which the content recommended is not based solely on the intrinsic content characteristics of items like similarity relations, but rather on the match between those content characteristics and the tasks the analyst is engaged in.

Prior work also explores how to show the recommendations to users. Toledo et al. [28] developed a recommender to support data analysis by displaying alternate views of a stacked graph using dwell time as a measure of user interest. Here, the recommendation is a view of the stacked graph. VisComplete [29] presented an approach for construction of visualization pipelines by recommending the remaining steps in a new pipeline based on a database of previously completed pipelines. These recommended steps are presented as suggestions displayed alongside the partial pipeline the user has created. Maguire et al. [30] developed a method for generating macros for workflow visualization; the recommended candidate

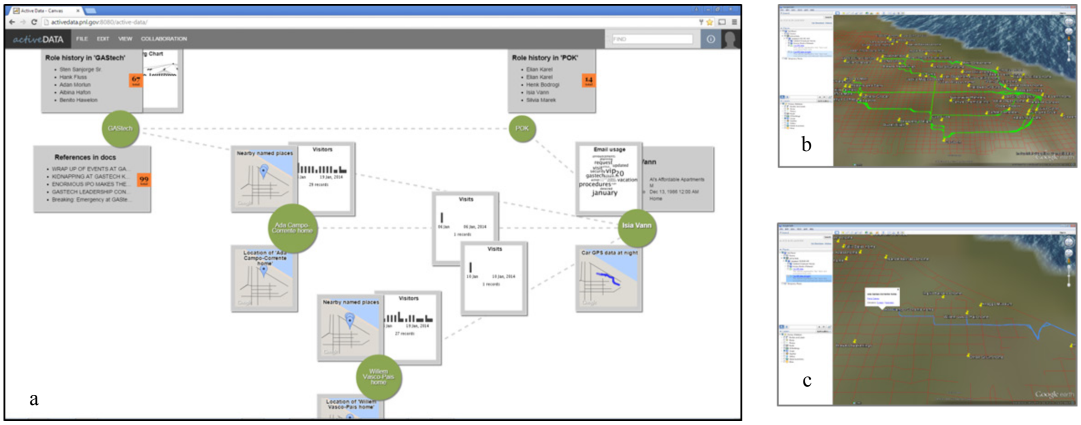

Figure 1. The ADE Canvas supports externalization of analytic thinking and displays recommendations in context of the ongoing analysis. a) In response to the user's entry of seeds, shown in green, ADE recommends data and relationships, shown in recommendation cards and dotted-line connections. This Canvas shows the state after the user has entered some seeds directly on the Canvas and has "pinned" other locations from Google Earth. b) Vehicle route recommendations for Isia Vann are displayed in Google Earth. Each geospatial recommendation from ADE appears in a separate layer. c) The user suppresses other layers to show vehicle travel at night, revealing suspicious visits to the homes of Willem Vasco-Pais and Ada Campo-Corrente. The user "pins to Canvas" to create new seeds for these locations.

macros are presented using glyphs to compactly summarize the relevant data. Our approach differs in that ADE recommendations take the form of subsets of data of multiple types appropriate to multiple tasks, supported by an explanation for the recommendation.

3 Design Guidelines

An initial workshop with researchers and expert information analysts was held to conceptualize a mixed-initiative analytic environment. Early in the prototyping process, a second workshop was held with approximately 14 information analysts to solicit feedback on an early version of ADE. Participants reviewed designs, watched a demonstration of the software and provided feedback. From the expert feedback, several themes emerged. 1) Users must remain in control of their process. Recommendations should enrich, not dictate, the process. 2) Recommendations should enhance the current task. Information presented for an earlier or later task is a nuisance. 3) It is essential for the user to understand what recommended information signifies and why it is recommended. 4) ADE should remove, rather than add, routine tasks for the user.

Based on this expert input and related research, we assert that an environment for mixed-initiative iterative sensemaking should support the following guidelines.

Guideline 1. Support externalization of thinking. As described in [10], mixed-initiative sensemaking systems should provide an environment in which users can capture information relevant to their ongoing task and refine their conceptual models. Development of a fully featured dynamic thinking environment is beyond the scope of the current prototype; our focus is on capturing entities and relationships relevant to a particular sensemaking task.

Guideline 2. Infer users' tasks based on their activities. Users should not be burdened with additional activities for the purpose of guiding the recommenders [20]. An appropriately instrumented environment should enable the inference of user's tasks based on clues from their interactions.

Guideline 3. Support iterative sensemaking by presenting semantically meaningful recommendations that enrich the current activity. Information that is relevant at one stage of the iterative sensemaking process may be irrelevant at earlier or later stages. Mixed-initiative iterative sensemaking environments should recommend relevant data based on the user's current activity and potential next step, rather than attempting to forecast several steps in the future. A task-oriented approach, which recommends data and relationships relevant to the current task, provides additional sophistication over a traditional search-based approach, which provides data in response to user-defined searches.

Guideline 4. Enable rapid visual interpretation of recommendations. Recommended data should be presented in compact visual form, to allow users to quickly assess the recommendation and explore or dismiss it with minimal effort.

Guideline 5. Provide recommendations in context. Recommendations should be presented in context of the user's analysis so that their potential relevance is apparent. Recommendations should be accompanied by a natural-language rationale that can be easily interpreted by the user.

4 Usage Example

Using these design guidelines, we developed the ADE prototype (Figure 1). In this section, we illustrate a use case for ADE using elements of the 2014 IEEE VAST Challenge [31]. In this fictitious scenario, several GAStech employees have disappeared and the Protectors of Kronos (POK) organization is suspected of involvement. Data include two weeks of GAStech email headers, credit card and loyalty card transaction histories, and vehicle tracking data for GAStech employees, a city map, historical documents, and news articles. We performed manual and automated data pre-processing to identify and geolocate businesses and employee homes, create a GAStech employee organization chart, and extract entities from narrative text documents.

To begin, the user creates "seeds" representing entities of interest, in this case "GAStech" and "POK." ADE attempts to infer the user's task based on these actions, but no specific task can be identified at this stage. The recommender assumes a general task and seeks information related to persons, places, or organizations with these names. In addition, the recommender seeks information that may relate these entities.

The resulting recommendation sets are automatically prioritized, formatted, and displayed on the Canvas near the corresponding seed (Figure 1a). Each recommendation set appears in a separate card. In addition, evidence of relationships between seeds appear as dashed lines between the seeds. A recommendation card on the line provides evidence for the suggested relationship.

The analyst could inspect the recommended data in one of her analytical tools (Microsoft Excel, Google Earth, or Gephi in this example). She may ignore or dismiss recommendations. The recommended relationship between GAStech and POK is of urgent interest, so she explores this first.

The recommendation card associated with this relationship is for Isia Vann. Turning over the card shows a natural-language rationale for this recommendation, as well as a numeric relevance score. The explanation states that Isia Vann has had a role in both organizations.

This is relevant, so she creates a new seed from this recommendation.

In response, ADE seeks and presents several new recommendations on the Canvas, including summaries of Isia Vann's email usage, general driving patterns, and late-night driving patterns. Each recommendation corresponds to relevant data found by an underlying subtask model.

The recommendation summarizing late-night driving raises the user's interest. She double-clicks to show the geospatial recommendations in Google Earth (Figure 1b-c). This reveals that Isia Vann made late-night visits to the homes of two GAStech executives: Willem Vasco-Pais and Ada Campo-Corrente. She pins these locations from Google Earth onto the Canvas to add them to her investigation. This produces additional recommendations, including a relevance-ordered list of visitors to these homes. Inspecting the list of visitors to Vasco-Pais' home in Microsoft Excel, she identifies another visitor to this home: Hennie Osvaldo. She pins Hennie Osvaldo to the ADE Canvas from Excel, resulting in new recommendations, including possible meeting locations between Hennie Osvaldo and Isia Vann—overlapping stops of their vehicles in which they were at most 300 feet apart.

Based on this, she hypothesizes that Hennie Osvaldo and Isia Vann are suspects. The user creates groups labeled "Suspects" and "Victims," which have special meaning in a criminal investigation task model, as they indicate predefined categories of individuals for which specific questions need to be answered. She drags the seeds for Isia Vann and Hennie Osvaldo to the "Suspects" group, adding them to the group. This triggers new recommendations that present evidence of particular interactions between the suspects. She adds Willem Vasco-Pais and Ada Campo-Corrente to the "Victims" group. As a result, new recommendations appear, summarizing the interactions between suspects and victims (Figure 2).

Using ADE, the analyst has progressed rapidly from exploratory analysis through several iterations of investigation to the preliminary identification of suspects and victims.

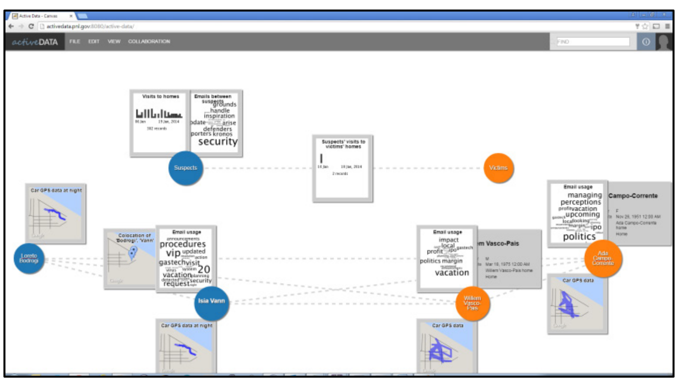

Figure 2. The user has created groupings for Suspects and Victims and has added individuals to the appropriate groups. ADE presents relevant recommendations based on this higher-order classification of the individuals, as shown by recommendations near the group seeds.

5 System Description

Below we describe the primary components of ADE.

5.1 System Overview

ADE consists of an Active Data Interface and an Active Data Assistant (Figure 3). The Active Data Interface is the user interface component, composed of a web-based Canvas user interface, a web server, a database, and interfaces to selected third-party analysis tools. The Active Data Assistant manages inference of user interests, instantiation of the appropriate task models, and recommendation of relevant data based on the current task. The Active Data Assistant passes recommendations to the Active Data Interface for presentation to the user.

The user incrementally externalizes her thought process by recording key entities and relationships on the Canvas throughout her analysis (Guideline 1). She spatially arranges and groups entities as appropriate.

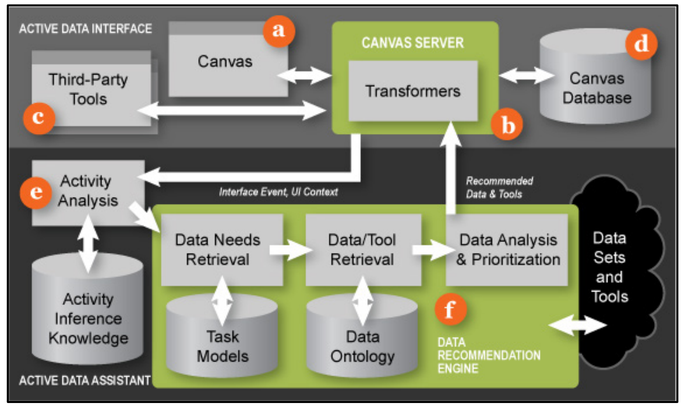

Figure 3. ADE consists of an Active Data Interface and an Active Data Assistant. The Active Data Interface consists of the Canvas (a), the web-based dynamic thinking space; the Canvas Server (b), which selects and transforms recommendations for presentation and coordinates communications with the Canvas and third-party tools (c); and the Canvas Database (d), which maintains the state of the Canvas. Based on communication from the Canvas Server, The Active Data Assistant performs activity analysis (e) to infer analytic tasks. The resulting information is passed to the Data Recommendation Engine (f) to search for data to satisfy the task models. This produces recommendations, which are prioritized and returned to the Canvas Data Server (b) for transformation and presentation.

ADE works as follows.

1) A user performs an interaction with the Canvas to record information of interest. Interactions of particular interest to ADE include creating or deleting a seed, creating a group, adding a member to a group, and creating a link between two seeds.

2) Each of these interactions generates a new Canvas state, which the Active Data Interface sends to the Active Data Assistant (Guideline 2). The state consists of the most recent user interaction as well as the state (displayed, grouped, linked) of all of the seeds, groups, and recommendations on the Canvas.

3) The Active Data Assistant performs activity recognition by considering the user's last interaction to add, remove or link entities, and the state of all entities on the Canvas. A catalog of possible activities, or task models, is compared to the state of the Canvas to determine candidate activities in which the user may be engaged. If data on the Canvas are sufficient to populate the input parameters of the task model, then this task is considered a candidate activity. The result is a mapping from user and Canvas to a particular activity.

4) Candidate tasks identified in this way trigger the execution of the corresponding data retrieval tasks (Guideline 3). Information from the Canvas populates parameters of modeled tasks associated with the recognized activity. This hierarchical collection of tasks is executed to retrieve and combine data. Some tasks are designed to retrieve data relevant to a single entity on the Canvas, while other tasks are designed to find information that connects two entities or entity groups. Tasks can range from simple information retrieval to complex analytic algorithms.

5) Recommended datasets are returned to the Canvas Server, where they are transformed into summarized visual representations (Guideline 4). These visual thumbnails help draw attention to patterns or anomalies. They are presented to the user on the Canvas in the context of the user's ongoing analysis (Guideline 5).

5.2 Active Data Assistant

The Active Data Assistant provides recommendations based on the user's interactions (Guideline 3). This requires the Active Data Assistant to do the following. 1) It must know what data sources and tools are available to the user. 2) It must know the conceptual schemas of those sources—the kinds of information they contain and how to access that information—and the capabilities of the associated analytic tools. 3) It must have (or be able to generate) structured models of tasks that the analyst might be executing. 4) It must track the user's current state [32], [33].

We assume the first two requirements are met by hand-engineered data and tool catalogs. Currently, tool choice is dictated by data type. If multiple tools were available for a given data type, tool choice would be based on attributes of the tools, given the task and the data volumes. In principle, the relevant characteristics of the data sources could be learned, as could some of the features of the tools. Similarly, with respect to the fourth requirement, we assume that the analyst's desktop and associated tools have been instrumented and that application programming interfaces (APIs) are available to support communication of the user's state. This instrumentation must be at the appropriate level of abstraction to convey operations that are analytically meaningful for the user. In this prototype, instrumentation problems are avoided by hand engineering the APIs between the Active Data Assistant and the Active Data Interface.

5.2.1 Task Models

The Active Data Assistant generates data recommendations via a library of task models and an engine that triggers and executes them. Each task model specifies the mapping between a user's analytic task and data relevant to that task. The models are parameterized to make representation of the task (and corresponding relevant data) specific to the current problem. The models are hierarchical in that they specify how a task can be decomposed into subtasks and how the data recommended for the various subtasks can be merged into data relevant for the parent task. In the current prototype, the task model library is authored and maintained by system developers.

Currently, task models are formulated in terms of specific data sources, so the models do not adapt easily to new data sources. In future work, we will build a data abstraction layer to specify the kinds of data required to execute a task model. Together with software that semi-automatically aligns data sources with a given ontology, this will allow future task models to be created at a higher level of abstraction not tied to specific data source schemas. We will formulate these data requirements in terms of a formal ontology characterizing the types of entities and relations among entities relevant to the task domain.

The components of a task model are as follows. 1) A *task name* that specifies the type of task. Table 1 lists the names of tasks implemented in the current prototype. 2) A set of *parameters*—variables that, when bound, instantiate the task according to the particular problem. For example, InvestigateSuspectsBehavior requires a single parameter representing a list of suspects being investigated. The task name is generally indicative of the parameters required. 3) A set of *data retrieval procedures*. These parameterized scripts specify how to retrieve a block of data relevant to the task. In addition to the script, a data retrieval procedure contains a parameterized English-language explanation of the data and its relevance for the task, and a numerical priority—an estimate of the probability that the retrieved data will be useful for completing the task. In our current prototype, data retrieval procedures are parameterized Structured Query Language (SQL) queries. Priorities are assigned by task model authors and are static. 4) A set of *subtasks* upon which completion of the parent task may depend. For example, InvestigateCrime includes two subtasks, InvestigateSuspectsBehavior and InvestigateVictims.

The task-subtask decomposition encoded in the task models is akin to Hierarchical Transition Network (HTN) planning models [43], although unlike HTNs, the subtasks in our task models do not necessarily constitute the complete definition of the parent task (e.g., investigating a crime entails more than investigating victims and suspects). 5) A set of *data merge procedures*, parameterized scripts that specify how to merge recommended data from the task's subtasks. Each data merge procedure includes an English explanation and a priority, as described earlier. In our current prototype, data merge procedures are parameterized SQL queries.

5.2.2 Recommendation Engine

The recommendation engine involves two steps: (1) perform activity recognition to infer the user's current analytic task(s), and (2) trigger and execute the appropriate task model to recommend data relevant to that task.

We take a simple approach to activity recognition in the current prototype. For each newly created seed, the activity recognition software identifies all possible tasks (Table 1) for which it can bind parameters (keeping the parameter types consistent with the seed types). In addition, special seed terms associated with criminal investigation, such as "Suspects" and "Victims," are explicitly mapped to specific default tasks (InvestigateSuspectsBehavior and InvestigateVictims, respectively). In future work, we intend to incorporate more robust activity recognition technology (e.g., [33]–[35]).

When the Active Data Assistant hypothesizes a task ***T*** based on a specific user activity, it retrieves the task model for ***T*** from the task model library. Typically, more than one ***T*** and one task model will be consistent with the current state of the analysis. The Active Data Assistant instantiates the parameters in the task models in accordance with the current state of the Canvas and executes the associated data retrieval procedures. Running a data retrieval procedure executes its query; any results are collected as a table. If the task model includes subtasks, the corresponding task models for the subtasks are invoked, and the resulting data is merged and joined into a table according to the data merge procedure. Rows are ordered according to a relevance score, described in Section 5.2.3.

Table 1. Active Data Assistant Task Models

Task Model Name
InvestigateCrime
InvestigateSuspectsBehavior
InvestigateVictims
FindConnectionBetweenPersonNames
FindLocationConnectionBetweenPersonNames
FindConnectionBetweenPersonNameAndOrganizationName
FindConnectionBetweenPersonNameAndPlaceName
FindConnectionBetweenOrganizationNames
FindEverythingAboutPersonName
FindEverythingAboutOrganizationName
FindEverythingAboutPlaceName
FindEverythingAboutString

The task models' data retrieval and data merge procedures range from simple search to sophisticated retrieval and data manipulation. The current task models for ADE include 1) basic full name normalization, 2) mention clustering and entity extraction within (but not across) documents, 3) proximity-based and region-based location and route retrieval, and 4) determination of temporal event overlap and time/proximity overlap.

When a user's interactions provide little evidence upon which to base a task hypothesis, the Active Data Assistant uses default tasks. This is particularly true early in the exploration process. For example, if the user types "John Doe" on a blank Canvas, the assistant cannot form a specific hypothesis about what role John Doe plays in the user's tasks. It can hypothesize a default task that the user wants to know interesting things about John Doe, where "interestingness" is determined by what others have found useful about people in the past. Adding "Mary Q Public" to the screen likely would still not provide enough evidence to form a specific task hypothesis, but the assistant can hypothesize a default task of looking for relationships between Doe and Public. Here, relationships may indicate direct linkages between the two entities or shared connections to a third entity.

In moving toward a scalable and robust ADE operational system, it is critical to have a workable process for creating and maintaining the task model library. The current model requires knowledge engineers to produce and maintain the task model library. In future work, we believe machine learning techniques can help partially automate the process by associating analyst-constructed queries with their current tasks.

5.2.3 Relevance Determination

If the task model returns data, the Active Data Assistant assigns relevance ratings for the overall recommendation and for the individual data rows in the recommendation. These ratings express the probability that a recommendation/row is relevant to the task being modeled. The relevance rating ***R*** for the recommendation as a whole is assigned by the task model. In the current prototype, it is a constant specified by the creator of the task model. In future work, we will explore the extent to which these recommendation rankings can be learned from user behavior, as a form of implicit or pseudo-relevance feedback.

Individual data rows for a recommendation are assigned a score, which is used for sorting. The row score estimates the relevance probability of the row, that is, the probability that the data in the row is relevant to the task. The minimum starting value for each the row-level relevance score is ***R***. A row score may increase based on the column values in that row as follows. A relevance probability is computed for the row given the column value, and in this way, a row score may increase based on the column values of that row.

A domain-independent metric based on outlier values is used to estimate column relevance. Outliers are detected using interquartile range, a standard statistical technique. For numeric columns, outlier values are detected. For nonnumeric columns, outlier frequencies of values are detected. Outlier relevance is capped at 0.1.

A column with special semantics may be assigned a custom relevance probability in place of the outlier metric. For example, a row related to a document has a column containing the number of mentions of a name within that document. The ratio of mentions to document size can be used as a relevance probability.

Column-level probabilities are assumed to be independent, although in practice the columns may be highly correlated. Furthermore, all relevance probabilities are presumed to be disjunctive; that is, a record is relevant if the recommendation, the row, or one or more columns of the row are judged to be relevant. These assumptions permit straightforward and sound aggregation of relevance probabilities by multiplying their complements (i.e., their irrelevance probabilities) and complementing that product. In future work we will explore learning these row and column-level probabilities from user behavior.

The recommended data are sorted by row relevance and sent to the Canvas Server, accompanied by the relevance score for the overall recommendation and the natural-language explanation for the recommendation. This natural-language explanation directly corresponds to the (sub)task model that produced the data.

5.3 Active Data Interface

The Active Data Interface captures user interactions and presents recommendations to the user. It consists of the Canvas web-based user interface, the Canvas Server, a supporting database for persisting Canvas content, and interfaces to third-party tools.

5.3.1 Canvas

The Canvas provides a thinking space in which the user can externalize her thinking about entities and relationships that emerge as relevant during the course of her analysis (Guideline 1). Unlike systems in which entities are explicitly and strongly typed, the Canvas does not enforce semantics [14], [15]. This supports a more friction-free user input but burdens the system to figure out what the user is expressing [6], [9], [21]. Entities can be added, removed, grouped, related, and organized as desired. Interactions are captured and used as a basis for inferring tasks (Guideline 2). Interactions being captured include creating a seed, creating a relationship, and creating a group, In turn, the system presents recommendations to the user in the context of her externalized thinking in the Canvas (Guideline 5).

ADE uses a consistent visual language for representation of user-created objects such as seeds (Figure 4a), relationships (Figure 4a), and groups (Figure 4b), as distinguished from system recommendations (Figure 4c–m).

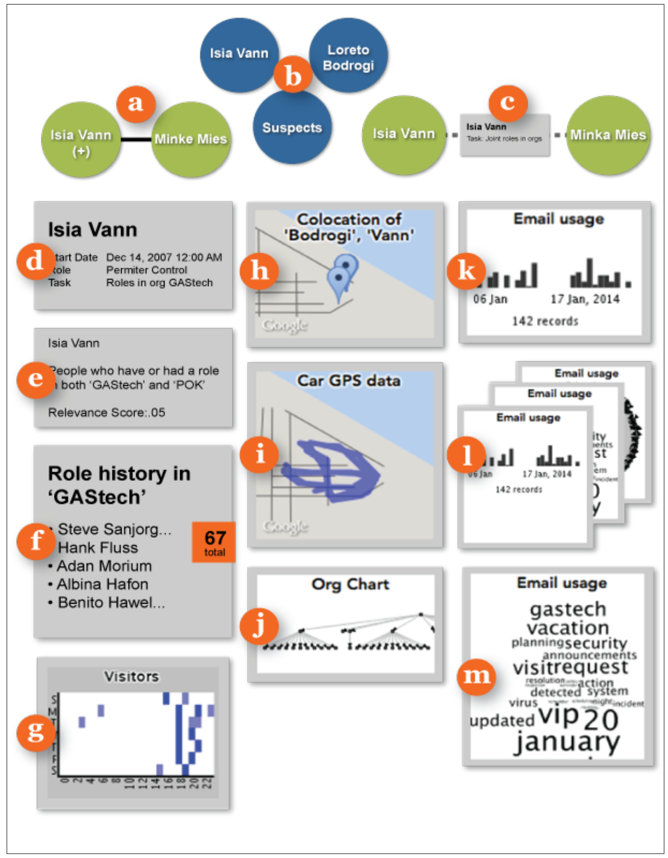

Figure 4. Visual representations in ADE. a) Entity "seeds" are represented as green circles. The (+) indicates that data have been merged into the seed; this data can be revealed by clicking the (+). The black line between seeds indicates that the user has explicitly created an edge between two seeds. b) A user-defined group of Suspects. Once added to a group, seeds assume the group color. c) Dashed lines between seeds indicate system-recommended relationships. A card in the center of the line gives detail about the suggested relationship. d) An entity recommendation includes the entity name and a list of attribute name-value pairs. e) Any recommendation can be "flipped" with a mouse click to display the natural-language explanation and relevance score for the recommendation. f) A list shows a recommendation consisting of several entities, presented in relevance order. Clicking an entity on the list creates an entity recommendation for it. The total list length appears on the right. g) Locations and h) routes are shown as markers and paths, respectively, on a map tile. i) Recommended sets of entities and relationships are shown in a subgraph. Temporal recommendations are shown using j) histograms of events over time and k) heatmaps showing day by hour patterns. l) Multiple representations are generated to emphasize different characteristics of the recommendation (such as textual, temporal, and geospatial). One representation is arbitrarily presented, and the user can cycle through the available representations. m) Large text recommendations are presented as a word cloud of prominent words.

Users create seeds through manual entry, pinning from a third-party tool, or conversion from an entity recommendation. Users may combine seeds or combine recommendations with a seed. When the user drags a seed onto another seed, those seeds are "merged" (combined) into a composite entity. When the user drags a recommendation onto a seed, that recommendation augments the seed, adding context to or data about the entity represented by the seed. Users define groups by creating a seed with a model-specific keyword like "Suspects" or with their own a user-defined term preceded by a "#" symbol. Each group is assigned a specific color.

Sensemaking tasks may require use of many analytic tools appropriate to the specific data types. ADE interfaces were developed for Microsoft Excel, Gephi, and Google Earth. Double-clicking any recommendation on the Canvas opens the associated data in the appropriate tool. We also implemented mechanisms to permit the user to "pin" the contents of an Excel spreadsheet cell or a Google Earth placemark to the Canvas.

5.3.2 Contextual Presentation of Recommendations

Transformers (Figure 3) produce compact visual representations of the raw recommendation sets provided by the Active Data Assistant. The presentations are selected based on the data attributes present in the recommendation set. These visual thumbnails are intended to provide insight into anomalies and patterns.

Active Data Interface uses the recommendation-level relevance scores assigned by the Active Data Assistant to select the most highly relevant recommendations for display. The items present on the Canvas are organized into a transparent hexagonal grid. This layout requires limited processing time and ensures sufficient space to render the cards for user inspection. The user can adjust the layout by moving seeds. Seeds are given priority and occupy the unoccupied hexagon nearest to the point at which the user placed them during a drag-and-drop operation. After all seeds have claimed a place in the grid, recommendations are placed surrounding their seed, until all six surrounding spaces are filled or all recommendations have been placed. Priority is given to the most relevant recommendations for each seed.

Recommendations placed on edges are a special case. These recommendations highlight aspects of a relationship between two entities (e.g., an employment record between a business and a person), so the recommendation is placed in an empty hexagon midway between the two entities. If there is not an empty hexagon in the vicinity, the edge recommendation is placed overlapping other recommendations at the midpoint of the edge.

6 Discussion

We discuss the strengths and limitations of our prototype, in the context of the design guidelines presented in this paper.

Guideline 1. Support externalization of thinking. ADE provides basic capabilities for externalizing representation of entities and relationships relevant to an iterative sensemaking activity. A true iterative sensemaking environment would require a richer visual and interactive vocabulary to be able to support semantically meaningful annotations by the user, alternate methods for organizing information, and critical thinking about assembled evidence.

Guideline 2. Infer users' tasks based on their activities. We have sought to eliminate the burden on the user to take steps strictly for the system's benefit, such as to provide explicit guidance about searches to perform, the syntax of those searches, or the data sources on which to search. The initial prototype provides a small set of activities through which the user can implicitly express interests (e.g., creating a seed, relationship, or group).

The current prototype makes use of explicitly recorded entities and relationships as a basis on which to infer tasks, but it does not take advantage of additional interactions that could provide direction to the Active Data Assistant. For example, we would like to incorporate inference of implicit relationships based on the user's organization and reorganization of the Canvas [6], [8], [9].

Likewise, interactions in third-party analytic tools could indicate user tasks and interests. The initial prototype demonstrates basic interactions with third-party tools, but more complete semantic-level integration would be required to provide the type of interoperability we envision. To fully realize this goal would require a common semantic API to share meaningful user actions between applications. This API should also permit recommendations from ADE to be passed to the third-party application.

Guideline 3. Support iterative sensemaking by presenting semantically meaningful recommendations that enrich the current activity. In the current prototype, we present a set of prioritized recommendations based on task models. In the early, exploratory phases of the task, the recommended information is relatively general and supports many potential tasks. As further analytic steps are completed, more specific models are executed to provide more refined recommendations.

ADE uses custom-written task models. While it would be extremely difficult to learn these models by observation of the user's activities, it might be feasible, deploying a well-instrumented analyst's desktop, to learn variants of existing task models.

The current use case assumes an unstructured, bottom-up approach, which is common for sensemaking tasks [3]. However, a more top-down, plan-driven approach to sensemaking provides rich opportunities for presentation of meaningful recommendations. Perer and Shneiderman [36] explored mechanisms for providing simple and flexible support for task execution. ADE could be applied to well-defined structured tasks to provide greater analytic support.

Guideline 4. Enable rapid visual interpretation of recommendations. Compact visual representations provide overviews of patterns in recommended data to help support selection of plausible or interesting next steps in the sensemaking process. In earlier versions of the prototype, only one representation was selected per recommendation. However, subject matter expert feedback and our own exploration made it clear that no one format was appropriate for revealing the multiple dimensions (geospatial, temporal, topical, or relational, for example) about which the particular recommendation may reveal patterns. The use of multiple alternate formats partially addresses this issue.

There are several approaches to optimizing the form of the visualization. Adapting to user preferences could lead to more useful initial representations.

Guideline 5. Provide recommendations in context. We assert that recommendations should be shown to the user within their ongoing analysis in a way that provides supporting context for the recommendation, and recommendations should be explained to the user in natural language. The current prototype presents one instantiation of such an approach.

7 Conclusion and Future Work

This paper presents a set of design guidelines for development of task-model-based mixed-initiative environments for supporting sensemaking. We have described a prototype that demonstrates these principles. We have described a motivating use case and discussed the design choices made in developing this prototype, along with the strengths and limitations of these choices.

There are many research questions to pursue further as a result of this work. Kang and Stasko [10] include detailed provenance tracking and reporting processes in their hypothetical system, which is a natural extension to the current work. In addition, we will extend the prototype to address sensemaking with dynamic datasets, which will raise questions of change blindness and attention management. We also intend to investigate techniques for addressing problems of scale as well as space management on the Canvas. We will explore generalization of our techniques to support greater adaptability to new data sources and more straightforward development and refinement of user models, as described in Section 5.2.1. We will explore the use of adaptive task and user models to improve the quality of recommendations presented. In addition, we plan to conduct a user study to evaluate the effects of this mixed-initiative environment on the iterative sensemaking process.

ACKNOWLEDGMENTS

We thank those who contributed to early conversations related to this work, including Larry Birnbaum, Danyel Fisher, Ellen Isaacs, Peter Lucas, Brian Moon, Catherine Plaisant, Emilie Roth, Jean Scholtz, Ted Selker, Steve Sickels, John Stasko, and Mark Zimmermann. We are grateful for the work of the technical teams who developed the ADE prototype, particularly William Deans, Mabry Tyson, Thomas Lee, David Gillen, and Ray Perrault. We acknowledge the support of the U.S. Government for this research.

REFERENCES

[1] J. J. Thomas and K. A. Cook, "Illuminating the path," 2005. [Online]. Available: http://nvac.pnl.gov/agenda.stm#book.

[2] P. Pirolli and S. Card, "Sensemaking Processes of Intelligence Analysts and Possible Leverage Points as Identified Though Cognitive Task Analysis," *Proc. 2005 Int. Conf. Intell. Anal. McLean Va.*, p. 6, 2005.

[3] P. Zhang, "Sensemaking: Conceptual Changes, Cognitive Mechanisms, and Structural Representations. A Qualitative User Study," University of Maryland, College Park, 2010.

[4] T. M. Green, W. Ribarsky, and B. Fisher, "Building and applying a human cognition model for visual analytics," *Inf. Vis.*, vol. 8, pp. 1–13, 2009.

[5] E. Horvitz, "Principles of Mixed-initiative User Interfaces," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 1999, pp. 159–166.

[6] A. Endert, "Semantic Interaction for Visual Analytics: Toward Coupling Cognition and Computation," *IEEE Comput. Graph. Appl.*, vol. 34, no. 4, pp. 8–15, Jul. 2014.

[7] T. Ruotsalo, G. Jacucci, P. Myllymäki, and S. Kaski, "Interactive Intent Modeling: Information Discovery Beyond Search," *Commun ACM*, vol. 58, no. 1, pp. 86–92, Dec. 2014.

[8] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang, "Dis-function: Learning Distance Functions Interactively," in *IEEE VAST*, 2012.

[9] A. Endert, P. Fiaux, and C. North, "Semantic interaction for visual text analytics," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2012, pp. 473–482.

[10] Y. Kang and J. Stasko, "Characterizing the intelligence analysis process: Informing visual analytics design through a longitudinal field study," 2011, pp. 21–30.

[11] P. Zhang and D. Soergel, "Towards a comprehensive model of the cognitive process and mechanisms of individual sensemaking," *J. Assoc. Inf. Sci. Technol.*, vol. 65, no. 9, pp. 1733–1756, 2014.

[12] C. Andrews and C. North, "Analyst's Workspace: An embodied sensemaking environment for large, high-resolution displays," in *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2012, pp. 123–131.

[13] W. Wright, D. Schroh, P. Proulx, A. Skaburskis, and B. Cort, "The Sandbox for analysis: concepts and methods," presented at the CHI '06, 2006, pp. 801–810.

[14] C. Andrews, A. Endert, and C. North, "Space to think: large high-resolution displays for sensemaking," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010, pp. 55–64.

[15] F. M. Shipman III and C. C. Marshall, "Formality Considered Harmful: Experiences, Emerging Themes, and Directions on the Use of Formal Representations in Interactive Systems," *Comput. Support. Coop. Work CSCW*, vol. 8, no. 4, pp. 333–352, Dec. 1999.

[16] N. Mahyar and M. Tory, "Supporting Communication and Coordination in Collaborative Sensemaking," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 1633–1642, Dec. 2014.

[17] E. A. Bier, E. W. Ishak, and E. Chi, "Entity Workspace: An Evidence File That Aids Memory, Inference, and Reading," in *Intelligence and Security Informatics*, vol. 3975, S. Mehrotra, D. D. Zeng, H. Chen, B. Thuraisingham, and F.-Y. Wang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 466–472.

[18] A. Endert, S. Fox, D. Maiti, and C. North, "The semantics of clustering: analysis of user-generated spatializations of text documents," in *Proceedings of the International Working Conference on Advanced Visual Interfaces*, 2012, pp. 555–562.

[19] "i2 Analyst's Notebook." [Online]. Available: http://www.i2inc.com/products/analysts_notebook/. [Accessed: 20-Apr-2006].

[20] A. Endert, M. S. Hossain, N. Ramakrishnan, C. North, P. Fiaux, and C. Andrews, "The human is the loop: new directions for visual analytics," *J. Intell. Inf. Syst.*, vol. 43, no. 3, pp. 411–435, Dec. 2014.

[21] L. Bradel, C. North, and L. House, "Multi-Model Semantic Interaction for Text Analytics," in *IEEE Conference on Visual Analytics Science and Technology*, 2014.

[22] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. New York: Cambridge University Press, 2008.

[23] F. Ricci, L. Rokach, and B. Shapira, "Introduction to Recommender Systems Handbook," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer US, 2011, pp. 1–35.

[24] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," *Adv. Artif. Intell.*, vol. 2009, pp. 1–19, 2009.

[25] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based Recommender Systems: State of the Art and Trends," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer US, 2011, pp. 73–105.

[26] R. J. Mooney and L. Roy, "Content-based book recommending using learning for text categorization," 2000, pp. 195–204.

[27] P. Melville, R. J. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering for improved recommendations," in *Proceedings of the Eighteenth National Conference on Artificial Intelligence*, Edmonton, Canada, 2002, pp. 187–192.

[28] A. Toledo, K. Sookhanaphibarn, R. Thawonmas, and F. Rinaldo, "Personalized Recommendation in Interactive Visual Analysis of Stacked Graphs," *ISRN Artif. Intell.*, vol. 2012, pp. 1–8, 2012.

[29] D. Koop, C. E. Scheidegger, S. P. Callahan, J. Freire, and C. T. Silva, "VisComplete: Automating Suggestions for Visualization Pipelines," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1691–1698, Nov. 2008.

[30] E. Maguire, P. Rocca-Serra, S.-A. Sansone, J. Davies, and Min Chen, "Visual Compression of Workflow Visualizations with Automated Detection of Macro Motifs," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2576–2585, Dec. 2013.

[31] M. Whiting, K. Cook, G. Grinstein, K. Liggett, M. Cooper, J. Fallon, and M. Morin, "VAST challenge 2014: The Kronos incident," presented at the Visual Analytics Science and Technology (VAST), 2014 IEEE Conference on, Paris, France, 2014, pp. 295–300.

[32] E. Charniak, "A Bayesian model of plan recognition," *Artif. Intell.*, vol. 64, no. 1, pp. 53–79, Nov. 1993.

[33] O. Madani, H. Bui, and E. Yeh, "Efficient online learning and prediction of users' desktop actions," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, Pasadena, California, USA, 2009, pp. 1457–1462.

[34] H. Bui, "A General Model for Online Probabilistic Plan Recognition," in *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2003, pp. 1309–1315.

[35] J. Shen, L. Li, T. G. Dietterich, and J. L. Herlocker, "A hybrid learning system for recognizing user tasks from desktop activities and email messages," 2006, pp. 86–92.

[36] A. Perer and B. Shneiderman, "Systematic yet flexible discovery: guiding domain experts through exploratory data analysis," 2008, pp. 109–118.