Spring 5-18-2018

Inferring Intent from Interaction with Visualization

Ran Wan
Washington University in St. Louis

Washington University in St. Louis

School of Engineering and Applied Science

Department of Computer Science and Engineering

Thesis Examination Committee:
Alvitta Ottley, Chair
Roman Garnett
Caitlin Kelleher

Inferring Intent from Interaction with Visualization

by

Ran Wan

A thesis presented to the School of Engineering and Applied Science
of Washington University in partial fulfillment of the
requirements for the degree of

Master of Science

May 2018
Saint Louis, Missouri

Contents

List of Tables

List of Figures

Acknowledgments

I want to first thank Dr. Ottley for allowing me to participant in such an exciting research project. Her wonderful vision in this subject and patient guidance play such a pivotal role in this project and in my entire research experience.

I want to thank Dr. Garnett for proposing the particle filtering algorithm, which is the building block of my model. His help in shaping and implementing the algorithm is tremendous.

I want to thank Engineering School for funding my 2017 Research Experiences for Undergraduates program as well as providing me with the opportunity to take advanced courses and participate in research in a relatively early stage of my undergraduate program.

Thank all the students who volunteered to participant in the pilot experiment study.

Lastly, I want to thank all the people who encouraged and motivated me during the project. It's been a great journey for me.

Ran Wan

Washington University in Saint Louis
May 2018

ABSTRACT OF THE THESIS

Inferring Intent from Interaction with Visualization

by

Ran Wan

Master of Science in Computer Science

Washington University in St. Louis, May 2018

Research Advisor: Professor Alvitta Ottley

Today's state-of-the-art analysis tools combine the human visual system and domain knowledge, with the machine's computational power. The human performs the reasoning, deduction, hypothesis generation, and judgment. The entire burden of learning from the data usually rests squarely on the human user's shoulders. This model, while successful in simple scenarios, is neither scalable nor generalizable. In this thesis, we propose a system that integrates advancements from artificial intelligence within a visualization system to detect the user's goals. At a high level, we use hidden unobservable states to represent goals/intentions of users. We automatically infer these goals from passive observations of the user's actions (e.g., mouse clicks), thereby allowing accurate predictions of future clicks. We evaluate this technique with a crime map and demonstrate that, depending on the type of task, users' clicks appear in our prediction set 79% – 97% of the time. Further analysis shows that we can achieve high prediction accuracy after only a short period (typically after three clicks). Altogether, we show that passive observations of interaction data can reveal valuable information about users' high-level goals, laying the foundation for next-generation visual

analytics systems that can automatically learn users' intentions and support the analysis process proactively.

Chapter 1

Introduction

This thesis is based on:

Ran Wan, Roman Garnett, & Alvitta Ottley. Inferring Intent from Interaction. In submission.

1.1 Introduction

Humans are entering a new era in which data increasingly surround us. This new access to data promises fresh opportunities for increased awareness, informed decision-making, and enhanced quality of life. Our most significant challenge now is not accessing more data, but making sense of it all. Visualization has been widely successful in helping people explore, reason, and make judgments with data. By and large, this success is due to fusing human intuition with interactive tools [26] and analysts in areas such as medicine, business, and government make daily decisions with visualizations.

Unfortunately, as data become larger and more complex, the utility of visualization systems in many real-world situations declines. The state-of-the-art analysis tools assume a work distribution that leverages the human visual system and domain knowledge with the machine's computational power [22, 30, 54]. Typically, it is the human that performs the reasoning, deduction, hypothesis generation, and judgment. The entire burden of learning from the data rests squarely on the human user's shoulders [28]. This model, while successful in simple scenarios, is neither scalable nor generalizable. For visualizations to continue to be useful in scenarios with increasingly complex and large datasets, we must create systems that can better support the human during the analysis process.

The work in this thesis is ambitious in setting an agenda for revolutionizing the design of visual analytics systems. We propose a more-intelligent system where the computer learns about the user, her analysis process, and her task as she uses the system to shape hypotheses and gain insight from the data. Beyond the scope of this thesis, our long-term goal is to create systems that will enable the human–computer team to function more like equal collaborators, but to achieve this, the system must have the capability to leverage interaction data to learn about the user's goals and the user herself.

Learning from interaction data is a critical challenge in the visualization community [93]. Prior work has largely focused on history tracking [8, 16, 58, 66, 102], annotating the analysis process for reuse [35, 59, 66, 72, 69, 92, 102], and automatic model parameter updates [13, 43, 42, 53, 120]. More recent work apply a grammar-based approach to capture the interaction process [31]. We build on this prior work but take a proactive approach to user modeling. Instead of analyzing interactions to inform future tasks, our approach focuses on supporting users' current tasks in real-time.

In this thesis, we introduce a probabilistic model to recognize and predict future behavior with visualization systems. At a high level, we use hidden unobservable states to represent goals/intentions of users and observable states to represent observed user actions (e.g., mouse clicks, eye gaze data), thereby allowing us to estimate and predict users goals/intentions through their actions. To evaluate our approach, we analyzed click-stream data from a map visualization that shows a real-world dataset of reported crimes (see figure 3.1). Our results show that the probabilistic model achieves, depending on the type of task, between 79% and 97% accuracy at predicting future mouse clicks. Further analysis shows that we can achieve high prediction accuracy in a short period (typically after three clicks). Altogether, we show that passive observations of interaction data can reveal valuable information about users' high-level goals. We discuss how this technique can be used to create next-generation visual analytics systems that can automatically learn users' intentions to support the analysis process better.

We make the following contributions:

- *A design-agnostic approach to modeling interaction with visualization:* We provide a design-agnostic approach to automatically extracting users' intent with visualization systems and demonstrate, using a scatter plot, how to model users' interests and actions.

- *Predicting future clicks from passive observations:* We demonstrate how to apply this model to a real-world visualization and dataset. Our proof-of-concept experiment validates that we can use this approach on real systems for real-time predictions. We demonstrate an overall prediction accuracy of 88% at guessing the next click based on prior observations.

- *Implications for designing adaptive visualizations:* We discuss techniques for adapting the visualization in real time and contribute to next-generation of visual analytics systems that actively support users during their analysis.

1.2 Prior Work

Analyzing interactions to learn about the user or an interface design has been a significant area of research in many different areas of computer science. For example, in the machine-learning community, researchers have used interaction data to model and predict users' browsing behaviors on websites and web search systems [41, 74, 75, 107]. Some researchers have also used interaction data to explore how the interface design can bias user behaviors [62, 73], and how to overcome these biases [73].

In databases, Battle et al. [7] analyzed interaction data to improve prefetching techniques. They showed that analyzing behavioral data resulted in a 430% improvement in system latency. In the HCI field, researchers showed that displaying interaction history of past users improves the problem-solving of future users [119]. Furthermore, Gajos et al. developed the SUPPLE system that can learn the type and degree of a user's disability by analyzing mouse interaction data [49, 50, 51]. Fu et al. developed statistical and machine-learning models to predict behavior on crowdsourcing annotation and web search tasks [48].

1.2.1 Analytic Provenance

It is a common belief that interaction logs contain crucial information about an analyst's reasoning process with a visualization [93]. Through interaction with a visual interface, analysts explore data, form and revise hypotheses, and make judgments. To better understand how analysts make sense of data, Pirolli and Card created the *sensemaking loop* which explains the analyst's progression from data foraging through hypothesis generation and insight [96]. The term *provenance* refers to the history of an object or idea, and *analytic provenance* researchers aim to track and analyze the analytics process [46, 98, 82, 83]. At a high level, the goal is to automatically capture and encode interactions with a visual interface to infer analysts' goals and intentions. Researchers and practitioners can then recall, replicate, recover actions, communicate, present, and perform meta-analyses on the analysis process [98].

A standard approach to recovering the analytic process is to capture low-level user actions such as mouse and keyboard events. For example, Cowley et al. developed *Glassbox* with the goal of logging interactions to infer intent, knowledge, and work-flow automatically [28]. Dou et al. demonstrated that is it possible to extract high-level information from interaction data [33]. They conducted a user study and recorded interactions while financial analysts used a visual analytics system to detect wire fraud. Through a manual analysis of the interaction data, they showed that is possible to recover analysts' strategies, methods, and findings. More recent work by Dabek and Caban introduced a grammar-based approach to modeling user interactions [31]. They used automatons to model users' behavior and demonstrated that their technique could capture user's analytic process.

A series of work focused on recording, annotating, and maintaining interaction history, and demonstrates the benefit of preserving a linear history for future use [8, 12, 16, 58, 66, 102]. *VisTrails*, for instance, automatically keeps track of the analyst's workflow and pipeline, making it possible for the user to resume, reuse and share their explorations [8, 16, 47]. Heer et al. [66] and Javed and Elmqvist [70] created graphical history tools that would allow users to track, recall and share their process. Gotz et al. developed tools for supporting the sensemaking process by augmenting existing data with user annotations [58].

1.2.2 Analyzing Interaction to Infer User Attributes

Researchers have also used interaction logs to infer user knowledge. Brown et al. used *Dis-Function* to learn analysts' knowledge through direct manipulation of visual elements [13]. They allow users to express their domain knowledge by grouping similar points then used this information to update the underlying distance function for their data projection automatically. Prior work also demonstrate how interaction data can be used to steer computation and refine model parameters [43, 42, 53, 88, 120]. For example, Endert et al. [43, 42] designed *ForceSPIRE,* which is a text data analysis tool that automatically updates the underlying layout model as users interact with documents. Guo et al. analyzed interaction logs to understand how analysts achieve insights [63].

Other researchers analyzed interaction data to infer individual characteristics. For instance, Brown et al. used machine-learning techniques to infer user attributes automatically [14]. They showed that off-the-shelf algorithms could successfully predict completion time and personality traits based on low-level mouse clicks and moves [14]. They also demonstrated the viability of making real-time inferences from passive observations. Ottley et al. analyzed clickstream data to demonstrate a correlation between personality traits and search strategies with hierarchical visualizations [87]. Lu et al. used eye-tracking data to select parameters for a visualization automatically [78]. Also utilizing eye gaze data, Steichen et al. [109] and Toker et al. [114] predicted cognitive traits such as visual working memory, personality, and perceptual speed.

Chapter 2

Inferring Technique

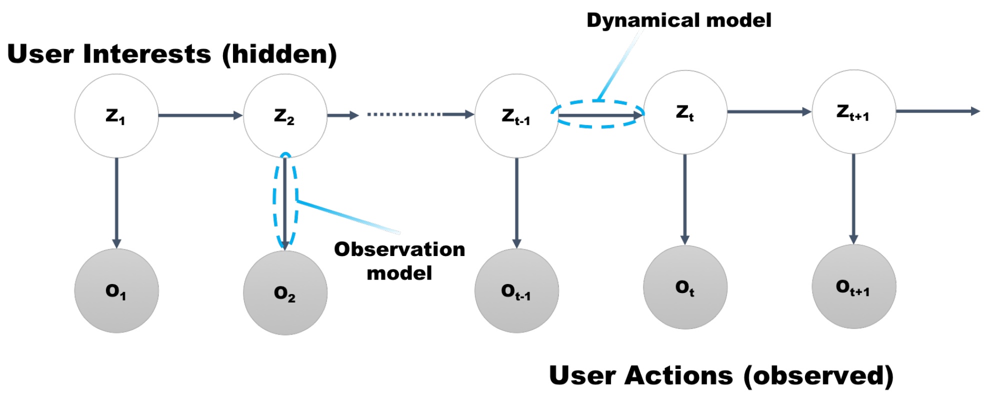

Figure 2.1: A hidden Markov model approach to modeling actions and interest with a visualization system. We represent users' interest as a sequence of latent variables in the hidden state space. Observable states are the user's actions. The conditional distribution of each observation depends on the state of the corresponding latent variable.

2.1 Overview of Approach

The previous section recaps prior work aimed at learning from interaction data. Much of the proposed approaches in the visualization community have primarily focused on analyzing behavior for tracking analytic provenance. While some of the prior work on applying machine-learning techniques to predict individual characteristics demonstrate the feasibility of real-time predictions, it is not clear how these techniques can generalize beyond the tested interfaces. Our objective is to introduce a generalizable and design-agnostic approach to user modeling.

To achieve this research goal, we model users' actions and their latent interests with a *hidden Makov model*. We will use *hidden* states to represent the latent interests of users and *observable* states to describe the actions of the user (e.g.. mouse clicks, mouse hovers, eye gaze data, etc.). Figure 2.1 shows an overview of the hidden Markov model used. We represent users' interests as a sequence of latent variables. The conditional distribution of each observation depends on the state of the corresponding latent variable. In the following section, we provide details for this approach and how we may automatically infer hidden user intent from passive observation.

2.2 Constructing a Probabilistic Model

First, we define a discrete time index t associated with interactions with a visualization. At the start of exploring the dataset, we define $t = 0$. This index will then increment every time a participant interacts with a visual element. Our model will presume that there is a *hidden, unobserved* state z_t representing the intent of the user at time t. We will assume that we can map the sequence of observed interactions $\{o_t\}$ to this hidden sequence of intentions. The task we consider here is how to *infer* the hidden intent of the user by observing their sequence of interactions. Our approach will be to construct a probabilistic model of the user's intent and their interactions, which we will use to drive the inference procedure. This model will be a *hidden Markov model,* presuming the user's intention evolves under a Markov process (that is, the intention at a particular time only depends on the intent at the previous time step), and interaction events are generated conditionally independently given this sequence of intents. To specify this model, we need to define the following:

- **Unobservable States:** A space of the possible intents.
- **Observable States:** A space of possible interactions.
- **Dynamical Model:** A model of the evolution of the user's intent over time.
- **Observation Model:** A model of how intent gives rise to observed actions.

We will discuss each of these in turn at a high level below, before considering an explicit construction of our proposal on a scatter plot.

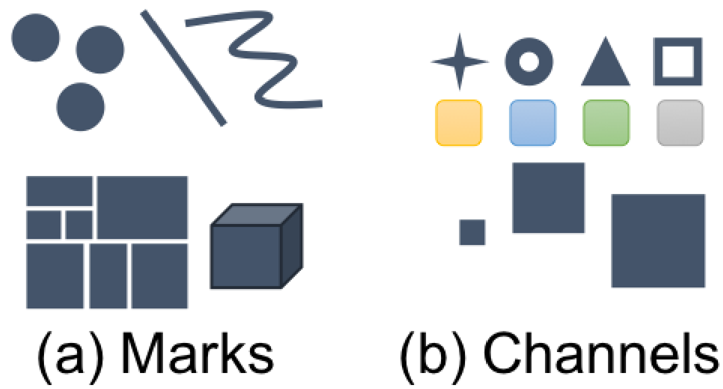

Figure 2.2: Visual marks and channels.

2.2.1 Defining Unobservable and Observable States

A crucial component of the probabilistic model is the specification of a hidden state space, which will represent the interest/intent of the user at a given time. In general, we propose that designers can tailor this space for a given scenario. One mechanism for reasoning about this intent is to reason about the aspects of a visualization that could be related.

For a given visualization, we define $\mathcal{M}$ as the mark space that specifies the types of visual marks and channels used in the visualization. Visual marks are geometric elements, and there are four primitive types: points, lines, areas, and volumes [9]. Visual channels describe the graphical properties of visual marks such as position, size, color, luminance, shape, texture, and orientation [9]. Together with Card et al.'s data-mapping principles [18] these design guidelines can be used to describe any existing visual representation [17]. We create $\mathcal{M}$ by decomposing the visualization into its primitive visual marks and channels.

In many scenarios, we may reasonably assume the users' interest at a given time to be related to some weighted subset of visualization marks, for example, dots of a particular size, color,

Class	Description	Example
U	*Unstructured* (can only distinguish presence or absence)	ErrorFlag
N	*Nominal* (can only distinguish whether two values are equal)	1,2,3
O	*Ordinal* (can distingush whether one value is less or greater but not difference or ratio)	<Small, Medium, Large>
I	*Interval* (can do substraction on values, but no natural zero and cannot compute ratios)	[10 Dec. 1978–4 Jun. 1982]
Q	*Quantitative* (can do arithmetic on values)	[0–100] kg
Q_s	—*Spatial variables*	[0–20] m
Q_m	—*Similarity*	[0–1]
Q_g	—*Geographical variables*	[30^{o}N–50^{o}N]Lat.
Q_t	—*Time variables*	[10–20] sec

Table 2.1: Card et al.'s data-mapping principles states any visual channel can be mapped to following data classes.

shape or in a specific location. In such a case we may define the latent intent space, $\mathcal{I}$, as a subspace of $\mathcal{M}$.

In contrast to the hidden intent space, the space of observed actions is typically easy to define. We may define o_t to be an observation of the user at time t, where this observation will be an interaction with visual elements (e.g., mouse clicks, mouse moves, eye gaze, etc.). We will represent each observation as a vector specifying the visual attributes of o_t.

2.2.2 Dynamical Model

The full specification of a hidden Markov model requires defining a probabilistic model of the dynamics of the hidden state space, that is how the user's latent intent changes from one time-step to the next. This is given by defining a probability distribution $p(z_{t+1} \mid z_t)$. We propose that this model should be reasonably easy to define in most visualization settings. In general, it is unlikely that the user's intent will change rapidly from one interaction event to the next. Therefore we can often choose this dynamics model to represent a simple random diffusion in the latent space:

$$z_{t+1} = z_t + \varepsilon,$$

where ε is some appropriate noise distribution. This model suggests that the user's intent tends to stay the same from time-step to time-step, perhaps with some slow drift as the user continues to interact with the system.

If a visualization setting may comprise a sequence of separate tasks, we may also construct dynamical models that loosely encode that user's intent may change in one of two ways: either the current task has not yet completed, in which case we may assume a simple drift model as described above. Otherwise, if the task has completed, we might model the intent at the next time step as being drawn from some broad distribution over the space of possible intents. In such a construction our dynamical model would be a mixture distribution with two components corresponding to the continuation of a task or beginning a new task.

2.2.3 Observation Model

We must also specify an observation model $p(o_t \mid z_t)$, which defines how latent user intent generates interactions. We must take care to define such an observation model appropriately for a given scenario, and we will demonstrate how we might construct an explicit example in our user study below. In a visualization setting, defining a reasonable choice for such a model is relatively straightforward. If a user's intent is represented by some values in the same space as the visual elements in the visualization, we may often construct an observation model that loosely specifies that "users click on objects related to their hidden intent." We will show an explicit construction of such a model in our scatter plot visualization case study below.

2.2.4 Predicting Movement

Our goal at each time stamp is to predict the user's possible next interactions given the set of the user's previously observed interactions. To approach this goal, we will use our hidden Markov model to infer the intent of the user at time t, z_t, given the interactions up to time t, $C_t = \{c_i\}_{i=1}^{t}$. Unfortunately, this inference is usually not possible in closed form, but we can use a particle filter. Particle filter is a well-established technique for inferring the hidden states of dynamical systems such as ours [34, 57].

We represent our belief about the latent state z_t given the previous clicks C_t with a set of m *particles* $\{z_t^{(i)}\}_{i=1}^m$, each particle a point in the intent space. These particles represent samples from the posterior distribution $p(z_t \mid C_t)$. Suppose for induction that we have a set of such particles. Particle filtering proceeds by repeating the following steps:

- We push the particles through the dynamical model $p(z_{t+1} \mid z_t)$ by sampling a new value for each particle:
$$z_{t+1}^{(i)} \sim p(z_{t+1} \mid z_t = z_t^{(i)}).$$
- We observe the next interaction event c_{t+1} and weight the particles according to the agreement with the observation by evaluating the observation model:
$$w^{(i)} = p(c_{t+1} \mid z_{t+1} = z_{t+1}^{(i)}).$$
- We sample a new set of m particles by sampling with replacement from the set of existing particles with probability equal to the weights $\{w^{(i)}\}$.

This set of resampled particles will represent a sample from the distribution $p(z_{t+1} \mid C_{t+1})$, and we may proceed inductively.

For each timestamp, we can get $p(z_t \mid C_t)$, which is the particle given all previous clicks. However, particles can be at any location on the visualization. Our goal is to find possible visual element users are going to interact with at the next time stamp.

To do this, we need one extra step. We treat every mark on the visualization as a potential candidate for the next interaction. We sum the weight every particle contribute to that candidate (where the observational model computes weight). A subset of marks with highest weights, α, will be considered as predictions. Notice that the size of α at this point is arbitrary.

$$d* = \arg\max_{d \in D} \sum_{p \in P} \mathbb{P}(d|p)$$

2.3 Explaining By Example

We now apply this model to a visualization interface. Using the simple scatter plot in Figure 2.3, we demonstrate how to define the hidden state space and discuss choices for the dynamical and observation models. In this example, we assume that users interact with visual marks by clicking on them.

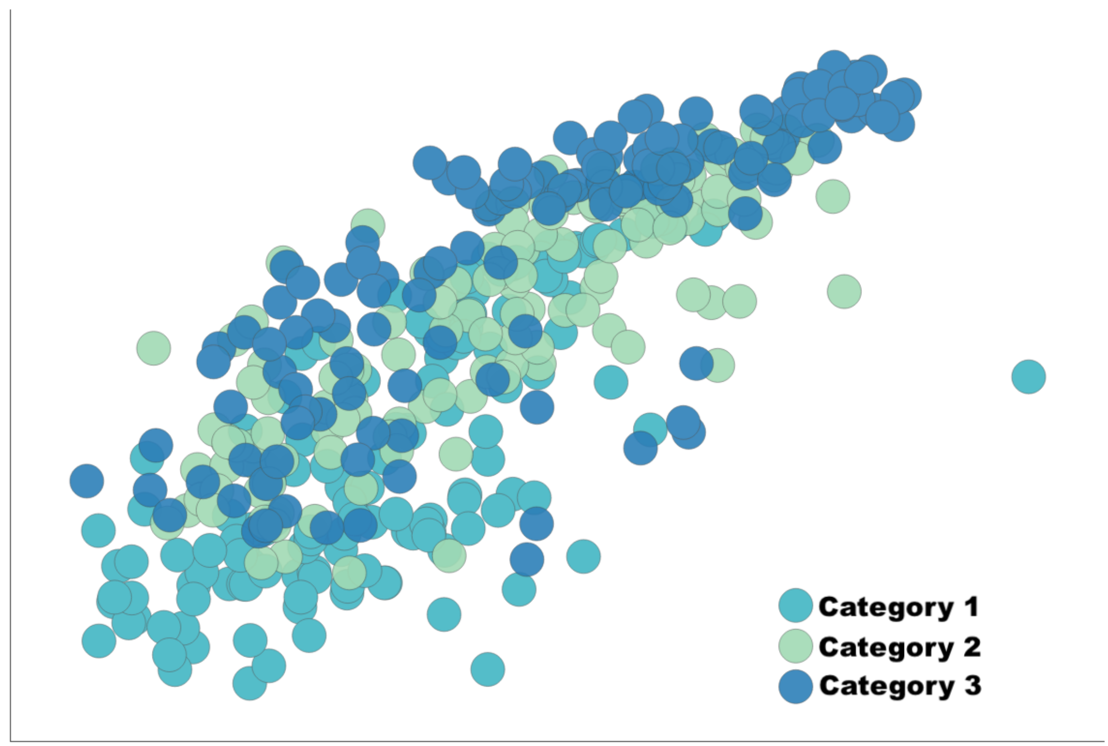

Figure 2.3: We use a simple scatter plot to demonstrate the approach.

2.3.1 Defining Unobservable and Observable States

We define c_t to be the click event at time t, which we will represent as a three-dimensional vector $c_t = (x'_t, y'_t, k'_t)$, where (x'_t, y'_t) is the x-coordinate and y-coordinate of the click and k'_t is the color of the circle clicked, represented by a discrete integer-valued index ranging over

Notation	Data Class	Description		Visual Structure
x'_t	Q	x-coordinate	$\rightarrow$	x-axis
y'_t	Q	y-coordinate	$\rightarrow$	y-axis
k'_t	N	category	$\rightarrow$	color

Table 2.2: Data mapping for the sample visualization.

the three possible values ($\{1, 2, 3\}$). Note that we use prime symbols to indicate quantities associated with a click event.

Next, we will define a hidden state space modeling the intent of the user. Each point in this hidden space is a vector specifying (1): a location (x, y) of interest, (2): a mark color k of interest, and (3): a trade-off parameter indicating the relative importance of location and mark color. For this example, we represent the trade-off parameter as a number $\pi \in [0, 1]$, with 1 indicating a complete focus on location and 0 indicating a complete focus on mark color. A point in this latent intent space is thus a four-dimensional vector $z = (x, y, k, \pi)$.

Our model assumes that at every discrete time step t in the interaction process (each time the user makes a click), the user has an underlying intent z_t corresponding to a vector in the intent space defined above. We seek to infer the intent of the user through observing the sequence of click events $\{c_t\}$. We will approach this inference problem via creating a hidden Markov model and performing inference with particle filtering.

Our model is fully specified by a dynamical model $p(z_t \mid z_{t-1})$ describing how the hidden state evolves and an observation model $p(c_t \mid z_t)$ describing how a hidden intent vector generates click events. We define each of these below.

2.3.2 Dynamical Model

For the dynamical model of the hidden state $p(z_t \mid z_{t-1})$, we adopt a simple stationary diffusion model. We assume that the intent at time $t + 1$ is typically similar to the intent at the previous time step t; that is, that intent does not change rapidly over time. We further assume that the four components of the intent vector evolve independently:

$$p(z_{t+1} \mid z_t) = p(x_{t+1} \mid x_t)p(y_{t+1} \mid y_t)p(k_{t+1} \mid k_t)p(\pi_{t+1} \mid \pi_t).$$

We model the evolution of the continuous location and location–color trade-off parameters with a simple Gaussian drift:

$$p(x_{t+1} \mid x_t, \sigma_x) = \mathcal{N}(x_{t+1}; x_t, \sigma_x^2);$$
$$p(y_{t+1} \mid y_t, \sigma_y) = \mathcal{N}(y_{t+1}; y_t, \sigma_y^2);$$
$$p(\pi_{t+1} \mid \pi_t, \sigma_\pi) = \mathcal{N}(\pi_{t+1}; \pi_t, \sigma_\pi^2).$$

The expected value of these parameters is equal to the previous value, with zero-mean Gaussian diffusion with parameter-dependent variance added. We will select these parameters σ_x, σ_y, and σ_π. Notice also that these three parameters are all also bounded values: the locations x and y indicates a position on the scatter plot and must lie in its domain, and the tradeoff parameter π must lie in the interval $[0, 1]$. Therefore, we need to deal with cases when the diffused value steps outside the boundary. Here we simply adopted a rule that whenever a diffused value steps outside the boundary for a variable, we move it onto the boundary in the direction of diffusion. For example, if π_{t+1} diffuses to value greater than 1, we will set it to 1; likewise if the diffused location (x_{t+1}, y_{t+1}) lies beyond the width and height of the scatter plot, we will project onto the nearest point on the canvas boundary.

Lastly, because mark color is a categorical value, we cannot directly apply normal diffusion to it. Here we used a discrete analog of that diffusion. We define a transition probability q and assume that with probability q the latent mark color of interest does not change. Otherwise, a new mark color of interest is chosen from all possible values with equal probability:

$$p(k_{t+1} \mid k_t, q) = q\delta(k_t) + (1 - q)\mathcal{U}(K),$$

where δ is a Kronecker delta distribution and $\mathcal{U}(K)$ is a uniform distribution over the mark colors. Again this choice models our assumption that intent typically changes slowly over time.

2.3.3 Observation Model

We must also specify an observational model $p(c_t \mid z_t)$ modeling the probability of a click event $c_t = (x'_t y'_t, k'_t)$ given the intent $z_t = (x_t, y_t, k_t, \pi_t)$ at time t. A brief summary of this observational model is that we flip a coin with heads probability equal to the location–color tradeoff parameter π_t. If the coin lands heads, we assume the user is focusing on location and will probably click somewhere near the location in (x_t, y_t). If not, we assume the user is focusing on mark color and will click on a mark of the color k_t. Specifically, we define:

$$p(c_t \mid z_t, \sigma_x, \sigma_y) = \pi \mathcal{N}(x'_t; x_t, \sigma_x^2)\mathcal{N}(y'_t; y_t, \sigma_y^2) + (1 - \pi)\mathcal{U}(k'_t; k_t),$$

where $\mathcal{U}(k'_t; k_t)$ denotes a uniform distribution over the available marks of color k_t. This above model therefore assumes that that if the user is interested in position (with probability π_t), she will click on a position on the scatter plot with probability proportional to a Gaussian distribution centered on (x_t, y_t) with diagonal covariance $[\sigma_x^2, 0; 0, \sigma_y^2]$. Again, we will specify these parameters.

Predicting Movements

To prediction movements, we can apply particle filter as described in Section 2.2.4. Based on the visual marks, we assume that a user performs one of three possible kinds of task at each time step. They are either interested in points based on their position on the scatter plot, or they are interested in points based on their color, or they are interest in the both position and color. When performing inference with our model, we expect that the inferred values of π will converge appropriately in each case to: 1 if the participant is doing a position-based task, 0 if they are performing an color-based task, and to a medial value in a mixed task.

Figure 2.4 shows an simulation of the algorithm when applied to a simple scatter plot. The simulated user begins by clicking on blue dots at the center of the projection, and within a few clicks, interest predictions converge to circles of the representative color with similar locations. At $t = 4$, the user selects a different color circle in the same region, and subsequent predictions update to include circles of different colors in a more tightly defined area.

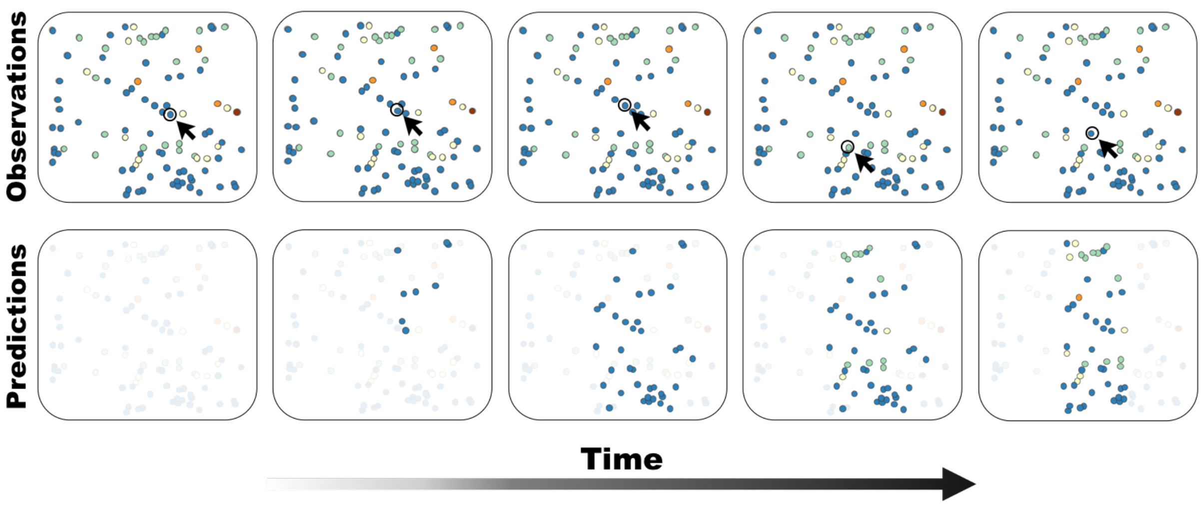

Figure 2.4: A simulation of the algorithm when applied to a simple scatter plot.

Chapter 3

Experiment

To test our approach, we designed a user study to track and analyze interactions. While we strive to create a method that is visualization agnostic, we choose a single visualization and dataset to reduce variability and complexity of the data collection.

3.1 Visualization

Figure 3.1 shows the experiment stimulus. We chose a map for our study because of its broad application and use. The dataset presented on the map were reported crimes in the city of St. Louis for March 2017 and that we gathered from the St. Louis Metropolitan Police Department's database [108]. The dataset contained 20 features and 1951 instances of reported crime with eight different categories: Homicide, Theft-Related, Assault, Arson, Fraud, Vandalism, Weapons, and Vagrancy.

To visualize the crime instances, we used a single visual mark (we represented each crime as a circle on the map). The visual channels used were position and color which denoted the location and type of crime respectively. To separate intentional from unintentional interaction, users interacted with the map by clicking on crime instances which triggered a tooltip displaying information about the type of crime and when it occurred.

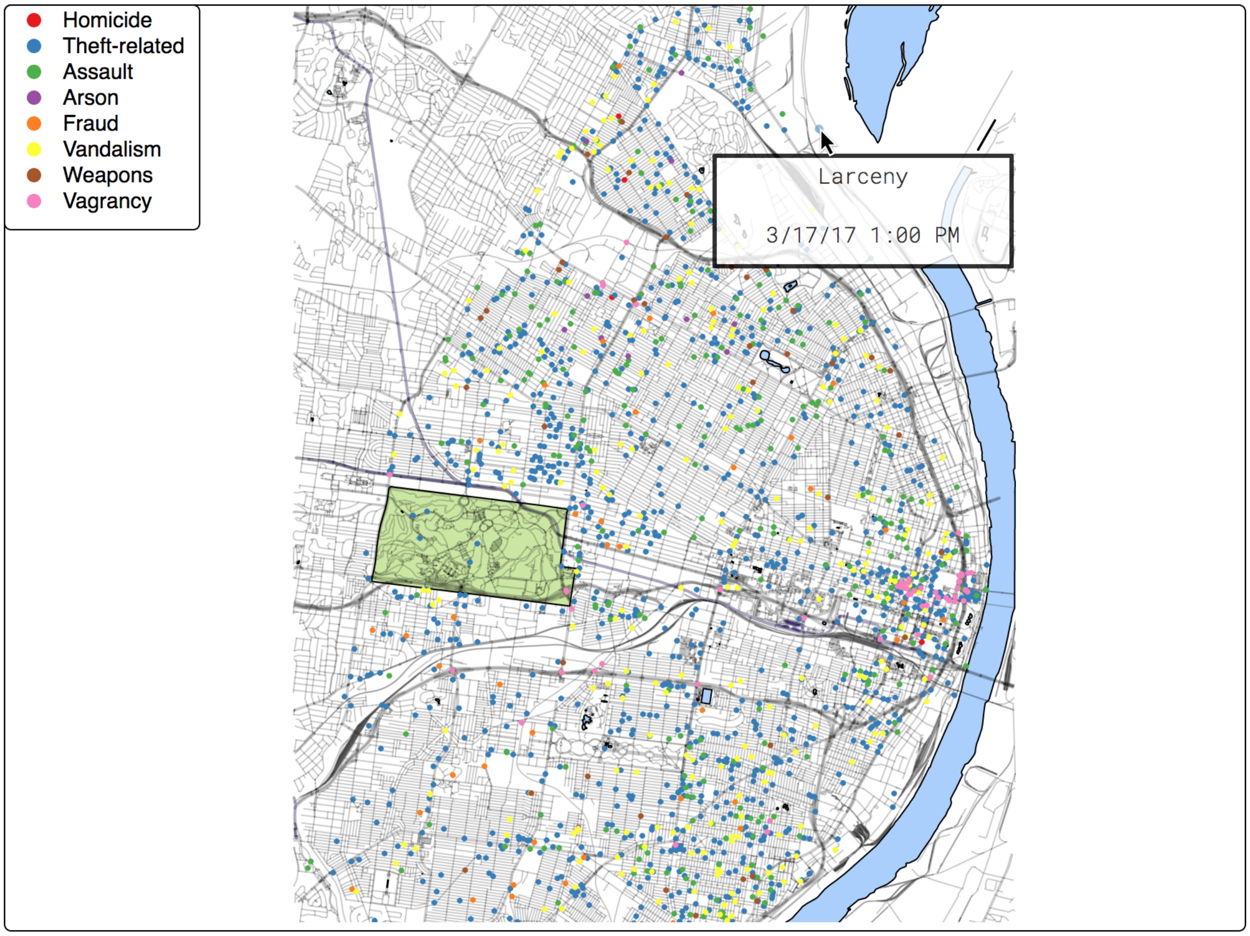

Figure 3.1: The interface used in our experiment. Participants used their mouse to pan and zoom the map. A tooltip displayed information about the crimes on click.

3.2 Participants

We recruited 30 participants via Amazon's Mechanical Turk. Participants were 18 years or older and were from the United States. Each participant had a HIT approval rate greater than 90% with at least 50 approved HITs. We paid a base rate of $1.00, an additional $0.50 for every correct answer plus $1.00 for each of the two optional post-surveys they completed. The maximum reward was $6.00.

There were 17 women and 13 men in our subject pool with ages ranging from 21 to 56 years ($\mu = 33.5$ and $\sigma = 10$). Sixty percent of the participants self-reported to have at least a college education.

Interest	Task	No. of instances
Geo-Based	Count the number of crimes that occurred during 7:00 AM and 12:30 PM in the red-shaded region.	29
	Count the number of crimes that occurred during AM in the red-shaded region.	43
Type-Based	One case of Homicide differs from the others with regard to time. What is the time of that case?	5
	How many cases of Arson occurred during PM?	14
Mixed	There are four types of Theft Related Crimes: Larceny, Burglary, Robbery, and Motor Vehicle Theft. Count the number of cases of Robbery in the red-shaded region.	85
	There are two types of Assault: Non-Aggravated and Aggravated. Count the number of Non-Aggravated Assaults in the red-shaded region.	37

Table 3.1: The tasks used in our experiment.

3.3 Task

In the main portion of the study, participants interacted with the crime map through panning, zooming and clicking to complete six search tasks and their associated question (see table 3.1). We divided these questions into three different task conditions. The three question types were meant to represent simple lookup tasks for which the participant had to consult the visualization. The questions were simplified versions of real-world tasks that represented a potential interest:

- **Geo-Based**: Different types of crime that are constrained to a specific geographical region.
- **Type-Based**: Same types of crime across the entire map.
- **Mixed**: Same types of crime **and** constrained to a specific geographic region.

The Geo-Based questions asked the participants to count the number of crimes within a specified geographical location that had a specific property. For example, *"Count the number of crimes that occurred during AM in the red-shaded region."* Participants clicked on every crime instance (a total of 43 dots) in the specified region (see figure 3.2 below). They then chose their response from a series of multiple choice options.

Unlike the Geo-Based questions, the Type-Based tasks were not bounded to a specific region. These questions required participants to explore the entire map and search for a specified category of crime. For instance, *"How many cases of Arson occurred during PM?"*. To answer the question correctly, the participant would click on each instance of Arson (a total of 14 violet dots) to count the number of cases that occurred during PM.

For Mixed tasks, participants interacted with points of the same category of crime in a specified area. For example, *"There are four types of Theft Related Crimes: Larceny, Burglary, Robbery, and Motor Vehicle Theft. Count the number of cases of Robbery in the red-shaded*

Figure 3.2: An example of a map shown for Geo-Based and Mixed tasks.

region.". Participants click on blue dots in the red-shaded area to reveal the tooltip (a total of 85 dots) and recorded the instances of Robbery.

While we used the same dataset throughout the experiment, each task focused on a different area of the map and a different type of crime. To correctly answer the questions, the design of the task required the participant to click on every valid point in the dataset. This was done to ensure a reasonably rich and large interaction dataset. Table 3.1 summarizes the tasks used in the study and well as the number of points participants interacted with for each.

3.4 Procedure

After selecting the task on Mechanical Turk, participants consented per [redacted for anonymity] IRB protocol. They then read the instructions for the study.

The main portion of the study began with a short video demonstrating the features of the interface. Specifically, we showed instructions for panning and zooming, and how to activate the tooltip. The participant then completed the six search tasks and entered their answers for each by selecting the appropriate multiple choice response. The order of the six tasks was counterbalanced to prevent ordering effects. Once the tasks were done, they completed two optional surveys (a paper folding task that measured spatial ability and a personality survey) then a short demographic questionnaire.

3.5 Data Collection and Cleaning

During the experiment, we recorded every mouse click event. We tracked the data point, its coordinates and a timestamp for the mouse event. Each participant completed 6 tasks (see table 3.1), resulting in 180 trials. To ensure the best quality data for our analysis, we filtered participants with incorrect answers. We further removed tasks with less than five mouse click events. After cleaning, 78 trials remained (23, 27, and 28 trials for Type-Based, Mixed and Geo-Based tasks respectively).

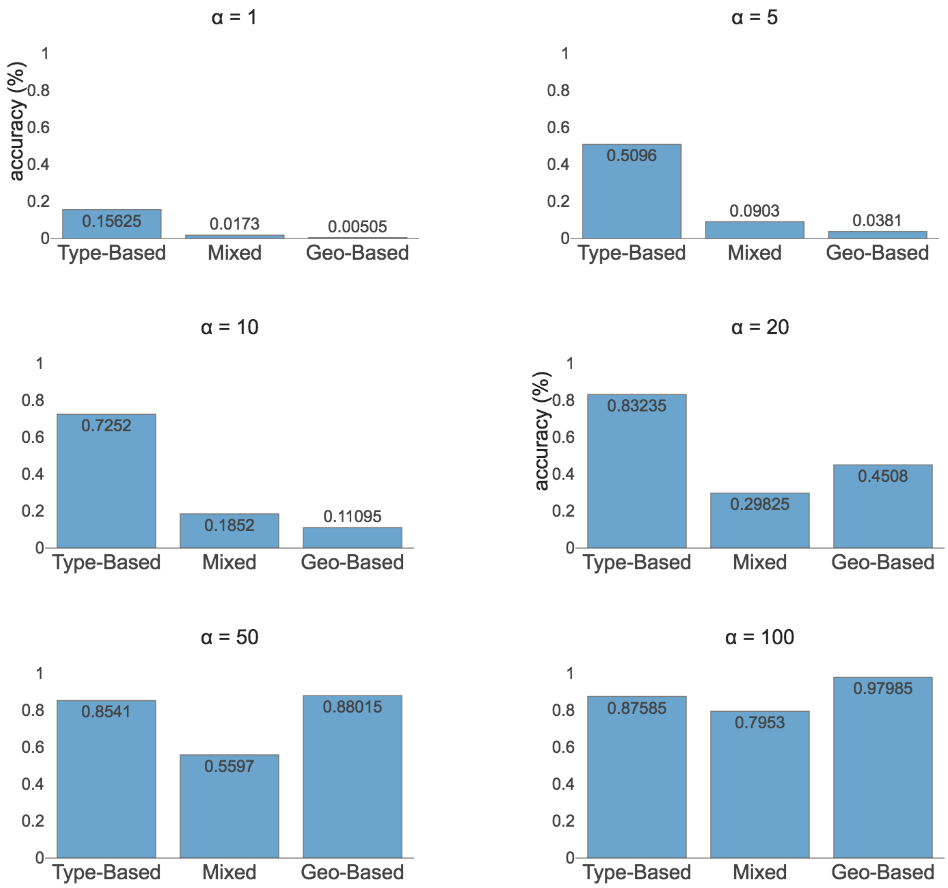

Figure 3.3: The average prediction accuracy across the three type of tasks (Geo-Based, Type-Based, and Mixed) for varying values of the prediction set size, α. For $\alpha = 100$ our algorithm successfully predicted the users' next click, on average, 88% of the times. The technique was particularly effective at predicting for Type-Based tasks, even for small values of α. For $\alpha = 10$, the algorithm achieves an accuracy of 72% for type-based tasks.

3.6 Applying the Model

3.6.1 Defining Unobservable and Observable States

We represented each click as a three-dimensional vector $c_t = (x'_t, y'_t, k'_t)$, where (x'_t, y'_t) is the latitude and longitude of the click and k'_t is the type of crime clicked, represented by a discrete integer-valued index ranging over possible values from the set $K = \{1, 2, \cdots, 7\}$. Each point in this hidden space is a vector specifying (1): a location (x, y) of interest, (2):

a crime type k of interest, and (3): a trade-off parameter indicating the relative importance of location and crime type. To specify the tradeoff parameter, we use a number $\pi \in [0, 1]$, with 1 indicating a complete focus on location and 0 meaning full focus on the type of crime. Therefore, a point in this latent intent space is a four-dimensional vector $z = (x, y, k, \pi)$.

3.6.2 Dynamical and Observation Model

Similar to scatter plot example in Section 2.3, we adopt a simple stationary diffusion model for the dynamical model of the hidden state $p(z_t \mid z_{t-1})$. We assume that intent does not change rapidly over time and that the four components of the intent vector evolve independently. Our implementation for the study replicates the dynamical and observation models described in Sections 2.3.2 and 2.3.3. For the location diffusion parameters, we took $s_x = s_y = 0.02$, where this value was as a fraction of the width and height of the map.

3.6.3 Predicting Movement

To predict movements, we applied particle filter as described in Section 2.2.4. Based on the experiment design, we assume that a user performs one of the three possible kinds of tasks at each time step: Geo-Based, Type-Based or Mixed. While the choice for α (the size of the prediction set) can be adapted for the application, our goal was to have a prediction set that is small, relative to the size of the dataset. We set $\alpha = 100$ which represents 5% of the dataset used in the study. This means that for a given timestep t, the algorithm chooses 100 points with the highest likelihood of being clicked at $t + 1$. For our evaluation, we also considered smaller values of α, namely 1, 5, 10, 20, and 50.

The parameters were set as $\sigma_x = \sigma_y = 0.1, \sigma_\pi = 0.05$. The location scale parameters were again a fraction of the width and height of the map. The probability of maintaining the same type of crime as the users' intent q was defined to be 0.95.

3.7 Results

3.7.1 Prediction Accuracy

After gathering the data, we analyzed our model's ability to observe mouse clicks and predict interactions before they occur. For each type of task (Geo-Based, Type-Based, and Mixed) we measured the overall predictive accuracy across all available clicks for all users. To allow time for the algorithm to learn users' intent, we begin our predictions at $t = 3$.

Figure 3.3 shows the model's accuracy across different values of α (1, 5, 10, 20, 50, and 100) for each of the three tasks. Overall, we found that the model performs particularly well for type-based tasks, even with small prediction sets. For instance, when limited to only five guesses for the next click, we see an average accuracy of 50% across all participants for the type-based task. The average accuracy increases to 72% when the set size is 10 (0.005% of the size of the full dataset).

Unsurprisingly, we see a direct correlation between accuracy and the size of the prediction sets. For $\alpha = 100$ (5% of the dataset), our technique attained an average of 88% at predicting the users' next clicks across all three task type ($\mu = 97.8\%$ for Geo-Based, $\mu = 87.5\%$ for Type-Based, and $\mu = 79.53\%$ for Mixed tasks). In other words, with high accuracy, we can predict that the next click will be within a small set of data points, relative to the dataset.

3.7.2 Accuracy Over Time

Our second analysis sort to evaluate our algorithm's performance as a function of the number of clicks observed. For each type task, we measure the prediction accuracy (set size = 100) for the first 20 clicks observed. We split the first 20 clicks into 4 observation windows. Each window has five predictions and we average the prediction accuracy within each window. Consistent with our previous analysis, we begin our predictions at $t = 3$. Figure 3.4 summarizes our findings. Our analysis reveals that the technique promptly achieves high prediction accuracies and performance remains constant with more observations.

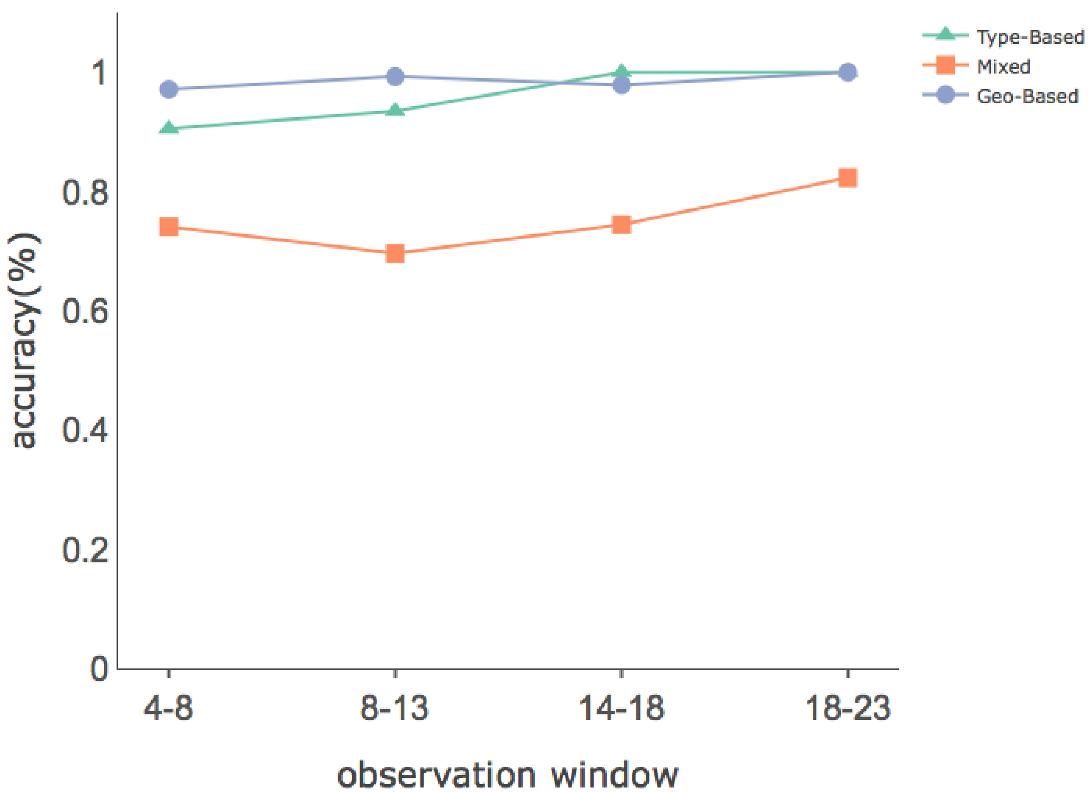

Figure 3.4: The average accuracy over time for the three types of tasks in the study. After learning from 3 click interactions, the algorithm immediately achieves high prediction accuracy. We found that prediction accuracies remain fairly constant over time.

3.8 Concatenating Trials

As we have shown previously, our model works fairly well for all three different types of tasks. While these tasks were designed based on realistic scenarios, they assume that the user has a specific and unchanging goal when they interact with the visualization. It's possible that user's interest change while interacting with the data. Therefore, a question arises: is the model robust enough to handle such a change in intention?

Target \ Prior	Type-Based	Mixed	Geo-Based
Type-Based	0.0258	0.0516	0.0710
Mixed	0.0132	0.0132	0.0132
Geo-Based	0.0020	0.0068	0.0078

Table 3.2: Accuracy of target trial when $\alpha = 1$

Target \ Prior	Type-Based	Mixed	Geo-Based
Type-Based	0.3161	0.4258	0.4129
Mixed	0.0680	0.0662	0.0788
Geo-Based	0.0224	0.0351	0.0205

Table 3.3: Accuracy of target trial when $\alpha = 5$

3.8.1 Prediction Accuracy

We conducted a further analysis on the user data we collected. Our goal is to see how our model can infer the new intention from a different prior belief. To be more specific, For each trial (target trial) in specific type, we attach interactions of a trial from a different task (prior trial) to it . We first run our algorithm on the prior trial. After that, without resetting our belief, we apply our model directly to the target trial and calculate the prediction accuracy.

The way we decide how to concatenate task is by the task type. For instance, for a type-based target trial, we then pick three prior trials from all three types and conduct three separate analysis. Notice that the prior trial and target trial should come from two separate tasks because the beliefs for the same task are almost identical. When prior trial and target trial have the same task type, they have to come from different task of that type (recall that we have two tasks for each type).

Note in Table 3.2 - 3.7, each accuracy is only the accuracy of target trial instead of the accuracy of concatenated trial. And just like in Section 3.7.1, we begin our prediction 3 clicks into the trial.

From these results, we can tell that concatenating tasks have almost no effect on type-based task. However, Geo-Based tasks are most susceptible to concatenation. Our interpretation

Prior / Target	**Type-Based**	**Mixed**	**Geo-Based**
Type-Based	0.6903	0.7355	0.6581
Mixed	0.1480	0.1432	0.1637
Geo-Based	0.0566	0.0771	0.0400

Table 3.4: Accuracy of target trial when $\alpha = 10$

Prior / Target	**Type-Based**	**Mixed**	**Geo-Based**
Type-Based	0.8387	0.9161	0.8258
Mixed	0.2659	0.2443	0.2768
Geo-Based	0.1746	0.2449	0.1454

Table 3.5: Accuracy of target trial when $\alpha = 20$

Prior / Target	**Type-Based**	**Mixed**	**Geo-Based**
Type-Based	0.8903	0.9355	0.8387
Mixed	0.5090	0.4290	0.5054
Geo-Based	0.4263	0.5415	0.4537

Table 3.6: Accuracy of target trial when $\alpha = 50$

Prior / Target	**Type-Based**	**Mixed**	**Geo-Based**
Type-Based	0.8903	0.9355	0.8452
Mixed	0.6925	0.5776	0.6504
Geo-Based	0.5424	0.6420	0.6322

Table 3.7: Accuracy of target trial when $\alpha = 100$

is in our model it's easier for the type of a particle to flip than for a particle to move from a location to another location that's far away.

3.8.2 Accuracy Over Time

Similar to Section 3.7.2, we evaluate our algorithm's performance as a function of the number of clicks observed on concatenated tasks. For each type task, we measure the prediction accuracy (set size = 100) from the beginning of the target trial. The prior trial only serves as a distraction, therefore it's accuracy will not be recorded. Consistent with our previous analysis, we discard our first three predictions. Figure 3.5 - 3.7 summarizes our findings. Our analysis reveals that our model have a smooth transition between tasks.

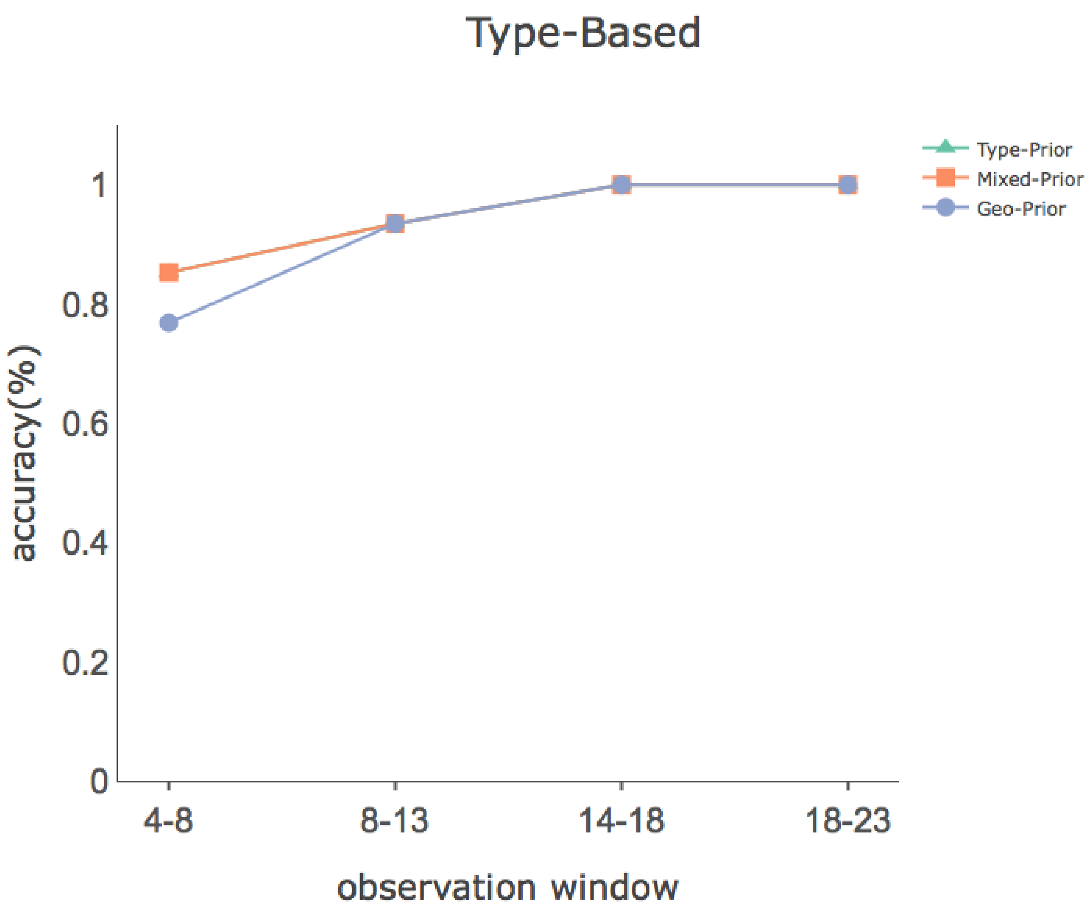

Figure 3.5: The average accuracy over time for Type-Based task after concatenation. We can see that, unlike unconcatenated tasks, concatenated Type-Based tasks need some more time to correct the prior belief. However, they can still achieve high prediction accuracies quickly.

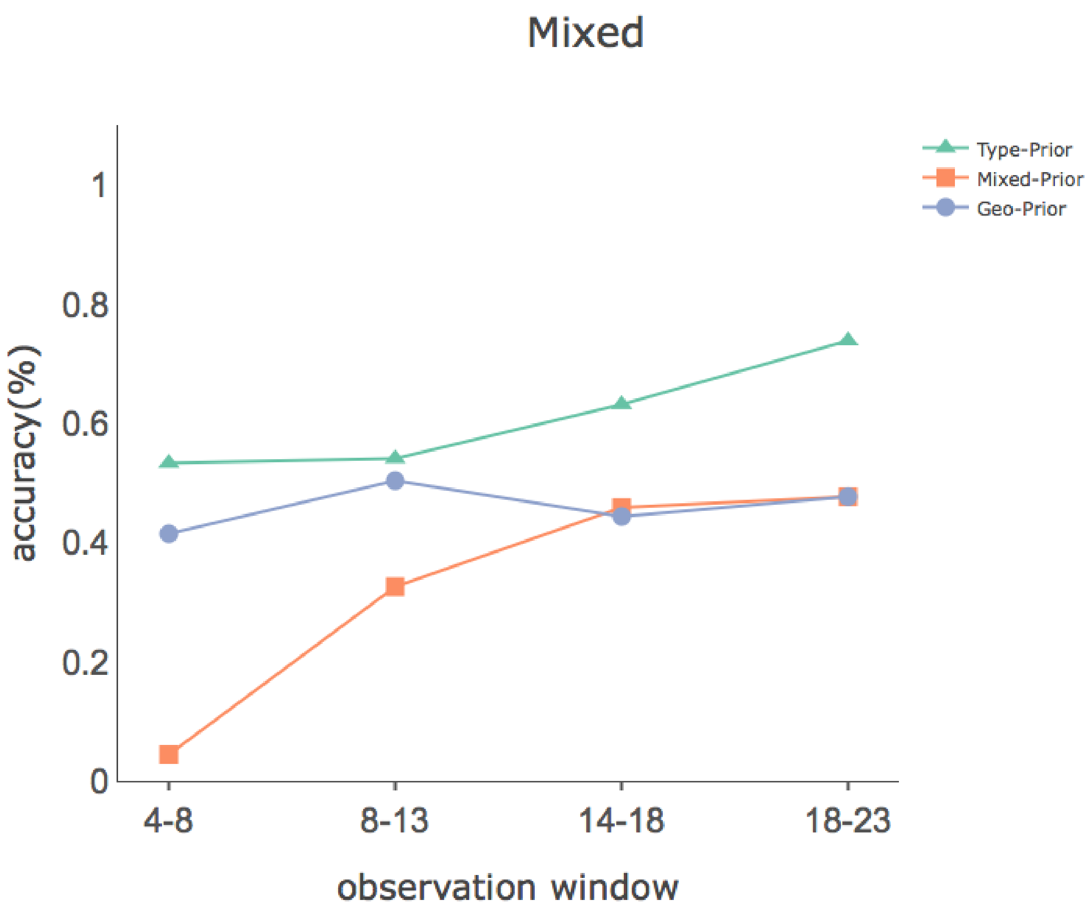

Figure 3.6: The average accuracy over time for Mixed task after concatenation. Concatenated Mixed tasks need some more time to correct the prior belief.

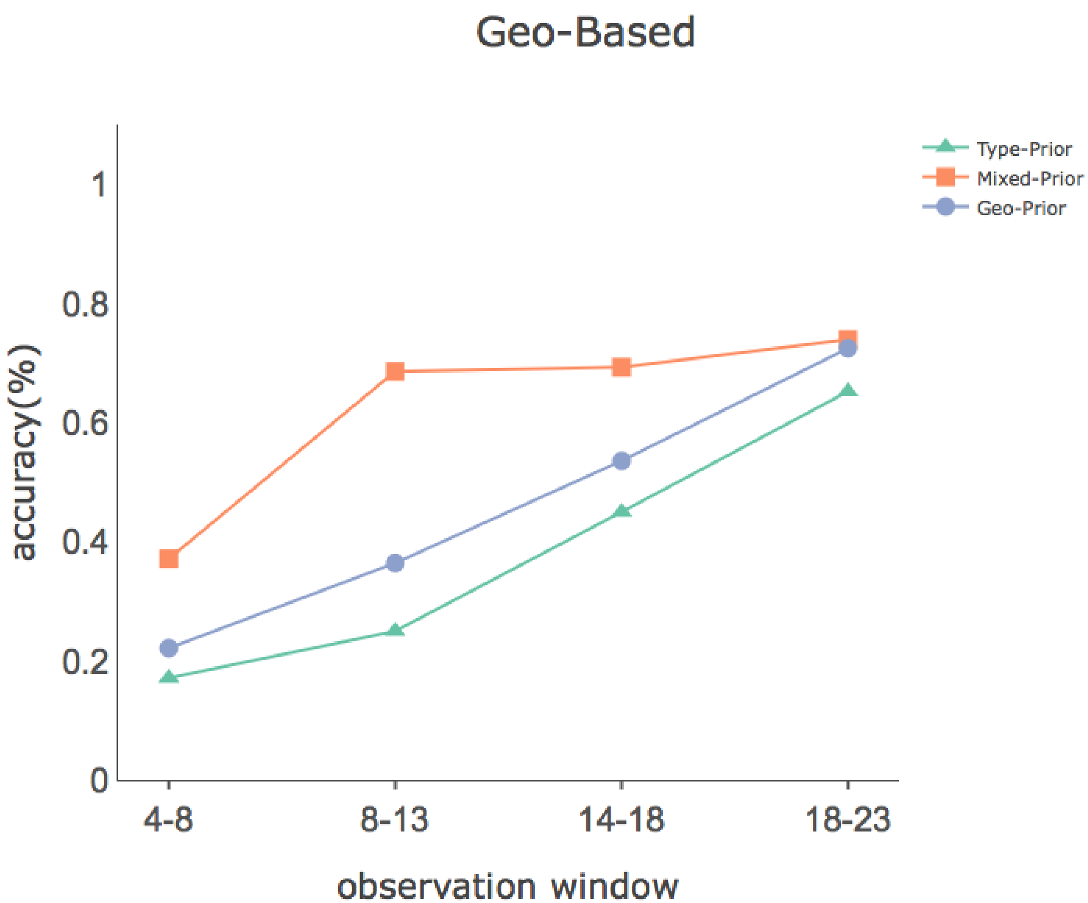

Figure 3.7: The average accuracy over time for Geo-Based task after concatenation. Concatenated Geo-Based tasks need even more time to correct the prior belief than other concatenated tasks because they don't reach stable accuracies by the end of 20 clicks.

Chapter 4

Discussion

In this thesis, we introduce a hidden Markov model approach for representing a user's interest and for predicting future, unobservable interests by passively monitoring their actions. The foundation of this approach is not novel. The hidden Marvok model is widely used for modeling sequence data in areas such as natural language processing [79], speech recognition [71, 97], and biological sequencing [36, 106]. However, we demonstrate its utility for modeling interest from interaction with a visualization system.

Our goal was to create a model that can be generalized to any visualization design. We leverage data mapping principles and the notion that we can represent a visualization as a set of primitive visual marks and channels. We argue researchers and designers can apply the approach to any visualization that can be specified in this manner. The model assumes that the visual marks are perceptually differentiable, and relies heavily on good design practices. To specify a user's interests and actions, we must first carefully define the mark space, $\mathcal{M}$. One way to improve this process is to automatically extract the visual marks and channels from the visualization's code. However, this is beyond the scope of the thesis.

We used a map visualization and real-world crime dataset to evaluate the feasibility of the technique. By applying our model, we successfully predicted actions before they occurred by observing users' click behavior. For a prediction set of 100 ($\alpha = 100$), we achieved an overall accuracy of 88%. For $\alpha = 100$, the algorithm was most accurate at predicting clicks when users performed *Geo-Based* tasks that required them to explore points in the particular region of the map. It is also notable that the technique attained high prediction accuracies for *Type-Based* tasks even when α was small. For instance, when we limited the algorithm to only ten guesses for the users' next click (effectively reducing the set of possible points of interest by 99.99%) the average accuracy across all users was 72%.

Our analysis also showed that the approach was successful in making inferences in a short period. After allowing a grace period of three observations for the algorithm to learn users' interest, we found that the technique attained high accuracy at the first prediction of z_{t+1}. Furthermore, we observed little change in accuracy over time. Lastly, our model is able to sustain a fairly high prediction accuracy for two type of tasks after the interference of a prior trial. This finding demonstrates the potential of applying the approach to model interactions for real-time applications.

4.1 Design Implications

The overall goal of this work is to create tools that can automatically learn user's intent and support them throughout the analysis process. The idea of tailoring an interface based on users' skills or needs has existed for many years in HCI [49], and researchers have explored the tradeoff between providing support and minimizing disruptions [1, 89, 105, 115]. For large datasets that may have overlapping points (see figure 2.3), a straightforward approach can be to redraw the points in the prediction set. Doing so can make it easier for users to interact with points that match their interests but may have initially been occluded by other visual marks. For more passive adaptations, designers can use the approach in this thesis to inform techniques for target assistance [5]. The *bubble cursor* technique, for example, do not change the visual appearance of the interface but increases the click radius for the given target, thereby making them more accessible [61]. Another possibility is *target gravity* which attracts the mouse to the target [5].

The hidden Markov model is a general framework with many possible variations for the model, the implementation, and parameters settings. Examples include choices for the diffusion parameters, number of particles for the particle filter, and prediction set sizes. These parameters can be tuned or customized based on the visualization or tasks and designers can apply the technique to a wide range of applications. We see this as a strength of the approach which can seed many opportunities for future work.

4.2 Potential Application

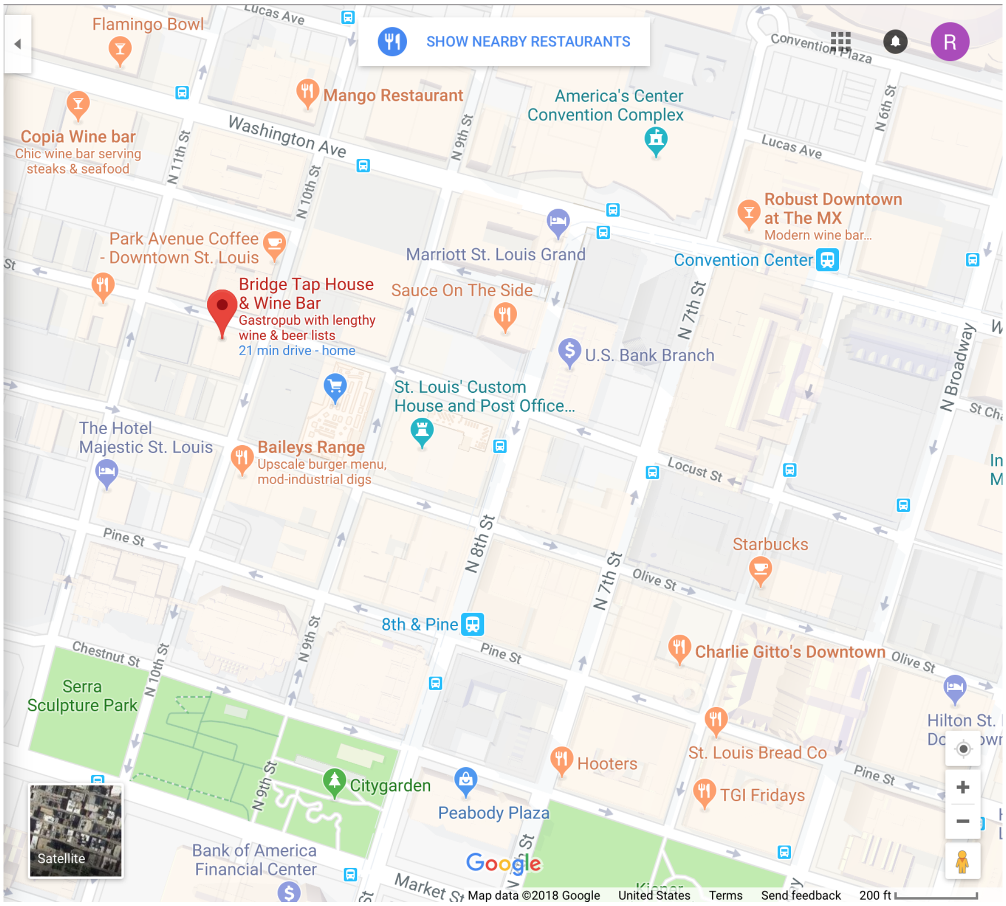

Figure 4.1: When user clicks a bar, more bars in the area will show up, some coffee shops disappear from the map.

When a user interacts with a visualization, especially for a map visualization, he or she usually has a search goal in mind. Our model has the potential to improve many existing visualizations. To be more specific, by inferring user's intent, the system can automatically prompt potential options, pre-fetching data points that are likely to be interacted later, obscure irrelevant visual elements.

Most map visualization nowadays don't have any adaptive feature. Even for Google Map, it only has some simple adaptive features. For example, if a user clicks on a restaurant, Google Map thinks the user is interested in restaurants in the area. Therefore, it will automatically display more restaurants in the area. This is a very user-friendly feature. However, this is

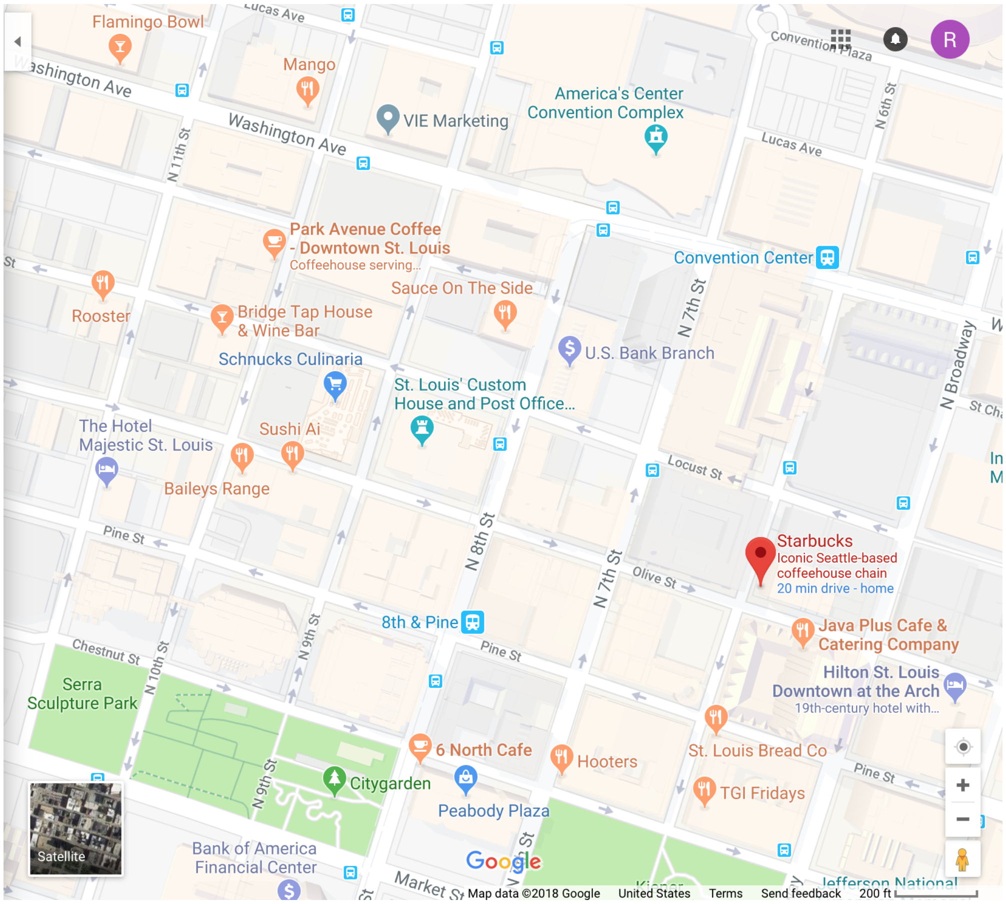

Figure 4.2: When user clicks Starbucks, more coffee shops in the area will show up , some bars disappear from the map.

actually a quite naive algorithm and it may not have the capability to handle more complex tasks. For instance, if the user is interested in both bar and coffee shop, the behavior will mostly resemble that in Figure 4.1 and Figure 4.2. No matter how many times you click on bars and coffee shops, the system will always assume you are only interest in one category - emphasizing one and obscuring the other. In the mean time, many hotels still shows up on the map even if the user never pays any attention to hotels.

Since our model use particle filtering to monitoring the interaction on the fly, it would be much more intelligent. If we apply our model to this problem, it's likely that after a couple of clicks, a substantial portion of the particles will either be related to bar or related to coffee shop. As a result, more helpful information will be displayed and more irrelevant information will be discarded.

In conclusion, our model has the potential to make interfaces like these a lot more intelligent.

Chapter 5

Future Work

Our preliminary work lays the foundation for creating next-generation visual analytics systems that can automatically learn users intentions to support the analysis process. We see this as the beginning of a new era for visual analytics systems, and we believe that the work in the thesis opens possibilities for future work toward more generalizable techniques for real-world systems.

5.1 Complex Tasks

One possible path for future work is to explore the model's performance for more complex tasks. In our experiment, we controlled the tasks by instructing participants to either search for a specific reported crime or identify a pattern in the dataset. However, the search patterns we observed may not generalize to scenarios where the goal is to explore the dataset. Is it also possible that there are some tasks that cannot be represented at as subspace of the visualization marks. Future work can evaluate the approach with exploratory and open-ended tasks.

5.2 Proof of Generalizability

The combination of visual marks and channels is an essential factor when defining the hidden state space for our probabilistic model. The map used in our experiment was simplistic compared to other real-world visual analytics systems. Future work can test the model using different combinations of visual variables and channels on a single map, or an entirely different type of visualization. Varying the complexity of the visualization will test the generalizability of the model. Future work can also vary the number of data points. For our proof-of-concept experiment, we used only reported crimes for March 2017, and we minimized the number of visual variables by grouping similar types of crimes to create categories (e.g., aggravated and non-aggravated assaults were consolidated to form the *Assault* crime category). It is essential to validate the technique by changing and increasing the size of the dataset, which can result in the drastic changes in the appearance and number of visual marks.

5.3 Adaptive Visualization

One of our end goals is to use our model to create visualization that better support users. We can proceed with our crime map case study and design possible adaptive features to help user finish tasks. We can explore whether these features indeed help users obtain more correct answers, faster speed and more pleasing interaction experience.

Chapter 6

Conclusion

In this thesis, we have proposed a generalizable and design-agnostic approach to modeling users' interests and actions with a visualization system. We used a hidden Markov model and represented users interests based on the primitive visual marks and channels of the visualization design. We demonstrate, with a simple scatter plot, how to apply this approach to a given visualization design.

To evaluate this technique, we conducted a user study and captured interaction data as participants explored a map showing a real-world crime dataset. The results of the study demonstrate that the approach is highly successful at modeling interaction and predicting users' next clicks. We observed an overall accuracy of 88% at guessing actions before they occur. These results are exciting and contribute to our overall goal of creating intelligent systems that learn about the user, her analysis process, and her task as she uses the system. We believe that the work in this thesis is a significant step toward this goal and can act as a catalyst for future work aimed at developing visual analytic systems that can better support users.

References

[1] Daniel Afergan, Evan M Peck, Remco Chang, and Robert JK Jacob. Using passive input to adapt visualization systems to the individual. 2013.

[2] Hirotugu Akaike. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723, 1974.

[3] Bryce Allen. Individual differences and the conundrums of user-centered design: Two experiments. *Journal of the american society for information science*, 51(6):508–520, 2000.

[4] Gennady Andrienko, Natalia Andrienko, Michael Burch, and Daniel Weiskopf. Visual analytics methodology for eye movement studies. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 18(12):2889–2898, 2012.

[5] Scott Bateman, Regan L Mandryk, Tadeusz Stach, and Carl Gutwin. Target assistance for subtly balancing competitive play. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2355–2364. ACM, 2011.

[6] Roberto Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on neural networks*, 5(4):537–550, 1994.

[7] Leilani Battle, Remco Chang, and Michael Stonebraker. Dynamic prefetching of data tiles for interactive visualization. In *Proceedings of the 2016 International Conference on Management of Data*, pages 1363–1375. ACM, 2016.

[8] Louis Bavoil, Steven P Callahan, Patricia J Crossno, Juliana Freire, Carlos E Scheidegger, Cláudio T Silva, and Huy T Vo. Vistrails: Enabling interactive multiple-view visualizations. In *Visualization, 2005. VIS 05. IEEE*, pages 135–142. IEEE, 2005.

[9] Jacques Bertin. Semiology of graphics: diagrams, networks, maps. 1983.

[10] Avrim L Blum and Pat Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1):245–271, 1997.

[11] Jeremy Boy, Francoise Detienne, and Jean-Daniel Fekete. Storytelling in information visualizations: Does it engage users to explore data? In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 1449–1458. ACM, 2015.

[12] Ken Brodlie, Andrew Poon, Helen Wright, Lesley Brankin, Greg Banecki, and Alan Gay. Grasparc: a problem solving environment integrating computation and visualization. In *Proceedings of the 4th conference on Visualization'93*, pages 102–109. IEEE Computer Society, 1993.

[13] Eli T Brown, Jingjing Liu, Carla E Brodley, and Remco Chang. Dis-function: Learning distance functions interactively. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 83–92. IEEE, 2012.

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

[15] Michael Buhrmester, Tracy Kwang, and Samuel D Gosling. Amazon's mechanical turk: A new source of inexpensive, yet high-quality, data? *Perspectives on psychological science*, 6(1):3–5, 2011.

[16] Steven P Callahan, Juliana Freire, Emanuele Santos, Carlos E Scheidegger, Cláudio T Silva, and Huy T Vo. Vistrails: visualization meets data management. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 745–747. ACM, 2006.

[17] Stuart Card, JD Mackinlay, and B Shneiderman. Information visualization. *Human-computer interaction: Design issues, solutions, and applications*, 181, 2009.

[18] Stuart K Card, Jock D Mackinlay, and Ben Shneiderman. *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.

[19] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[20] Chaomei Chen. Individual differences in a spatial-semantic virtual environment. *Journal of the American Society for Information Science*, 51(6):529–542, 2000.

[21] Chaomei Chen and Mary Czerwinski. Spatial ability and visual navigation: An empirical study. *New Review of Hypermedia and Multimedia*, 3(1):67–89, 1997.

[22] Min Chen, David Ebert, Hans Hagen, Robert S Laramee, Robert Van Liere, Kwan-Liu Ma, William Ribarsky, Gerik Scheuermann, and Deborah Silver. Data, information, and knowledge in visualization. *IEEE Computer Graphics and Applications*, 29(1), 2009.

[23] Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. Click models for web search. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 7(3):1–115, 2015.

[24] Cheryl A Cohen and Mary Hegarty. Individual differences in use of external visualisations to perform an internal visualisation task. *Applied Cognitive Psychology*, 21(6):701–711, 2007.

[25] Cristina Conati and Heather Maclaren. Exploring the role of individual differences in information visualization. In *Proceedings of the working conference on advanced visual interfaces*, pages 199–206. ACM, 2008.

[26] Kristin A Cook and James J Thomas. Illuminating the path: The research and development agenda for visual analytics. 2005.

[27] Lynne Cooke. Is the mouse a "poor man's eye tracker"? In *Annual Conference-Society for Technical Communication*, volume 53, page 252, 2006.

[28] Paula Cowley, Lucy Nowell, and Jean Scholtz. Glass box: An instrumented infrastructure for supporting human interaction with information. In *System Sciences, 2005. HICSS'05. Proceedings of the 38th Annual Hawaii International Conference on*, pages 296c–296c. IEEE, 2005.

[29] CRA-W. https://cra.org/cra-w/dreu/. Accessed October 8, 2017.

[30] R Jordan Crouser, Alvitta Ottley, and Remco Chang. Balancing human and machine contributions in human computation systems. In *Handbook of Human Computation*, pages 615–623. Springer, 2013.

[31] Filip Dabek and Jesus J Caban. 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):41–50, 2017.

[32] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

[33] Wenwen Dou, Dong Hyun Jeong, Felesia Stukes, William Ribarsky, Heather Richter Lipford, and Remco Chang. Recovering reasoning processes from user interactions. *IEEE Computer Graphics and Applications*, 29(3), 2009.

[34] Arnaud Doucet, Simon Godsill, and Christophe Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, 10(3):197–208, 2000.

[35] Cody Dunne, Nathalie Henry Riche, Bongshin Lee, Ronald Metoyer, and George Robertson. Graphtrail: Analyzing large multivariate, heterogeneous networks while supporting exploration history. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1663–1672. ACM, 2012.

[36] Richard Durbin, Sean R Eddy, Anders Krogh, and Graeme Mitchison. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998.

[37] Jennifer G Dy and Carla E Brodley. Feature subset selection and order identification for unsupervised learning. In *ICML*, pages 247–254, 2000.

[38] Jennifer G Dy and Carla E Brodley. Visualization and interactive feature selection for unsupervised data. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 360–364. ACM, 2000.

[39] Jennifer G Dy and Carla E Brodley. Feature selection for unsupervised learning. *Journal of machine learning research*, 5(Aug):845–889, 2004.

[40] David M Eddy. Probabilistic reasoning in clinical medicine: Problems and opportunities. 1982.

[41] Magdalini Eirinaki and Michalis Vazirgiannis. Web mining for web personalization. *Transactions on Internet Technology (TOIT)*, 3(1):1–27, 2003.

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

[43] Alex Endert, Patrick Fiaux, and Chris North. Semantic interaction for visual text analytics. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 473–482. ACM, 2012.

[44] Imola K Fodor. A survey of dimension reduction techniques. Technical report, Lawrence Livermore National Lab., CA (US), 2002.

[45] George Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of machine learning research*, 3(Mar):1289–1305, 2003.

[46] Juliana Freire, David Koop, Emanuele Santos, and Cláudio T Silva. Provenance for computational tasks: A survey. *Computing in Science & Engineering*, 10(3), 2008.

[47] Juliana Freire, Cláudio T Silva, Steven P Callahan, Emanuele Santos, Carlos E Scheidegger, and Huy T Vo. Managing rapidly-evolving scientific workflows. In *International Provenance and Annotation Workshop*, pages 10–18. Springer, 2006.

[48] Eugene Yujun Fu, Tiffany CK Kwok, Erin You Wu, Hong Va Leong, Grace Ngai, and Stephen CF Chan. Your mouse reveals your next activity: Towards predicting user intention from mouse interaction. In *Computer Software and Applications Conference (COMPSAC), 2017 IEEE 41st Annual*, volume 1, pages 869–874. IEEE, 2017.

[49] Krzysztof Gajos and Daniel S Weld. Supple: automatically generating user interfaces. In *Proceedings of the Ninth International Conference on Intelligent User Interfaces*, pages 93–100. ACM, 2004.

[50] Krzysztof Z Gajos, Daniel S Weld, and Jacob O Wobbrock. Decision-theoretic user interface generation. In *AAAI*, volume 8, pages 1532–1536, 2008.

[51] Krzysztof Z Gajos, Jacob O Wobbrock, and Daniel S Weld. Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1257–1266. ACM, 2008.

[52] Rocio Garcia-Retamero and Mirta Galesic. Who profits from visual aids: Overcoming challenges in people's understanding of risks. *Social science & medicine*, 70(7):1019–1025, 2010.

[53] Supriya Garg, Julia Eunju Nam, IV Ramakrishnan, and Klaus Mueller. Model-driven visual analytics. In *Visual Analytics Science and Technology, 2008. VAST'08. IEEE Symposium on*, pages 19–26. IEEE, 2008.

[54] James J Gibson. *The ecological approach to visual perception: classic edition*. Psychology Press, 2014.

[55] Gerd Gigerenzer and Ulrich Hoffrage. How to improve bayesian reasoning without instruction: Frequency formats. *Psychological review*, 102(4):684, 1995.

[56] Girls Who Code. https://girlswhocode.com/. Accessed October 8, 2017.

[57] Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140, pages 107–113. IET, 1993.

[58] David Gotz, Michelle X Zhou, and Vikram Aggarwal. Interactive visual synthesis of analytic knowledge. In *Visual Analytics Science And Technology, 2006 IEEE Symposium On*, pages 51–58. IEEE, 2006.

[59] Liang Gou, Xiaolong Zhang, Airong Luo, and Patricia F Anderson. Socialnetsense: supporting sensemaking of social and structural features in networks with interactive visualization. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on*, pages 133–142. IEEE, 2012.

[60] Tear Marie Green and Brian Fisher. Towards the personal equation of interaction: The impact of personality factors on visual analytics interface interaction. In *Visual Analytics Science and Technology (VAST), 2010 IEEE Symposium on*, pages 203–210. IEEE, 2010.

[61] Tovi Grossman and Ravin Balakrishnan. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 281–290. ACM, 2005.

[62] Zhiwei Guan and Edward Cutrell. An eye tracking study of the effect of target rank on web search. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 417–420. ACM, 2007.

[63] Hua Guo, Steven R Gomez, Caroline Ziemkiewicz, and David H Laidlaw. 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):51–60, 2016.

[64] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[65] John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

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

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

[68] Jeff Huang, Ryen White, and Georg Buscher. User see, user point: gaze and cursor alignment in web search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1341–1350, New York, NY, USA, 2012. ACM.

[69] Petra Isenberg and Danyel Fisher. Pairgrams: Understanding collaborative analysis behavior with visualization. In *CHI Workshop on Analytic Provenance: Process+ Interaction+ Insight*, 2011.

[70] Waqas Javed and Niklas Elmqvist. Explates: spatializing interactive analysis to scaffold visual exploration. In *Computer Graphics Forum*, volume 32, pages 441–450. Wiley Online Library, 2013.

[71] Frederick Jelinek. *Statistical methods for speech recognition.* MIT press, 1997.

[72] Dong Hyun Jeong, Wenwen Dou, Heather Richter Lipford, Felesia Stukes, Remco Chang, and William Ribarsky. Evaluating the relationship between user interaction and financial visual analysis. In *Visual Analytics Science and Technology, 2008. VAST'08. IEEE Symposium on*, pages 83–90. IEEE, 2008.

[73] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. Unbiased learning-to-rank with biased feedback. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 781–789. ACM, 2017.

[74] Pranam Kolari and Anupam Joshi. Web mining: Research and practice. *Computing in science & engineering*, 6(4):49–53, 2004.

[75] Raymond Kosala and Hendrik Blockeel. Web mining research: A survey. *ACM SIGKDD Explorations Newsletter*, 2(1):1–15, 2000.

[76] Martin HC Law, Mario AT Figueiredo, and Anil K Jain. Simultaneous feature selection and clustering using mixture models. *IEEE transactions on pattern analysis and machine intelligence*, 26(9):1154–1166, 2004.

[77] Washington University Course Listings. https://acadinfo.wustl.edu/courselistings/semester/listing.as Accessed November 13, 2017.

[78] Aidong Lu, Ross Maciejewski, and David S Ebert. Volume composition and evaluation using eye-tracking data. *ACM Transactions on Applied Perception (TAP)*, 7(1):4, 2010.

[79] Christopher D Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT press, 1999.

[80] Geoffrey J McLachlan and Kaye E Basford. *Mixture models: Inference and applications to clustering*, volume 84. Marcel Dekker, 1988.

[81] Missouri Louis Stokes Alliance for Minority Participation (MoLSAMP). https://undergradresearch.missouri.edu/molsamp/. Accessed October 8, 2017.

[82] Phong H Nguyen, Kai Xu, and BL Wong. A survey of analytic provenance. *Middlesex University*, 2014.

[83] Chris North, Remco Chang, Alex Endert, Wenwen Dou, Richard May, Bill Pike, and Glenn Fink. Analytic provenance: process+ interaction+ insight. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pages 33–36. ACM, 2011.

[84] Alvitta Ottley, R Jordan Crouser, Caroline Ziemkiewicz, and Remco Chang. Manipulating and controlling for personality effects on visualization tasks. *Information Visualization*, 14(3):223–233, 2015.

[85] Alvitta Ottley, Evan M Peck, Lane T Harrison, Daniel Afergan, Caroline Ziemkiewicz, Holly A Taylor, Paul KJ Han, and Remco Chang. Improving bayesian reasoning: the effects of phrasing, visualization, and spatial ability. *IEEE transactions on visualization and computer graphics*, 22(1):529–538, 2016.

[86] Alvitta Ottley, Ran Wan, Surina Puri, and Roman Garnett. Inferring interest and intent from interactions. 2018. Manuscript in preparation.

[87] Alvitta Ottley, Huahai Yang, and Remco Chang. Personality as a predictor of user strategy: How locus of control affects search strategies on tree visualizations. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3251–3254. ACM, 2015.

[88] Daniel Paurat, Roman Garnett, and Thomas Gärtner. Interactive exploration of larger pattern collections: A case study on a cocktail dataset. *Proc. of KDD IDEA*, pages 98–106, 2014.

[89] Evan M Peck. *Designing Brain-Computer Interfaces for Intelligent Information Delivery Systems*. PhD thesis, Tufts University, 2014.

[90] Evan M Peck, Beste F Yuksel, Lane Harrison, Alvitta Ottley, and Remco Chang. Towards a 3-dimensional model of individual cognitive differences: Position paper. In *Proceedings of the 2012 BELIV Workshop: Beyond Time and Errors - Novel Evaluation Methods for Visualization*, BELIV '12, pages 6:1–6:6, New York, NY, USA, 2012. ACM.

[91] Evan M M Peck, Beste F Yuksel, Alvitta Ottley, Robert JK Jacob, and Remco Chang. Using fnirs brain sensing to evaluate information visualization interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 473–482. ACM, 2013.

[92] William Pike, Richard May, and Alan Turner. Supporting knowledge transfer through decomposable reasoning artifacts. In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*, pages 204c–204c. IEEE, 2007.

[93] William A Pike, John Stasko, Remco Chang, and Theresa A O'connell. The science of interaction. *Information Visualization*, 8(4):263–274, 2009.

[94] Peter Pirolli and Stuart Card. Information foraging in information access environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 51–58. ACM Press/Addison-Wesley Publishing Co., 1995.

[95] Peter Pirolli and Stuart Card. Information foraging. *Psychological review*, 106(4):643, 1999.

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

[97] Lawrence R Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition*, volume 14. PTR Prentice Hall Englewood Cliffs, 1993.

[98] Eric D Ragan, Alex Endert, Jibonananda Sanyal, and Jian Chen. Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes. *IEEE transactions on visualization and computer graphics*, 22(1):31–40, 2016.

[99] Jorma Rissanen. A universal prior for integers and estimation by minimum description length. *The Annals of statistics*, pages 416–431, 1983.

[100] Julian B Rotter. Generalized expectancies for internal versus external control of reinforcement. *Psychological monographs: General and applied*, 80(1):1, 1966.

[101] Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[102] Yedendra Babu Shrinivasan and Jarke J van Wijk. Supporting the analytical reasoning process in information visualization. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1237–1246. ACM, 2008.

[103] Pete B Shull, Wisit Jirattigalachote, Michael A Hunt, Mark R Cutkosky, and Scott L Delp. Quantified self and human movement: a review on the clinical impact of wearable sensing and feedback for gait analysis and intervention. *Gait & posture*, 40(1):11–19, 2014.

[104] Padhraic Smyth. Clustering using monte carlo cross-validation. In *Kdd*, volume 1, pages 26–133, 1996.

[105] Erin Treacy Solovey, Francine Lalooses, Krysta Chauncey, Douglas Weaver, Margarita Parasi, Matthias Scheutz, Angelo Sassaroli, Sergio Fantini, Paul Schermerhorn, Audrey Girouard, et al. Sensing cognitive multitasking for a brain-based adaptive user interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 383–392. ACM, 2011.

[106] Erik LL Sonnhammer, Gunnar Von Heijne, Anders Krogh, et al. A hidden markov model for predicting transmembrane helices in protein sequences. In *Ismb*, volume 6, pages 175–182, 1998.

[107] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan. Web usage mining: Discovery and applications of usage patterns from web data. *ACM SIGKDD Explorations Newsletter*, 1(2):12–23, 2000.

[108] St. Louis Metropolitan Police Department. `http://www.slmpd.org/crime_mapping.shtml`. Accessed November 13, 2017.

[109] Ben Steichen, Giuseppe Carenini, and Cristina Conati. User-adaptive information visualization: using eye gaze data to infer visualization tasks and user cognitive abilities. In *Proceedings of the 2013 international Conference on Intelligent User Interfaces*, pages 317–328. ACM, 2013.

[110] John Stutz and Peter Cheeseman. Autoclass-a bayesian approach to classification. *FUNDAMENTAL THEORIES OF PHYSICS*, 70:117–126, 1996.

[111] Melanie Swan. Emerging patient-driven health care models: an examination of health social networks, consumer personalized medicine and quantified self-tracking. *International journal of environmental research and public health*, 6(2):492–525, 2009.

[112] Melanie Swan. Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self 2.0. *Journal of Sensor and Actuator Networks*, 1(3):217–253, 2012.

[113] Melanie Swan. The quantified self: Fundamental disruption in big data science and biological discovery. *Big Data*, 1(2):85–99, 2013.

[114] Dereck Toker, Cristina Conati, Ben Steichen, and Giuseppe Carenini. Individual user characteristics and information visualization: Connecting the dots through eye tracking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 295–304. ACM, 2013.

[115] Erin Treacy Solovey, Daniel Afergan, Evan M Peck, Samuel W Hincks, and Robert JK Jacob. Designing implicit interfaces for physiological computing: guidelines and lessons learned using fnirs. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(6):35, 2015.

[116] Archie Tse. https://github.com/archietse/malofiej-2016/blob/ master/tse-malofiej-2016-slides.pdf, 2016.

[117] Maria C Velez, Deborah Silver, and Marilyn Tremaine. Understanding visualization through spatial ability differences. In *Visualization, 2005. VIS 05. IEEE*, pages 511–518. IEEE, 2005.

[118] Fernanda B Viegas, Martin Wattenberg, Frank Van Ham, Jesse Kriss, and Matt McKeon. Manyeyes: a site for visualization at internet scale. *IEEE transactions on visualization and computer graphics*, 13(6), 2007.

[119] Alan Wexelblat and Pattie Maes. Footprints: history-rich tools for information foraging. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 270–277. ACM, 1999.

[120] Ling Xiao, John Gerth, and Pat Hanrahan. Enhancing visual analysis of network traffic using a knowledge representation. In *Proceedings of the IEEE Symposium On Visual Analytics Science And Technology (VAST)*, pages 107–114. IEEE, 2006.

[121] Ji Soo Yi. Implications of individual differences on evaluating information visualization techniques. 2010.

[122] Ji Soo Yi, Youn ah Kang, and John Stasko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE transactions on visualization and computer graphics*, 13(6):1224–1231, 2007.

[123] Caroline Ziemkiewicz and Robert Kosara. Preconceptions and individual differences in understanding visual metaphors. In *Computer Graphics Forum*, volume 28, pages 911–918. Wiley Online Library, 2009.

[124] Caroline Ziemkiewicz, Alvitta Ottley, R Jordan Crouser, Krysta Chauncey, Sara L Su, and Remco Chang. Understanding visualization by understanding individual users. *IEEE computer graphics and applications*, 32(6):88–94, 2012.

[125] Caroline Ziemkiewicz, Alvitta Ottley, R Jordan Crouser, Ashley Rye Yauilla, Sara L Su, William Ribarsky, and Remco Chang. How visualization layout relates to locus of control and other personality factors. *IEEE transactions on visualization and computer graphics*, 19(7):1109–1121, 2013.