Eurographics Conference on Visualization (EuroVis) 2019
M. Gleicher, H. Leitte, and I. Viola
(Guest Editors)

Volume 38 (2019), Number 3

Follow The Clicks: Learning and Anticipating Mouse Interactions During Exploratory Data Analysis

Alvitta Ottley, Roman Garnett, and Ran Wan

Computer Science and Engineering, Washington University in St. Louis

Abstract

The goal of visual analytics is to create a symbiosis between human and computer by leveraging their unique strengths. While this model has demonstrated immense success, we are yet to realize the full potential of such a human-computer partnership. In a perfect collaborative mixed-initiative system, the computer must possess skills for learning and anticipating the users' needs. Addressing this gap, we propose a framework for inferring attention from passive observations of the user's click, thereby allowing accurate predictions of future events. We demonstrate this technique with a crime map and found that users' clicks can appear in our prediction set 92% – 97% of the time. Further analysis shows that we can achieve high prediction accuracy typically after three clicks. Altogether, we show that passive observations of interaction data can reveal valuable information that will allow the system to learn and anticipate future events.

CCS Concepts

• ***Human-centered computing*** → ***Visual analytics; Visualization theory, concepts and paradigms;***

1. Introduction

The overarching goal of **visual analytics** is to create a symbiosis between human and machine. Visualization serves as a medium that allows users to collaborate with computers in ways that takes advantage of their distinct strengths [KAF*08]. Both Crouser and Chang [CC12, COC13] and Green et al. [GRF08] describe an affordance-based partnership model that leverages the human's unique skills (e.g., reasoning and social awareness) with the machine's computational powers. Typically, the human drives the analysis process by exploring the data to form hypotheses and develop insights. Success in the analytic process hinges on the user's ability to perform meaningful interactions with the data and on the machine's ability to provide the right information at the right time [EM10, KAF*08].

Although this model has shown remarkable success, for many analysts, information overload is a major concern [CAG*06, ME12]. In many ways, today's tools fall short of their full potential. A useful collaborative tool should possess the ability to learn about what the user is doing, what the user will be doing, what the user ought to be doing, and whether the current trajectory will solve the problem at hand. Current visual analytics tools do not yet possess the ability to learn and anticipate actions, and therefore are unable to tailor their outputs.

The work in this paper aims to model attention during visual data exploration. We propose a context-aware, data-driven prediction system that integrates advancements from artificial intelligence to detect future interactions based on past observations. Specifically, using clicks as a proxy for *attention*, we create a hidden Markov model that represents evolving attention as a series of unobservable states giving rise to actions. We can then automatically infer elements of interest from passive observations of the user's clicks, thereby allowing accurate predictions of future interactions.

For a proof of concept, we conducted a user study and collected click-stream data as participants explored a map visualization of reported crimes (see figure 4). Our results show that the probabilistic model can achieve, depending on the type of task, between 92% and 97% accuracy at predicting that the next click will fall within a small subset of 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 can reveal valuable information about users' attention.

The ability to anticipate future actions opens the door for many opportunities to improve analysts' experience. For instance, the machine can proactively perform tasks such as prefetching, calculation of summaries statistics, suggestion formation, bias or error identification, and target selection assistance for overcrowded interfaces. We discuss how the proposed technique can help create next-generation visual analytics systems that can automatically learn users' focus to support the analysis process better.

We make the following contributions:

- *A generic approach to modeling attention and click dynamics on a visualization:* We present a framework for automatically learning future click events during data exploration and demonstrate, using a crime map, how to model users' attention 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 the participants' clicks appear in our prediction set (5% of the dataset) on average 95% of the time.
- *Implications for designing mixed-initiative visualization tools:* We discuss techniques for supporting the user in real time.

2. Prior Work on Modeling and Predicting User Actions

Predicting future actions has been an important area of research across many fields. For example, in machine-learning, researchers have used interaction data to model and predict users' browsing behaviors on websites and web search systems [EV03,KB00,RWC18, SCDT00,WRC16]. In databases, Battle et al. [BCS16] analyzed interaction data to improve prefetching techniques. They showed that they can successfully predict future actions and that analyzing behavioral data resulted in a 430% improvement in system latency. In HCI, Fu et al. developed statistical and machine-learning models to predict behavior on crowdsourcing annotation and web search tasks [FKW*17]. These are just a few of the many examples of related work across a vast number of research communities. Most relevant to this work is research in the area of *Analytic Provenance* and prior work on computational modeling of selective attention.

2.1. Tracking User Actions

It is a common belief that interaction logs contain crucial information about an analyst's reasoning process with a visualization [PSCO09]. Through interaction with a visual interface, analysts explore data, form and revise hypotheses, and make judgments. The term *provenance* refers to the history of an object or idea, and *analytic provenance* researchers aim to track and analyze the analyst's process [NCE*11]. 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 [RESC16].

Although the scope of Analytical Provenance research is much broader than the goal of the current work (we leverage interaction logs to predict future actions), research in this area have demonstrated a variety of techniques for tracking actions and workflow. For example, Cowley et al. developed *Glassbox* with the goal of logging interactions to infer intent, knowledge, and work-flow [CNS05]. They captured low-level system actions such as keyboard and mouse data, file actions (e.g. open, close, and save), and browser events. Heer et al. recorded actions (e.g. VizQL statements) and system states to create a graphical history tool [HMSA08]. More recent work by Feng et al. logged mouse hovers and search queries to demonstrate metrics for quantifying data exploration [FPH18]. Dabek and Caban tracked behavioral data such as clicks on visual elements and tabs. They introduced a grammar-based approach to modeling user interactions [DC17] and demonstrated that their technique could be used to capture and compare users' analytic process. In this paper, we leverage low-level interaction (e.g. mouse clicks) to model and predict selective attention during data exploration.

2.2. Computational Modeling of Selective Attention

Modeling visual attention in images and videos has been an important area of research in psychophysics, computational modeling and neurophysiology (see [BI13] for a review existing work). Current attention models generally fall into two main categories: *bottom-up approaches* and *top-down approaches*. Bottom-up attention models (stimulus driven) are based on the low-level features of the visual scene, while top-down models (goal driven) are determined by phenomenons such task, goals, experience and knowledge.

Seminal work by Koch and Ullman [KU87] used a purely bottom-up model that decodes a scene based on pre-attentive visual features (e.g. color, depth, and direction of motion) to create a *saliency map* – a two-dimensional topological map that encodes conspicuity across the entire scene. The central thesis of their work is salient features within a stimulus "stands out", thus attracting overt attention. They used a winner-take-all neural network to determine the most salient locations, and defined rules for shifting the processing focus which can be biased by proximity and similarity preferences. Much of the existing work on computational modeling of selective attention have adopted the idea of bottom-up feature extraction and saliency map (e.g. [IKN98], [IK00], and [IK01]) to simulate human viewing behavior. We therefore propose a bottom-up approach for modeling and detecting attention during visual data exploration and demonstrate how this method can be used to predict future attending regions and actions.

2.3. Predicting Actions and Attributes

A handful of work in the Visualization community has directly addressed the prediction of user actions and attributes. For instance, Wall et al. introduced a framework for quantifying different types of biases and proposed a Markov chain technique for predicting bias in real time [WBFE17]. Work by Brown et al. used machine-learning techniques to infer user attributes automatically [BOZ*14]. They showed that off-the-shelf algorithms could successfully predict completion time and personality traits based on low-level mouse clicks and moves [BOZ*14]. Recent work by Fan et al. used a convolution neural network to infer brush selection from a simple click and drag interaction design [FH18]. They demonstrated that their technique can quickly and accurately predict users' intended selections.

3. General Modeling Framework

Deviating from past work, we focus on modeling users attention and interaction in real time with the high-level goal of supporting the user during exploratory data analysis. In this work, we use mouse clicks as a proxy for users' selective attention on a visualization tool. We propose and demonstrate, for the first time to the

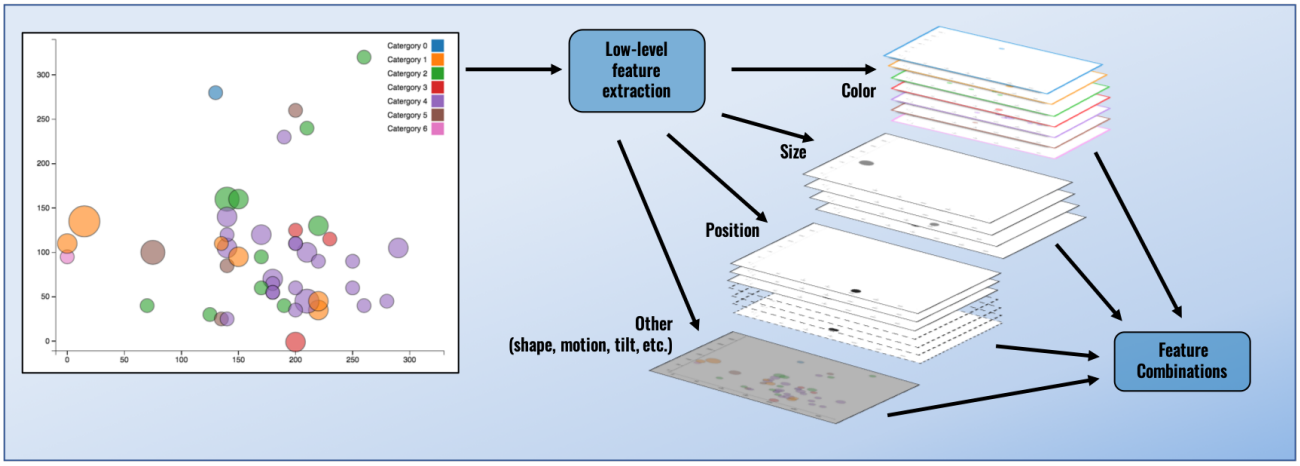

Figure 1: Extracting low-level features.

best of our knowledge, a model for predicting clicks before they occur.

To model attention, we use a bottom-up approach that utilizes low-level visual features (detailed in figure 1). It is important to note that this low-level encoding does not incorporate top-down signals that may be derived from the task at hand. Although a number of researchers have created taxonomies for the types of tasks that are feasible on a visualization system [DE98, GZ09, YaKS07, ZF98], we lack a thorough understanding of how top-down forces may drive the user's attention and analysis process. The work in this paper leverages well established graphical [Ber83] and data mapping principles [CMS09], as well as prior work on using low-level visual features to model attention in images [IKN98, IK00, IK01, KU87].

3.1. A Hidden Markov Model Approach

Predicting attention during exploratory data analysis has two main characteristics: 1) The data are streaming, and 2) There are no available training data. These properties eliminate supervised machine learning approaches, as well as many unsupervised techniques (e.g. RNN), that typically would require waiting until time T to accumulate a significant batch of interactions before beginning predictions. Bayesian methods such as *hidden Markov model,* have demonstrated success with time-varying data, do not require extensive training data, and can stably adapt to real-time changes. Prior work in the visualization community has also used a similar Bayesian method for representing bias from mouse interaction [WBFE17].

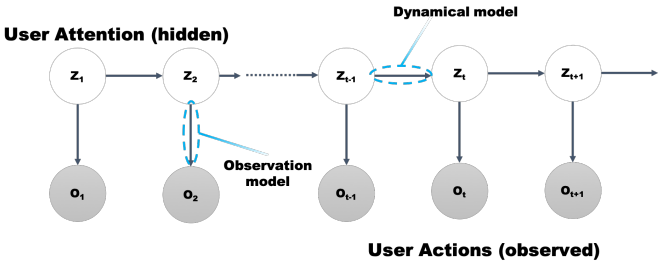

Figure 2: A hidden Markov model approach to modeling attention and actions. We represent evolving attention as a sequence of latent variables in the hidden state space. Observable states are the user's actions. The conditional distribution of each observation depends on the state of the corresponding latent variable.

We construct a hidden Markov model, presuming the user's attention evolves under a Markov process (that is, the attention at a particular time depends on their attention at the previous time step), and interaction events are generated conditionally independently given this sequence of attention shifts. Figure 2 shows an overview of the hidden Markov model used. We represent selective attention as a sequence of latent variables. The conditional distribution of each observation depends on the state of the corresponding latent variable. To specify this model, we need to define the following:

- **Unobservable states:** A space of the possible "interests" driven by the salient visual features.
- **Observable states:** A space of possible interactions.
- **Dynamical model:** A model of the evolution of the user's attention over time.
- **Observation model:** A model of how attention gives rise to observed actions.

3.2. Defining Unobservable and Observable States

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 attention 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 focus areas. The task

we consider here is how to *infer* the hidden attention/focus of the user by observing their sequence of interactions.

In order to create a model of user interaction, we must first understand the mechanisms that drives the user to interact with a particular visual element. We also assume that innate biological models of selective attention drive interactions. At a high level, we build on Koch and Ullman's model of visual attention [KU87] and learn a *saliency map* for a given time step.

3.2.1. Unobservable States

We begin by segmenting the visualization based on the low-level visual features. We define $\mathscr{M}$ 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 [Ber83]. Visual channels describe the graphical properties of visual marks such as position, size, color, luminance, shape, texture, and orientation [Ber83]. Together with Card et al.'s data-mapping principles [CMS99] these design guidelines can be used to describe any existing visual representation [CMS09]. We create $\mathscr{M} = \{f_1, ..., f_N\}$ by decomposing the visualization into its primitive visual marks and channels, as detailed in figure 1.

A crucial component of the probabilistic model is the specification of a hidden state space, which will represent the attention of the user at a given time. In general, we propose that designers can tailor this space for a given scenario. In many scenarios, we may reasonably assume the users' attention at a given time to be related to some weighted subset of visualization marks, for example, visual marks of a particular size, color, shape, or in a specific location. In such a case we may define the latent attention at time t, as $z_t = \{f_{1_t}, ..., f_{N_t}, \pi_t\}$ where π represents the feature weights, and $\{f_{1_t}, \dots f_{N_t}\}$ represent feature values describing the user's focus at time t. We provide more details for the feature weights below.

3.2.2. Observable States

In contrast to the hidden attention 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 event with a visual element (e.g., mouse clicks, mouse moves, eye gaze, etc.). We will represent each observation $o_t = \{f'_1, ..., f'_N\}$ as the set feature values that describes the visual element.

3.3. 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 attention shifts from one time-step to the next. We define z_t to be the latent attention of the user at time t.

3.3.1. Single Task

We model shifts of attention by defining a probability distribution $p(z_{t+1} \mid z_t)$ describing the evolution of attention. We propose that this model should be reasonably easy to define in most visualization settings. In general, it is unlikely that the user's focus 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 (e.g., zero-mean Gaussian noise for real-valued features or a discrete distribution favoring $z_{t+1} = z_t$ for discrete features). This model assumes that focus of attention is likely to remain constant from time t to $t+1$, with some slow decay as the user continues to interact with the system. This is consistent with prior research that suggest that selective attention does not change drastically over time [KU87].

Table 1: Mathematical symbols.

Symbols	Description
t	the time an event occurs.
$\mathscr{M} = \{f_1, ..., f_N\}$	Mark Space: The set of N visual features extracted from the visualization (e.g., position, size, and color).
$o_t = \{f'_1, ..., f'_N\}$	observation interaction at time t (e.g., mouse click). We consider set of values for the N features.
$\pi = [\pi(f_1), ..., \pi(f_N)]$	bias vector for all features $f \in \mathscr{M}$.
$z_t = \{f_{1_t}, ..., f_{N_t}, \pi_t\}$	latent attention at time t.

3.3.2. Multiple Tasks

If a visualization setting may comprise a sequence of separate tasks, we may also construct dynamical models that loosely encode that user's attention 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 attention at the next time step as being drawn from some broad distribution over the space of possible focus points. 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. Such an approach has been used to model user intent in online games from observed low-level behavior [GGE*14].

3.3.3. Evolution of Attention

Koch and Ullman hypothesized that it is useful to consider bias when modeling attention shifts [KU87]. There are two primary mechanism that drives the evolution of attention: spatial proximity and similarity [KU84, KU87]. Motivated by the prior work, we adopt a *bias vector* $\pi = [\pi(f_1), ..., \pi(f_N)]$ to capture the relative importance of the various components of the mark space where $\pi(f) \in [0, 1]$.

For the dynamical model of the hidden state $p(z_t \mid z_{t-1})$, we assume that the attention at time $t+1$ is typically similar to the attention at the previous time step t; that is, that attention does not change rapidly over time. We further assume that each component

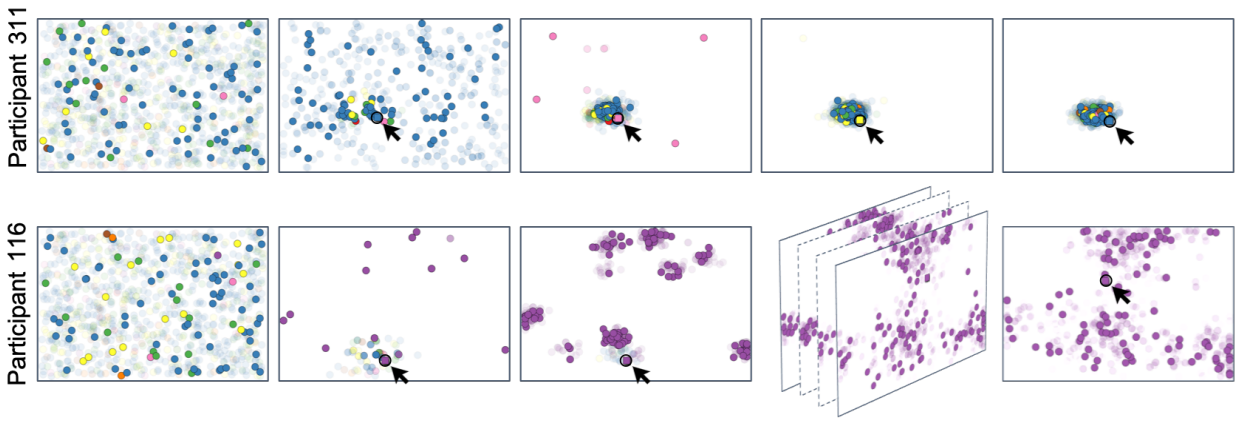

Figure 3: The first row shows the initial state of the particles as well as the inferences after the first four clicks for participant #311. The arrows indicate the observed clicks. Within a few clicks, the predictions for the user's evolving attention converge to circles within the general area of the observed clicks. The second row shows the particles from participant #116 who clicked points of a single color across different regions of the map. Here, we see a broad spread for the location parameters but a convergence of the type parameters over time.

of the attention vector evolves independently:

$$p(z_{t+1} \mid z_t) = p(f_{1_{t+1}} \mid f_{1_t}) \ldots p(f_{N_{t+1}} \mid f_{N_t}) p(\pi_{t+1} \mid \pi_t).$$

We also suggest that the relative importance of the various components of the mark space should remain relatively stable over time, and can adopt a diffusion for the bias parameter π as well:

$$p(\pi_{t+1} \mid \pi_t) = \mathcal{N}(\pi_{t+1}; \pi_t, \sigma_\pi^2).$$

Note that we must account for boundary effects and normalization effects when defining the dynamical model; in practice, we may simply project out-of-range values onto their feasible domains.

3.3.4. Continuous Features

For continuous features f such as position, we may model the evolution of features using additive zero-mean Gaussian noise:

$$p(f_{t+1} \mid f_t, \sigma_f^2) = \mathcal{N}(f_{t+1}; f_t, \sigma_f^2),$$

where the parameter σ_f^2 is the variance of the drift. For strictly positive values such as size or intensity, we could use a similar diffusion on the logarithm of the value instead, or we could simply project onto the feasible domain.

3.3.5. Categorical Features

One possibility for modeling the evolution of an arbitrary discrete parameter f such as color or shape is a simple "biased coin flip" model favoring no change:

$$p(f_{t+1} \mid f_t, \rho) = \rho\, \delta(f_t) + (1-\rho)\, \mathscr{U}_{\setminus f_t},$$

where ρ is a parameter modeling the fickleness of the user, $\delta(f_t)$ is the Kronecker delta distribution with support f_t, and $\mathscr{U}_{\setminus f_t}$ is the uniform distribution over the values not equal to f_t. This distribution effectively says the user's attention does not change with probability ρ; otherwise, it changes to a different value with equal probability.

3.3.6. Ordinal Features

We suggest treating ordinal feature as either categorical or continuous and using one of the above.

3.4. Observation Model

We must also specify an observation model $p(o_t \mid z_t)$, which defines how latent user attention 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 use case scenario below. In a visualization setting, defining a reasonable choice for such a model is relatively straightforward. If a user's attention 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 interact with elements related to their hidden attention space." We will show an explicit construction of such a model in Section 4.

3.5. 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 events. To approach this goal, we will use our hidden Markov model to infer the attention of the user at time t, z_t, given the interactions up to time t, $O_t = \{o_i\}_{i=1}^t$. Unfortunately, this inference is usually not possible in closed form, but we can use a particle filter. Particle filtering is a well-established technique for inferring the hidden states of dynamical systems such as ours [DGA00, GSS93].

We represent our belief about the latent state z_t given the previous events O_t with a set of m *particles* $\{z_t^{(i)}\}_{i=1}^m$, each particle a point in the attention space. These particles represent samples from the posterior distribution $p(z_t \mid O_t)$. 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 o_{t+1} and weight the particles according to the agreement with the observation by evaluating the observation model:

$$w^{(i)} = p(o_{t+1} \mid z_{t+1} = z_{t+1}^{(i)}).$$

- 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 $p(z_{t+1} \mid O_{t+1})$, and we may proceed inductively. For each timestamp, we can get $p(z_t \mid O_t)$, which is the particle given all previous interaction events. 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 contributes to each candidate using the observational model. The attention space thus is a set of weighted particles, and the particle weights are probability masses. We select the visual marks with the highest probability mass by summing the weights of every particle associated with that mark. The top-K visual marks with the highest probability mass becomes our prediction set.

4. Illustrative Example

We now demonstrate how to apply this model to a visualization interface (see figure 4). We chose a map for our study because geographic maps are one of the most popular wed-based visualization types [BDM*18] and has broad application and use. Below 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.

4.1. Defining Unobservable and Observable States

We define o_t to be the click event at time t, which we will represent as a three-dimensional vector $o_t = (x_t', y_t', k_t')$, 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 the eight possible values $\{1, \ldots, 8\}$. Note that we use prime symbols to indicate quantities associated with a click event.

Next, we will define a hidden state space modeling the attention 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 bias parameter indicating the relative importance of location and mark color. For this example, we represent the bias 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 attention space is thus a four-dimensional vector $z = (x, y, k, \pi)$.

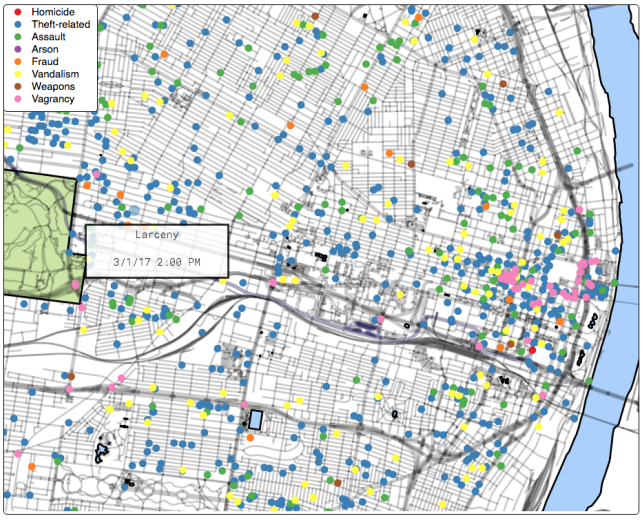

Figure 4: The interface used in our experiment. Participants used their mouse to pan and zoom the map. A tooltip displayed information about the crimes on click.

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 attention z_t corresponding to a vector in the attention space defined above. We seek to infer the attention of the user through observing the sequence of click events $\{o_t\}$. 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 $p(o_t \mid z_t)$ describing how a hidden attention vector generates click events. We define each of these below.

4.2. Dynamical Model

Here, we adopt a simple stationary diffusion model. As detailed in Section 3.3.3, we assume that the four components of the attention 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 bias 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 map and must lie in its domain, and the bias 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 map, 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 following our suggestion in Section 3.3.3. We define a transition probability ρ and assume that with probability ρ that the latent mark color of interest does not change. Otherwise, we chose a new mark color of interest from all possible values with equal probability:

$$p(k_{t+1} \mid k_t, \rho) = \rho\,\delta(k_t) + (1-\rho)\mathscr{U}_{\setminus k_t},$$

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

4.3. Observation Model

We must also specify an observational model $p(o_t \mid z_t)$ modeling the probability of a click event $e_t = (x'_t, y'_t, k'_t)$ given the attention $z_t = (x_t, y_t, k_t, \pi_t)$ at time t. A brief summary of this observational model is that we flip a coin with heads probability equal to the location–color bias 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:

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

where $\mathscr{U}(k'_t; k_t)$ denotes a uniform distribution over the available marks of color k_t. This above model therefore assumes that if the user is interested in position (with probability π_t), she will click on a position on the map 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.

4.3.1. Predicting Movements

To predict movements, we can apply a particle filter as described in Section 3.5. Figure 3 shows the initial state of the particle as well as subsequent updates to the model's inference for two participants from the study. We can observe that participant 311 begins by clicking on dots at the center of the projection, and within a few clicks, the attention inferences converge to circles in a tightly defined area. For participant 116, we see a convergence of the type parameters, but a divergence of the location parameters. We make available a demo of the particle filtering approach at https://washuvis.github.io/particles/.

5. Proof of Concept Evaluation

For a proof of concept, we designed a user study to track and analyze mouse interactions. 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 [St.]. 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.

5.1. 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.

5.2. 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. 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:

- **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 questions were simplified versions of real-world tasks that represents a potential interest. For instance, a person who in interested in buying a house may visit a crime map to learn about the frequent types of crimes in the neighborhood (Geo-Based). A fire marshal may be interested in trends across reported cases of Arsons (Type-Based), or an investor may want to learn about theft crimes that tend to occur near a potential business site (Mixed).

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 crime instances (a total of 43 dots) in the specified region. 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

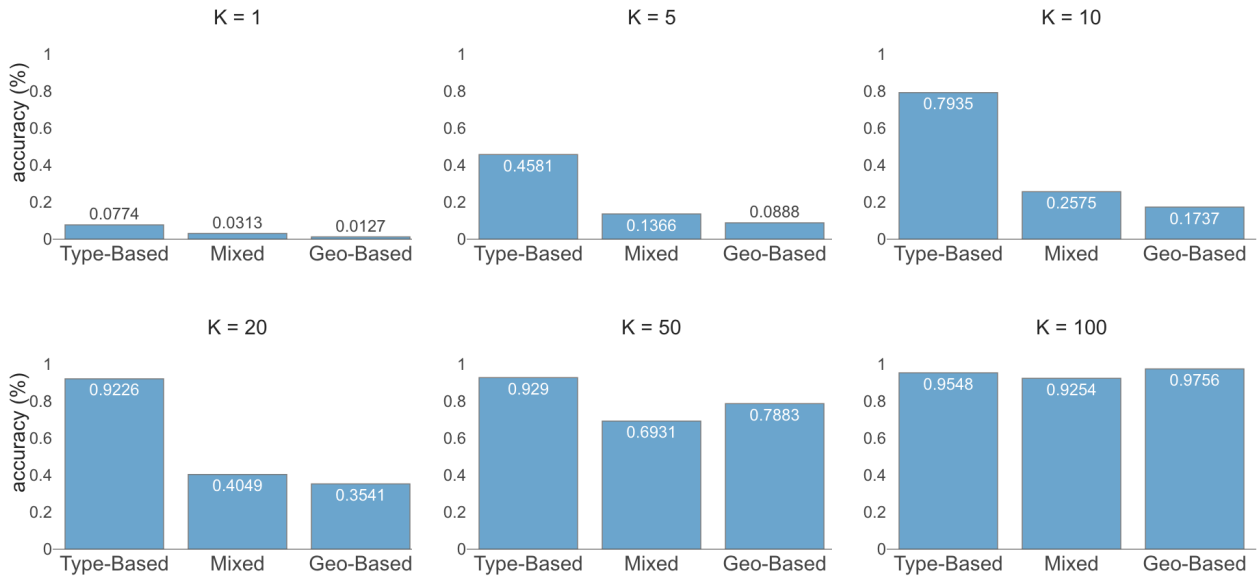

Figure 5: The average prediction accuracy across the three type of tasks (Geo-Based, Type-Based, and Mixed) for varying values of the prediction set size. We report our results as top-K accuracy. For $K = 100$ our algorithm successfully predicted the users' next click, on average, 95% of the times. This means that we can successful predict that the next click will be within a small subset of the dataset.

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 instances 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 region."* Participants clicked 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. We designed the tasks to encouraged participants to click a large number of valid points. This was done to ensure a reasonably rich and large interaction dataset.

5.3. Procedure

After selecting the task on Mechanical Turk, participants consented per Washington University's IRB protocol. They read the instructions for the study, then 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 a short demographic questionnaire.

5.4. 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 (two per task type), resulting in 180 trials.

Crowdsourced-based studies are notoriously noisy. It is impossible to know whether a participant was distracted during the task, or was clicking through the tasks simply to get paid. To ensure the best quality data, it is therefore important to filter out inattentive participants and invalid responses [KCS08, KZ10]. Consistent with prior crowdsourced studies in the visualization community that analyzed interaction data, we used the tasks' ground truth to both incentivize accurate responses [OYC15] and to filter potentially distracted participants [BOZ*14, OYC15]. After cleaning, 78 trials remained (28, 23, and 27 trials for Geo-Based, Type-Based and Mixed tasks respectively).

5.4.1. Predicting Movement

To predict movements, we applied particle filter as described in Section 3.5. We report the top-K visual mark retrieval for our method where $K = \{1, 5, 10, 20, 50, 100\}$. Although the choice for K (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 focus mainly on $K = 100$ 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 $t + 1$.

5.4.2. Parameters

We used 1000 particles. The parameters were set as $\sigma_x = \sigma_y = 0.1, \sigma_\pi = 0.45$. The probability of maintaining the same type of crime as the users' attention ρ was 0.96. These parameter were handpicked based on the prior works' hypotheses for the dynamics of selective attention [KU84]. However, tuning the hyperparameters using Bayesian optimization (we tested the technique proposed by Snoek et al. [SLA12]) results in more accurate predictions for smaller values of K.

5.5. Results

5.5.1. Prediction Accuracy

After gathering the data, we analyzed our model's ability to observe mouse clicks and predict interactions before they occur. To allow time for the algorithm to learn users' attention, we begin our predictions at $t = 3$. If the click at $t+1$ falls within our prediction set, we consider this a success. For each type of task (Geo-Based, Type-Based, and Mixed) we measured the overall predictive accuracy across all available clicks for all users:

$$\frac{\sum successfulPredictions}{\sum predictions}$$

Figure 5 shows the model's accuracy across different values of $K = \{1, 5, 10, 20, 50, 100\}$ for each of the three tasks. Unsurprisingly, we see a direct correlation between accuracy and the size of the prediction sets. For $K = 100$ (5% of the dataset), our technique attained an average of 95% at predicting the participants' next clicks across all three task types ($\mu = .9548$, $\sigma = .1245$ for Type-Based, $\mu = .9254$, $\sigma = .0485$ for Mixed, and $\mu = .9756$, $\sigma = .0719$ for Geo-Based 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.

Overall, we found that the model performs well for type-based tasks, even with small prediction sets. For instance, when limited to only ten guesses for the next click, we see an average accuracy of 79% across all participants for the type-based task. The average accuracy increases to 92% when the set size is 20 (0.01% of the size of the full dataset). This may be due to the small number of instances for type based tasks (there were only 5 cases of Homicides and 14 cases of Arsons in the dataset).

5.5.2. Accuracy Over Time

Our second analysis sought to evaluate our algorithm's performance as a function of the number of clicks observed. For each type of task, we measure the prediction accuracy (set size = 100) for the first 20 clicks observed. Consistent with our previous analysis, we begin our predictions at $t = 3$. Figure 6 summarizes our findings. Our analysis reveals that the technique promptly achieves high prediction accuracies and performance remains fairly stable with more observations.

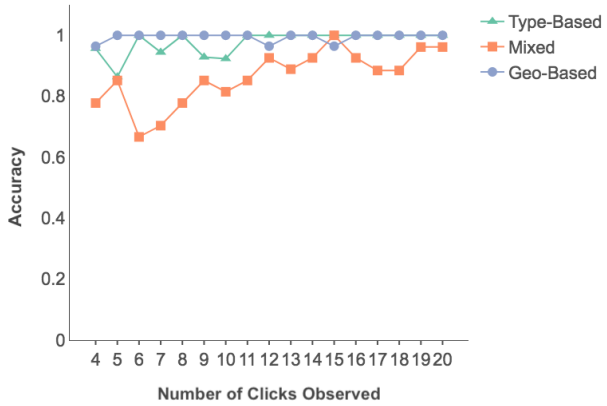

Figure 6: The average accuracy over time for the three types of tasks in the study. After learning from 3 click interactions, the algorithm immediately achieves high prediction accuracy. We found that prediction accuracies remain fairly constant over time.

6. Discussion

The hidden Markov model is a general framework that is widely used for modeling sequence data in areas such as natural language processing [MS99], speech recognition [Jel97, RJ93], and biological sequencing [DEKM98, SVHK*98]. However, we demonstrate its utility for modeling attention from interaction with a visualization system. There are 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. A designer may tune these parameters or customize them based on the visualization or task. We see this as a strength of the approach which can seed many opportunities for future work.

Although, the evaluation uses a single interface, we posit that the approach in this paper is generalizable under transparent assumptions. We leverage data mapping principles and the notion that we can represent a visualization as a set of primitive visual marks and channels. 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 evolving attention, we must first carefully define the mark space, $\mathscr{M}$. One way to improve this process is to automatically extract the visual marks and channels from the visualization's code.

Applying a simpler solution to our illustrative example may be possible. For instance, a k-nearest neighbors approach (k-NN) based on a moving window of past interactions might result in similar accuracies. While k-NN and other pattern recognition methods may work well for simple scenarios, we opted for a technique that predicts based on a sequence of features. The model in this paper provides a generalizable approach that allows us to modify the definition of the latent state space (e.g., attention or "intent") or even test hypotheses by simulating how different latent variables give rise to actions. We can, for example, use this model to reason about the distinct forces that may drive interactions across task types. Additionally, the proposed particle filtering approach has a computational complexity of $O(n)$ for inferences and $O(nm)$ for each pre-

diction, where m is the number of marks on the visualization and n is the number of particles. For 1000 particles, the running time is 0.058 seconds per iteration, demonstrating that the technique is appropriate for real-time inference and predictions [CRM91, Mil68].

Modeling attention can be a rich signal for inferring goals, intention and interest [Hor99a, HKPH03], and information about users' current and future attention can be useful for allocating computational resources [HKPH03] or for supporting data exploration [FH18]. For example, the system can perform precomputation or pre-fetching based on its predictions. For large datasets that may have overlapping points, 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 paper to inform techniques for target assistance [BMSG11]. The *bubble cursor* technique, for example, does not change the visual appearance of the interface but increases the click radius for the given target, thereby making them more accessible [GB05]. Another possibility is *target gravity,* which attracts the mouse to the target [BMSG11]. Future work can explore how to utilized such techniques to support the user during analytic tasks.

The general idea of mixed initiative systems [AGH99, Hor99a, Hor99b, Hor07] or tailoring an interface based on users' skills or needs has existed for many years in HCI [GW04]. Researchers have explored the tradeoff between providing support and minimizing disruptions [APCJ13, SLC*11]. In visualization, Ceneda et al. provides a comprehensive overview of techniques for the providing guidance in visual analytics [CGM*17]. The work in the paper aligns well with this broader research agenda.

7. Limitations and Future Work

Although the work in this paper focuses on learning and anticipating future actions, we believe that the proposed framework opens possibilities for future work. One possible path for future work is to investigate the model's performance on 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. While these tasks were designed based on realistic scenarios, they assume that the user has a specific and unchanging goal. As a result, the search patterns we observed may not generalize to open-ended scenarios, or when the user's interest change while interacting with the data. It is also possible that there are some scenarios where the user's attention cannot be represented as subspace of the visualization marks (e.g., attending to negative space). Future work can evaluate the approach with open-ended tasks.

One of the main challenges of modeling user behavior in the visualization community is there are no large-scale training datasets. Researchers typically have to design and conduct user studies to collect data. This limitation not only excludes many machine learning approaches but it increases the difficulty of demonstrating generalizability of a technique. 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. It is also common for designers to aggregate the data based on the zoom level of the interface. 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.

8. Conclusion

In this paper, we have proposed a generalizable approach to modeling users' evolving attention and actions with a visualization system. We used a hidden Markov model and represented attention using the primitive visual marks and channels of the visualization design. We demonstrated with a simple map how to apply this approach to a given visualization design.

We conducted a user study and captured mouse click 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 a top-K accuracy of 95% at guessing actions before they occur for $K = 100$. 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 paper 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.

9. Acknowledgments

We thank Surina Puri for her help with the experiment design and data collection. This project was supported by the National Science Foundation under Grant No. 1755734.

Appendix: Extended Results

Mechanical Turk data is notoriously noisy, and it is common practice to filter inattentive participants. We focused our previous analyses on subjects who provided the correct answers. However, noisiness is an inherent feature in real-world data, and it is relevant to demonstrate how the technique fare under realistic circumstances.

Table 2: The top-K accuracy for the minimally filtered dataset.

1	5	10	20	50	100
0.0542	0.3833	0.6292	0.7417	0.7625	0.8042
0.0304	0.1300	0.2395	0.3992	0.6893	0.9083
0.0149	0.0948	0.1826	0.3659	0.8018	0.9752

In this appendix, we present the algorithm's performance with the minimally filtered data. To guarantee that the algorithm has enough time to learn, we only excluded participants with five clicks or less. From table 2, we can observe prediction accuracy trends that are similar to our main results. It is important to note that this dataset also included participants who apparently ignored the experiment instructions and were likely clicking through the tasks to get paid. Still, for K = 100, the next click appeared in the prediction set, on average, 90% of the times. The number of sessions was 32, 44, and 43 for Type-based, Mixed, and Geo-Based tasks respectively.

References

[AGH99] ALLEN J., GUINN C. I., HORVTZ E.: Mixed-initiative interaction. *IEEE Intelligent Systems and their Applications 14*, 5 (1999), 14–23. 10

[APCJ13] AFERGAN D., PECK E. M., CHANG R., JACOB R. J.: Using passive input to adapt visualization systems to the individual. 10

[BCS16] BATTLE L., CHANG R., STONEBRAKER M.: Dynamic prefetching of data tiles for interactive visualization. In *Proceedings of the 2016 International Conference on Management of Data* (2016), ACM, pp. 1363–1375. 2

[BDM*18] BATTLE L., DUAN P., MIRANDA Z., MUKUSHEVA D., CHANG R., STONEBRAKER M.: Beagle: Automated extraction and interpretation of visualizations from the web. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), ACM, p. 594. 6

[Ber83] BERTIN J.: Semiology of graphics: diagrams, networks, maps. 3, 4

[BI13] BORJI A., ITTI L.: State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence 35*, 1 (2013), 185–207. 2

[BMSG11] BATEMAN S., MANDRYK R. L., STACH T., GUTWIN C.: Target assistance for subtly balancing competitive play. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011), ACM, pp. 2355–2364. 10

[BOZ*14] BROWN E. T., OTTLEY A., ZHAO H., LIN Q., SOUVENIR R., ENDERT A., CHANG R.: Finding waldo: Learning about users from their interactions. *IEEE Transactions on visualization and computer graphics 20*, 12 (2014), 1663–1672. 2, 8

[CAG*06] CONTI G., ABDULLAH K., GRIZZARD J., STASKO J., COPELAND J. A., AHAMAD M., OWEN H. L., LEE C.: Countering security information overload through alert and packet visualization. *IEEE Computer Graphics and Applications 26*, 2 (2006), 60–70. 1

[CC12] CROUSER R. J., CHANG R.: An affordance-based framework for human computation and human-computer collaboration. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2859–2868. 1

[CGM*17] CENEDA D., GSCHWANDTNER T., MAY T., MIKSCH S., SCHULZ H.-J., STREIT M., TOMINSKI C.: Characterizing guidance in visual analytics. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2017), 111–120. 10

[CMS99] CARD S. K., MACKINLAY J. D., SHNEIDERMAN B.: *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999. 4

[CMS09] CARD S., MACKINLAY J., SHNEIDERMAN B.: Information visualization. *Human-computer interaction: Design issues, solutions, and applications 181* (2009). 3, 4

[CNS05] COWLEY P., NOWELL L., SCHOLTZ J.: 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* (2005), IEEE, pp. 296c–296c. 2

[COC13] CROUSER R. J., OTTLEY A., CHANG R.: Balancing human and machine contributions in human computation systems. In *Handbook of Human Computation*. Springer, 2013, pp. 615–623. 1

[CRM91] CARD S. K., ROBERTSON G. G., MACKINLAY J. D.: The information visualizer, an information workspace. In *Proceedings of the SIGCHI Conference on Human factors in computing systems* (1991), ACM, pp. 181–186. 10

[DC17] DABEK F., CABAN J. J.: 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 (2017), 41–50. 2

[DE98] DIX A., ELLIS G.: Starting simple: adding value to static visualisation through simple interaction. In *Proceedings of the working conference on Advanced visual interfaces* (1998), ACM, pp. 124–134. 3

[DEKM98] DURBIN R., EDDY S. R., KROGH A., MITCHISON G.: *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998. 9

[DGA00] DOUCET A., GODSILL S., ANDRIEU C.: On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing 10*, 3 (2000), 197–208. 5

[EM10] ELLIS G., MANSMANN F.: Mastering the information age solving problems with visual analytics. In *Eurographics* (2010), vol. 2, p. 5. 1

[EV03] EIRINAKI M., VAZIRGIANNIS M.: Web mining for web personalization. *Transactions on Internet Technology (TOIT) 3*, 1 (2003), 1–27. 2

[FH18] FAN C., HAUSER H.: Fast and accurate cnn-based brushing in scatterplots. In *Computer Graphics Forum* (2018), vol. 37, Wiley Online Library, pp. 111–120. 2, 10

[FKW*17] FU E. Y., KWOK T. C., WU E. Y., LEONG H. V., NGAI G., CHAN S. C.: Your mouse reveals your next activity: Towards predicting user intention from mouse interaction. In *Computer Software and Applications Conference (COMPSAC), 2017 IEEE 41st Annual* (2017), vol. 1, IEEE, pp. 869–874. 2

[FPH18] FENG M., PECK E., HARRISON L.: Patterns and pace: Quantifying diverse exploration behavior with visualizations on the web. *IEEE transactions on visualization and computer graphics* (2018). 2

[GB05] GROSSMAN T., BALAKRISHNAN R.: 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* (2005), ACM, pp. 281–290. 10

[GGE*14] GARNETT R., GÄRTNER T., ELLERSIEK T., GUÐMONDSSON E., ÓSKARSSON P.: Predicting Unexpected Influxes of Players in EVE Online. In *Proceedings of the 2014 IEEE Conference on Computational Intelligence and Games* (2014). 4

[GRF08] GREEN T. M., RIBARSKY W., FISHER B.: Visual analytics for complex concepts using a human cognition model. In *Visual Analytics Science and Technology, 2008. VAST'08. IEEE Symposium on* (2008), IEEE, pp. 91–98. 1

[GSS93] GORDON N. J., SALMOND D. J., SMITH A. F.: Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)* (1993), vol. 140, IET, pp. 107–113. 5

[GW04] GAJOS K., WELD D. S.: Supple: automatically generating user interfaces. In *Proceedings of the Ninth International Conference on Intelligent User Interfaces* (2004), ACM, pp. 93–100. 10

[GZ09] GOTZ D., ZHOU M. X.: Characterizing users' visual analytic activity for insight provenance. *Information Visualization 8*, 1 (2009), 42–55. 3

[HKPH03] HORVITZ E., KADIE C., PAEK T., HOVEL D.: Models of attention in computing and communication: from principles to applications. *Communications of the ACM 46*, 3 (2003), 52–59. 10

[HMSA08] HEER J., MACKINLAY J., STOLTE C., AGRAWALA M.: Graphical histories for visualization: Supporting analysis, communication, and evaluation. *IEEE transactions on visualization and computer graphics 14*, 6 (2008). 2

[Hor99a] HORVITZ E.: Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (1999), ACM, pp. 159–166. 10

[Hor99b] HORVITZ E.: Uncertainty, action, and interaction: In pursuit of mixed-initiative computing. *IEEE Intelligent Systems 14*, 5 (1999), 17–20. 10

[Hor07] HORVITZ E. J.: Reflections on challenges and promises of mixed-initiative interaction. *AI Magazine 28*, 2 (2007), 3. 10

[IK00] ITTI L., KOCH C.: A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision research 40*, 10-12 (2000), 1489–1506. 2, 3

[IK01] ITTI L., KOCH C.: Computational modelling of visual attention. *Nature reviews neuroscience 2*, 3 (2001), 194. 2, 3

[IKN98] ITTI L., KOCH C., NIEBUR E.: A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence 20*, 11 (1998), 1254–1259. 2, 3

[Jel97] JELINEK F.: *Statistical methods for speech recognition*. MIT press, 1997. 9

[KAF*08] KEIM D., ANDRIENKO G., FEKETE J.-D., GÖRG C., KOHLHAMMER J., MELANÇON G.: Visual analytics: Definition, process, and challenges. In *Information visualization*. Springer, 2008, pp. 154–175. 1

[KB00] KOSALA R., BLOCKEEL H.: Web mining research: A survey. *ACM SIGKDD Explorations Newsletter 2*, 1 (2000), 1–15. 2

[KCS08] KITTUR A., CHI E. H., SUH B.: Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2008), ACM, pp. 453–456. 8

[KU84] KOCH C., ULLMAN S.: *Selecting One Among the Many: A Simple Network Implementing Shifts in Selective Visual Attention.* Tech. rep., MASSACHUSETTS INST OF TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE LAB, 1984. 4, 9

[KU87] KOCH C., ULLMAN S.: Shifts in selective visual attention: towards the underlying neural circuitry. In *Matters of intelligence*. Springer, 1987, pp. 115–141. 2, 3, 4

[KZ10] KOSARA R., ZIEMKIEWICZ C.: Do mechanical turks dream of square pie charts? In *Proceedings of the 3rd BELIV'10 Workshop: Beyond time and errors: Novel evaluation methods for information visualization* (2010), ACM, pp. 63–70. 8

[ME12] MENGIS J., EPPLER M. J.: Visualizing instead of overloading: Exploring the promise and problems of visual communication to reduce information overload. *Information Overload: An International Challenge for Professional Engineers and Technical Communicators* (2012), 203–229. 1

[Mil68] MILLER R. B.: Response time in man-computer conversational transactions. In *AFIPS Fall Joint Computing Conference (1)* (1968), pp. 267–277. 10

[MS99] MANNING C. D., SCHÜTZE H.: *Foundations of statistical natural language processing*. MIT press, 1999. 9

[NCE*11] NORTH C., CHANG R., ENDERT A., DOU W., MAY R., PIKE B., FINK G.: Analytic provenance: process+ interaction+ insight. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (2011), ACM, pp. 33–36. 2

[OYC15] OTTLEY A., YANG H., CHANG R.: 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* (2015), ACM, pp. 3251–3254. 8

[PSCO09] PIKE W. A., STASKO J., CHANG R., O'CONNELL T. A.: The science of interaction. *Information Visualization 8*, 4 (2009), 263–274. 2

[RESC16] RAGAN E. D., ENDERT A., SANYAL J., CHEN J.: Characterizing provenance in visualization and data analysis: an organizational framework of provenance types and purposes. *IEEE transactions on visualization and computer graphics 22*, 1 (2016), 31–40. 2

[RJ93] RABINER L. R., JUANG B.-H.: *Fundamentals of speech recognition*, vol. 14. PTR Prentice Hall Englewood Cliffs, 1993. 9

[RWC18] RAMACHANDRAN A., WANG L., CHAINTREAU A.: Dynamics and prediction of clicks on news from twitter. In *Proceedings of the 29th on Hypertext and Social Media* (2018), ACM, pp. 210–214. 2

[SCDT00] SRIVASTAVA J., COOLEY R., DESHPANDE M., TAN P.-N.: Web usage mining: Discovery and applications of usage patterns from web data. *ACM SIGKDD Explorations Newsletter 1*, 2 (2000), 12–23. 2

[SLA12] SNOEK J., LAROCHELLE H., ADAMS R. P.: Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems* (2012), pp. 2951–2959. 9

[SLC*11] SOLOVEY E. T., LALOOSES F., CHAUNCEY K., WEAVER D., PARASI M., SCHEUTZ M., SASSAROLI A., FANTINI S., SCHERMERHORN P., GIROUARD A., ET AL.: Sensing cognitive multitasking for a brain-based adaptive user interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011), ACM, pp. 383–392. 10

[St.] ST. LOUIS METROPOLITAN POLICE DEPARTMENT: http://www.slmpd.org/crime_mapping.shtml. Accessed November 13, 2017. 7

[SVHK*98] SONNHAMMER E. L., VON HEIJNE G., KROGH A., ET AL.: A hidden markov model for predicting transmembrane helices in protein sequences. In *Ismb* (1998), vol. 6, pp. 175–182. 9

[WBFE17] WALL E., BLAHA L. M., FRANKLIN L., ENDERT A.: Warning, bias may occur: A proposed approach to detecting cognitive bias in interactive visual analytics. In *IEEE Conference on Visual Analytics Science and Technology (VAST)* (2017). 2, 3

[WRC16] WANG L. X., RAMACHANDRAN A., CHAINTREAU A.: Measuring click and share dynamics on social media: A reproducible and validated approach. In *News@ ICWSM* (2016). 2

[YaKS07] YI J. S., AH KANG Y., STASKO J.: Toward a deeper understanding of the role of interaction in information visualization. *IEEE transactions on visualization and computer graphics 13*, 6 (2007), 1224–1231. 3

[ZF98] ZHOU M. X., FEINER S. K.: Visual task characterization for automated visual discourse synthesis. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1998), ACM Press/Addison-Wesley Publishing Co., pp. 392–399. 3