



HAL
open science

Modeling and Understanding Human Routine Behavior

Nikola Banovic, Tofi Buzali, Fanny Chevalier, Jennifer Mankoff, Anind K Dey

► **To cite this version:**

Nikola Banovic, Tofi Buzali, Fanny Chevalier, Jennifer Mankoff, Anind K Dey. Modeling and Understanding Human Routine Behavior. ACM CHI Conference on Human Factors in Computing Systems 2016, ACM, May 2016, Santa Clara, California, United States. pp.248 - 260, 10.1145/2858036.2858557 . hal-01416119

HAL Id: hal-01416119

<https://inria.hal.science/hal-01416119>

Submitted on 14 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

types of routines from diverse types of behavior: people's daily schedules and commutes [11] and activities that describe how people operate a vehicle [18]. We show that the extracted routine patterns are at least as predictive of behaviors in the two behavior logs as the baseline we establish with existing algorithms. Next, we recruited researchers that work with human activity and routine data to verify that patterns extracted using our approach are meaningful and match the ground truth reported in previous work [11, 18]. For the purposes of this task, we developed a tool that enabled the researchers to visually explore and compare the routines extracted using our approach.

Our first set of results show that our approach enables extraction of a reasonable set of human readable patterns of routine behavior from behavior logs. Our second set of results demonstrate that the models of causal relationships provided by our approach can help researchers explore, understand, and form new insights about human routines from behavior logs without having to manually search for those patterns in raw data. Another benefit of our approach compared to existing systems is that it promises efficient automated prediction and reasoning about routines even under uncertainty that is inherent in human behavior. We discuss how such models and knowledge about routines can inform the design of novel systems that help people improve their behavior.

UNDERSTANDING HUMAN ROUTINE BEHAVIOR

Many different stakeholders may care about understanding routine behavior. For example, a clear picture of the many aspects of routine behavior can help researchers to generate theories and models of human routine behavior. Designers may build on such theories to design technologies that help people to improve their routines [11, 12]. Models of human routines may be used for prediction and automation [43]. Individuals may wish to reflect about their own routines for better understanding and supporting behavior change [23].

Although routine behaviors result from low-level cognitive plans, which can be modeled using existing cognitive architectures (*e.g.*, [4]), such models are difficult to apply in understanding high-level routine behaviors. Thus our focus in this work is on how routines are expressed in actions people perform in different contexts.

Through an analysis of stakeholder use cases, and a review of the literature, we derive three research questions that these stakeholders, and in particular researchers, who analyze routines may wish to answer in order to meet their goals. We review existing approaches to understanding routines based on how well they answer those questions.

Current Approaches to Understanding Routine Behavior

Visualizing data from behavior logs is a common way for researchers to identify routines. Logged behavior data is often visualized on a timeline as a sequence of events. The simplicity of this approach makes it applicable to a variety of domains, such as schedule planning to show uncertainty

of duration of different events [2], visualizing family schedules [11], and representing events related to patient treatments [31]. More advanced timelines enable the user to specify properties of the timeline for easier viewing. For example, Spiral Graph [39] aligns sequential events on a spiral timeline using a user-defined period. However, due to the complexity and size of behavior logs, simply visualizing raw event data does not guarantee that the user will be able to find patterns of behaviors that form routines.

Other visualization approaches enable users to find patterns in time-series data by manually querying and highlighting different parts of behavior event sequences [9, 41, 42] or manually aggregating common sequences of events based on features in the data [20, 28] until meaningful patterns emerge. The user is then able to judge the quality and saliency of the patterns by visual inspection. However, during the early exploratory stages, users might not always know what features are important and contribute to routine behavior patterns, making manual exploration challenging.

One major limitation of existing visualizations lies in their lack of support for both context and actions. Rather, they focus on isolated events, or temporal evolution of a particular state (*e.g.*, sleep vs. awake) or variable (*e.g.*, the amount of steps walked per day). Visualizing both context and actions is challenging partially because even advanced interactive visualizations have difficulty in visualizing routine patterns that depend on multiple heterogeneous variables, especially as the number of variables grows.

Automated routine extraction and summarization is another option for exploring routines in behavior logs. However, patterns extracted using the existing methods often do not include important aspects of routine behavior. For example, T-patterns [7, 26] can automatically find recurrences of events in behavior logs, but do not explicitly capture the contexts and actions. Other methods, such as Topic Models [15], include features that describe both context and actions, but without modeling the structure of possible variations from those routines. Methods based on Hierarchical Task Networks [24] and Eigen decomposition [14, 35] capture the structural components of the contexts and actions, but do not explicitly model the causal relationship between the two that defines the routine [17]. While these algorithms are helpful for extracting routine, they are not sufficient for helping to understand routine behaviors.

This leads to our first research question that researchers want to answer: *RQPat: How can we expose the full complexity of routine behavior patterns?* To extract and understand routines, it is critical to discern which features of the context influence routine actions and which features of the actions people have a demonstrated preference for.

Where Routines End and Variations Begin

How people respond to routine *variations* is also an important part of routine behavior. Human routine behavior is not static, which makes variations from the routines

inevitable [16]. Variations often result from new contexts or unforeseen circumstances for which people have not yet found a routine [11], or occasions when people want to explore alternative ways to accomplish their tasks [16]. For example, parents may vary their routine when their child has a new scheduled activity that is not part of their current routines or when they unexpectedly have to pickup their child from an existing activity [11]. Understanding variations is also important to understand the tradeoffs between different behaviors that people have. For example, in designing systems that help people comply with their gym routine, it is important to understand what other activities cause people to depart from their gym routine.

However, existing machine learning algorithms (e.g., [5, 8, 18, 22]) purposefully disregard variations in human behavior to focus on classifying and predicting only the most frequent human activity. Also, some variations may happen infrequently in data and are difficult to detect using those existing algorithms. Some specific infrequent variations may be detectable (such as detecting when parents are going to be late to pickup their children [12]). However, this requires a case-by-case approach to address each kind of variation, which can be difficult to apply if all possible variations are not known *a priori*.

This leads to our second research question researchers want to answer: *RQVar: How can we expose variations of routines and enable the discovery of the cause of those behaviors?* Similar to our first question, routine-modeling approaches must identify aspects of context that influence how people behave in the face of variations.

Routines Across Individuals and Populations

The third issue to consider is how to explore the differences in routines within and across individuals and populations. Different people often develop their own routines to deal with their individual contexts. For example, differences in family daily routines can explain the responsibilities of different family members [11].

Also, to help people improve their routines it is important to understand the difference between peoples' desired routines and their actual routines. For example, to design systems that help people better organize their schedules to avoid being late to pickup their children requires understanding differences between routines for days when they are late and days when they are on time.

Such comparisons can be performed between routines of an individual, but also between routines of different people or populations. For example, to create interventions that help aggressive drivers improve their driving style requires an understanding of the differences between aggressive and non-aggressive driving routines [18]. Based on how close drivers are to aggressive or non-aggressive routines, they can be classified into aggressive and non-aggressive groups. Researchers can then identify routine behaviors that need to change to make aggressive drivers less aggressive overall.

Existing algorithms can, of course, be applied to individual or population models of routines. For example, a standard strategy is to build a model for each population and then compare them to establish the differences and similarities between the populations. Population models can then be visualized using existing approaches: Hierarchical Task Networks [24] can be visualized using existing Probabilistic Context-Free Grammar-based tools (e.g., [10]), and Arc Diagrams [38] can visualize temporal patterns extracted using T-patterns [26]. Those population models can then be coordinated and displayed across multiple views [33]. However, such visualization techniques are not yet widely adopted. Finding differences in models is still often based on intuition and expert domain knowledge, and limited to visually comparing the distributions of feature values.

This leads to our third question researchers want to answer: *RQComp: How can we support comparison of routines across individuals and populations?* Therefore, we need to ensure that different aspects of routines and routine variations that support comparison between different people and populations are included in the models.

MODELING HUMAN ROUTINE BEHAVIOR

In this section we present our approach to modeling human routine behavior that applies an existing decision-theoretic algorithm to the domain of routine modeling. To identify routine patterns (*RQPat*), we explicitly model the causal relationship between contexts in which different routines occur and the actions that people perform in those contexts. Unlike models that extract only the most frequent routines, our approach also models possible variations from those routines (*RQVar*), even in infrequent contexts. Our approach does this by modeling probability distributions over different possible behaviors, which allows the researcher to make sense about which of those behaviors form routines and which form variations. Our approach can model both individual and population routine behavior, and thus allows comparisons between those models (*RQComp*).

Data Modeling

Human behavior data is often collected using different sensors and stored into behavior logs. In our approach we first convert the behavior logs into sequences of events representing people's current context and the actions they perform in that context. We then use those sequences of events to model human routine behavior.

Model of Human Routine Behavior

We model demonstrated routine behavior using a Markov Decision Processes (MDP) framework [32]. MDP is particularly well suited for modeling human routine behavior because it explicitly models the user's context, the actions that can be performed in that context, and the preferences people have for different actions in different contexts. A Markov decision process is a tuple:

$$\mathcal{M}_{MDP} = (S, A, P(s'|s, a), R(s, a)) \quad (1)$$

It consists of a set of states S ($s \in S$) representing context, and actions A ($a \in A$) that a person can take. In addition, the model includes an action-dependent probability distribution for each state transition $P(s'|s, a)$, which specifies the probability of the next state s' when the person performs action a in state s . This state transition probability distribution $P(s'|s, a)$ models how the environment responds to the actions that people perform in different states. When modeling human behavior, the transitions are often stochastic (each pair (s, a) can transition to many transition states s' with different probabilities). However, if the person has full control over the environment, they can also be deterministic (*i.e.*, for each pair (s, a) there is exactly one transition state s' with probability 1). Finally, there is a reward function $R(s, a) \rightarrow \mathbb{R}$ that the person incurs when performing action a in state s , which represents the utility that people get from performing different actions in different contexts.

People’s behavior is then defined by sequences of actions they perform as they go from state to state until reaching some goal state. In an MDP framework, such behavior is defined by a deterministic policy ($\pi: S \rightarrow A$), which specifies actions people take in different states. Traditionally, the MDP is “solved” using algorithms, such as value iteration [6], to find an optimal policy (with the highest expected cumulative reward). However, our goal is to find the expected frequencies of different states and the probability distribution of actions given states ($P(a|s)$) instead—information necessary to identify people’s routines and variations.

Learning Routine Patterns from Demonstrated Behavior

In this section we explore how the MaxCausalEnt algorithm [46], an algorithm typically used to predict human behavior [43, 44], can be applied in a novel way to extract routine behavior patterns from observed data. MaxCausalEnt algorithm makes its predictions by computing a policy ($\pi: S \rightarrow A$) that best predicts the action people take in different states. Our main contribution is our insight that in the process of computing this policy, MaxCausalEnt algorithm computes two other functions that express how likely it is that a state and action are part of a routine: 1) the expected frequency of states (D_s), and 2) probability distribution of actions given states ($P(a|s)$). We now describe how we compute these two functions and how they relate to routines.

Inverse Reinforcement Learning (IRL) [29] approaches, which MaxCausalEnt is based on, assume that people assign a utility function (modeled as the reward functions $R(s, a)$), which they use to decide which action to perform in different demonstrated states. Each state and action combination in our MDP model is expressed by a feature vector $\mathcal{F}_{S,A}$. For example, in an MDP that models daily commute routines, states can have features that describe all possible locations that a person can be at, and actions can have features that describe if the person is staying at or

leaving the current location. As is common for IRL algorithms [29, 45], we assume a parametric reward function that is linear in $\mathcal{F}_{S,A}$, given unknown weight parameters θ :

$$R(s, a) = \theta^T \cdot \mathcal{F}_{s_t, a_t} \quad (2)$$

We begin the process of recovering the expected state frequencies (D_s) and probability distribution of actions given states ($P(a|s)$) by trying to learn the person’s reward functions $R(s, a)$ from demonstrated behavior. This problem reduces to matching the model feature function expectations ($E_{P(S,A)}[\mathcal{F}(S, A)]$) with demonstrated feature expectations ($E_{\tilde{P}(S,A)}[\mathcal{F}(S, A)]$) [1]. To match the expected counts of different features, we use MaxCausalEnt IRL [45], which learns the parameters of the MDP model to match the actual behavior of the person. Unlike other approaches described earlier, MaxCausalEnt explicitly models the causal relationships between context and actions, and keeps track of the probability distribution of different actions that people can perform in those contexts.

To compute the unknown parameters θ , MaxCausalEnt considers the causal relationships between all the different features of the states and the actions. The Markovian property of MDP, which assumes that the actions a person performs only depend on the information encoded by the previous state, makes computing the causal relationships between the states and actions computationally feasible. MaxCausalEnt extends the Principle of Maximum Entropy [19] to cases where information about probability distribution is sequentially revealed, as is the case with behavior logs. This principle ensures that the estimated probability distribution of actions given states ($P(a|s)$) is the one that best fits the state and action combinations from the sequences in the behavior logs.

MaxCausalEnt IRL maximizes the causal entropy ($H(\mathbf{A}^T \parallel \mathbf{S}^T)$) of the probability distribution of actions given states ($P(A_t|S_t)$):

$$\operatorname{argmax}_{P(A_t|S_t)} H(\mathbf{A}^T \parallel \mathbf{S}^T) \quad (3)$$

such that:

$$E_{P(S,A)}[\mathcal{F}(S, A)] = E_{\tilde{P}(S,A)}[\mathcal{F}(S, A)]$$

$$\forall_{S_t, A_t} P(A_t|S_t) \geq 0$$

$$\forall_{S_t, A_t} \sum P(A_t|S_t) = 1$$

The first constraint in the above equation ensures that the feature counts calculated using the estimated probability distribution of actions given states ($P(A_t|S_t)$) matches the observed counts of features in the data, and the other two ensure that $P(A_t|S_t)$ is an actual probability distribution.

Using the action-based cost-to-go (Q), which represents the expected value of performing action a_t in state s_t , and state-based value (V) notation, which represents the

expected value of being in state s_t , the procedure for MDP MaxCausalEnt IRL reduces to [45]:

$$Q_{\theta}^{soft}(a_t, s_t) = \sum_{s_{t+1}} P(s_{t+1}|s_t, a_t) \cdot V_{\theta}^{soft}(s_{t+1}) \quad (4)$$

$$V_{\theta}^{soft}(s_t) = \text{softmax}_{a_t} Q_{\theta}^{soft}(a_t, s_t) + \theta^T \cdot \mathcal{F}_{s_t, a_t}$$

Note that this is similar, but not the same as stochastic value iteration [6], which would model optimal and not observed behavior. The probability distribution of actions given the states is then given by:

$$P(a_t|s_t) = e^{Q_{\theta}^{soft}(a_t, s_t) - V_{\theta}^{soft}(s_t)} \quad (5)$$

The probability distribution of actions given states $P(a|s)$ and the state transition probability distribution $P(s'|s, a)$ are used in a forward pass to calculate the expected state frequencies (D_s). This optimization problem can then be solved using a gradient ascent algorithm. Ziebart [46] provides proofs of these claims and detailed pseudocode for the algorithm above.

Extracting Models of Routine Behavior

We illustrate our routine modeling approach on two previously collected data sets from the literature that contain logs of demonstrated human behavior. The first data set contains daily commute routines of all family members from three two-parent families with children from a mid-sized city in North America [11]. The data set was used to predict the times the parents are likely to forget to pickup their children [12]. The other data set contains driving routine behavior of aggressive and non-aggressive drivers as they drive on their daily routes [18]. The data set was used to classify aggressive and non-aggressive drivers.

We picked these two data sets to show the generalizability of our approach to different types of routines. The two data sets contain routine tasks people perform on a daily basis, but that are very different in nature. The family daily routine data set incorporates the traditional spatio-temporal aspect of routines most of the existing work focuses on. The driving data set contains situational routines that are driven by other types of context (*e.g.*, the surrounding traffic, the current position of the car in the intersection).

The two data sets also differ in granularity of the tasks. The commute routines happen over a longer period of time and the granularity of the task is very coarse with few actions that people can perform in different contexts (*e.g.*, stay at the current place or leave and go to another place). The daily routines are therefore defined by the states the people are in. The aggressive driving data set contains fine-grained actions, which often occur in parallel, that people perform to control the vehicle (*e.g.*, control the gas and brake pedals and the steering wheel). Driving routines are therefore primarily defined by the drivers' actions in different driving situations. The driving data set also showcases the ability of our approach to capture population models (*e.g.*, aggressive drivers *vs.* non-aggressive drivers) and enable comparison of routines across different populations.

The data in our two example data sets consists of sequences of sensor readings, which we convert into sequences of events represented by state action pairs. Parsing the raw data we are able to extract: 1) a set of states S defined by a set of features \mathcal{F}_{s_t} which represent context, 2) a set of actions A defined by a list of binary features \mathcal{F}_{a_t} which represent activities that the people can perform, and 3) empirically estimated state-action transition dynamics ($P(s'|s, a)$). At any discrete event step, the state features contain values of all the contextual sensor readings at that event, and actions contain feature values describing the activity the people performed at that event. We then estimate state-action transition dynamics based on the frequencies of state transitions in the state-action event sequences and estimate the expected state frequency counts (D_s) and the state-action probability distributions ($P(a|s)$) as described in the previous section.

Family Daily Routines Data Set

Situations when one of the parents is unable to pickup or drop-off a child create stress for both parents and children [11]. To better understand the circumstances under which these situations arise, it is important to identify when the parents are responsible for picking up and dropping off their children (*RQPat*), when variations from normal routines occur and how parents handle those situations (*RQVar*). This requires finding and understanding how the parents organize their daily routines around those pickups and drop-offs (*RQComp*).

This data contains location sampling (latitude and longitude) at one-minute intervals for every family member (including children) in three families from a mid-sized city in North America [11]. Location information was manually labeled based on information from bi-weekly interviews with participants. Participants also provided information about their actual daily routines during those interviews.

We converted the location logs into sequences of states and actions representing each individual's daily commute for each day in the data set. State features included the day of the week, hour of the day, participant's current place, and whether the participant stayed at the location from the previous hour, arrived at the location during the current hour, or left the location during the hour (Table 1). Action features included the participant's current activity that could be performed in those states (Table 2). Participants could stay for another hour, leave the location, and once they have left a location go to another location. The data contained a total of 149 days.

We modeled the state transition probabilities ($P(s'|s, a)$) as a stochastic MDP to model the environment's influence on arrival time to a destination. The participants could stay or leave a place with 100% probability. Once the participants leave their current location, their arrival time at their destination depends on their desired arrival time and the environment (*e.g.*, traffic, travel distance). To model the

Table 1. State features capturing the different contexts of a daily commute.

Feature	Description
Day	Day of week {M, T, W, Th, F, Sa, Su}
Time	Time of day in increments of 1 hour {0-23}
Location	Current location
Activity	Activity in the past hour {STAYED AT, ARRIVED AT, TRAVELING FROM}

Table 2. Action features representing actions that people can perform when at a location.

Feature	Description
Activity	Activity people can perform in current context {STAY AT, TRAVEL TO}
Location	The current location to stay at or next location to go to

influence of the external variables, we empirically estimate the probability that participants have arrived at another place within an hour or not. The median number of states and actions per family were 14,113 and 85 respectively, for all combinations of possible features.

Aggressive Driving Behavior Data Set

Drivers that routinely engage in aggressive driving behavior present a hazard to other people in traffic [3]. To understand aggressive driving routines, it is important to explore the types of contexts aggressive drivers are likely to prefer (e.g., turn types, car speed, acceleration) and the driving actions they apply in those contexts (e.g., throttle and braking level, turning) (*RQPat*). Aggressive drivers might also be prone to dangerous driving behavior that does not occur frequently (e.g., rushing to clear intersections during rush hour [36]). Such behavior might manifest itself as variations from established routines (*RQVar*).

It is also important to compare the routines of aggressive drivers with non-aggressive drivers to understand how aggressive drivers can improve their routine (*RQComp*). To understand those differences, it is not enough to compare the contexts both groups of drivers find themselves in, but also the actions that drivers perform in those contexts. This is because both aggressive and non-aggressive drivers can attain similar driving contexts, but the quality of the execution of driving actions may differ. For example, both types of drivers might stop at a stop sign on time, but aggressive drivers might have to brake harder or make more other unsafe maneuvers than non-aggressive drivers.

This data set contains driving data from 22 licensed drivers (11 male and 11 female; ages between 21 and 34) from a mid-sized city in North America [18]. Participants were asked to drive their own cars on their usual daily driving routes over a period of 3 weeks. Their cars were instrumented with a sensing platform consisting of an Android-based smartphone, On-board Diagnostic tool (OBD2), and an inertial measurement unit (IMU) mounted to the steering wheel of the car. Ground truth about participants' driving styles (aggressive vs. non-aggressive) was established using their self-reported driving violations and responses to the driver behavior questionnaire [18]. The

driving data collected in the study included: car location traces (latitude and longitude), speed, acceleration, engine RPM, throttle position, and steering wheel rotation. Sensor data was recorded every 500 milliseconds.

We use a subset of this data focused on intersections (where instances of aggressive driving are likely to occur [36]). We used location traces of the participants' driving routines to manually label intersections and the position of the vehicle in those intersections. One of the limitations of this data set is that there is no information about other vehicles and traffic signs and signals that represent the environment. We then split the intersection instances into sequences of sensor readings that start 2 seconds before the car enters the intersection, and end 2 seconds after the car exits the intersection. This resulted in a total of 49,690 intersections from a total of 542 hours of driving data from 1,017 trips.

To model states we combined the driver's goals (e.g., make a right turn), the environment (e.g., position in intersection), and the current state of the vehicle (e.g., current speed) into features of the states (Table 3). Actions in our model represent how the driver operates the vehicle by steering the wheel, and depressing the gas (throttle) and brake pedals. We aggregate the driver's actions between different stages of the intersection and represent the median throttle and braking level, and note any spikes in both throttle and braking. We consider the movement of the steering wheel to estimate whether the driver turned in one smooth action, or if the turn required one or more adjustments. Table 4 shows action features in our model. We identified 7,272 different states and 446 different actions in the data set.

Table 3. State features capturing the different contexts the driver can be in.

Feature	Description
Goals	
Maneuver	The type of maneuver at the intersection {STRAIGHT, RIGHT TURN, LEFT TURN, U-TURN}
Environment	
Position	Current position of the car in the intersection {APPROACHING, ENTERING, EXITING, AFTER}
Rush hour	Whether the trip is during rush hour or not {TRUE, FALSE}
Vehicle	
Speed	Current speed of the vehicle (5-bin discretized)
Throttle	Current throttle position (5-bin discretized)
Acceleration	Current positive/negative acceleration (9-bin discretized)
Wheel Position	Current steering wheel position {STRAIGHT, TURNING, RETURNING}
Turn	Current turn vehicle is involved in {STRAIGHT, SMOOTH, ADJUSTED}

Table 4. Action features representing actions that drivers can perform between stages of the intersection.

Feature	Description
Pedal	Median throttle (gas and brake pedal) position (10-bin discretized)
Throttle Spike	Sudden increases in throttle {NONE, SUDDEN, INTERMITTENT}
Brake Spike	Sudden braking {NONE, SUDDEN, INTERMITTENT}
Turn style	Type of turn driver performed in intersection {STRAIGHT, SMOOTH, ADJUSTED}

VALIDATING THE MODELS OF ROUTINE BEHAVIOR

In this section, we evaluate the quality of the routines extracted using our model from the two data sets. First, we show that the routine actions we extract are predictive of the majority of behaviors in the data; *i.e.*, that the algorithm is sufficiently predictive for modeling routines. Accuracy of this prediction task also quantifies the variability of the routines in the model, where high accuracy suggests low variability. It also shows that the extracted routines generalize to contexts and actions that we have not observed during model training. Second, we show that the routines extracted using our approach are meaningful. We show that the patterns extracted using our approach correspond to the actual routines and routine variations in our two example behavior logs (*RQPat* & *RQVar*). We also show that those routines and variations show the real differences between modeled populations (*RQComp*).

Quantifying Routineness of Human Behavior

Although we are not interested in the predictive power of the MaxCausalEnt IRL *per se*, we use the task of predicting the next action given a state to evaluate our model’s ability to extract routine. Using 10-fold cross validation for each person in each dataset, we compare the performance of this algorithm for extracting routine behavior with a simple Zero-R algorithm, which always predicts the overall most frequent action, and a first-order Markov Model algorithm, which always predicts the most frequent action for each state. We chose these two baselines because they explicitly establish the frequency of actions in the training set. Matching or exceeding these baselines means that the algorithm has correctly identified frequent routine actions and that the predictive power of the algorithm is sufficiently high to model routines.

The mean accuracy of the MaxCausalEnt on the family daily routines dataset was 0.81 (SD=0.09), compared to first-order Markov Model mean accuracy of 0.66 (SD=0.07) and ZeroR mean accuracy of 0.51 (SD=0.09). MaxCausalEnt algorithm likely outperformed the first-order Markov Model because of its ability to better generalize from training data. The accuracy of MaxCausalEnt algorithm also suggests low variability of routines in people’s daily schedules.

The mean accuracy of the MaxCausalEnt on individual models of driving routines was 0.54 (SD=0.05) compared to first-order Markov Model mean accuracy of 0.58 (SD=0.06) and ZeroR mean accuracy of 0.33 (SD=0.06). MaxCausalEnt algorithm and the first-order Markov Model had similar accuracies likely because in each fold the training set was representative of the testing set. However, decision-theoretic guarantees of MaxCausalEnt that ensure it makes the least number of assumptions to fit the observed data make it less likely to overfit the training data in general. Relatively low accuracy of both MaxCausalEnt and the first-order Markov Model on this data set suggests that there is a lot of variability in the driving routines.

Validating the Quality of the Extracted Routine

We now show that the routine patterns extracted using our approach match the actual routines of people. To do this, we recruited researchers that work with machine learning and data mining in the domain of human behavior, and asked them to identify the routines and variations extracted using our approach. We then confirmed that those patterns matched the ground truth behaviors established in the previous work [11, 18]. This allowed us to verify that the patterns extracted using our approach are meaningful and represent the actual routines.

Study Software

To make the routine behavior models created using our approach accessible to participants and allow them to investigate the extracted routine patterns, we developed a simple visualization tool. To maintain a level of familiarity, we base our visual encoding of routine behavior elements on a traditional visual representation of an MDP as a graph (Figure 1). Our MDP graph contains nodes representing states (as circles) and actions (as squares), directed edges from state nodes to action nodes (indicating possible actions people can perform in those states), and directed edges from actions to states (indicating state transitions for any given state and action combination).

To enable participants to see changes in features of states and actions, we encode state features and action features as a series of color-coded circular marks arranged in a spiral shape within the nodes. Each feature has a dedicated hue. Feature values that are present in the node are represented by a dark shade, and feature values not present in a light shade of that color. A dark boundary serves as a separator between features. More details, in text are always available simply by moving the cursor over a node (Figure 1.C).

To show frequent behaviors in the model, we visually represent the probability of different graph elements using line thickness. Thickness of the outside line of the state and action nodes encodes the frequency of that state in a behavior sequence (D_s), where thicker lines indicate states that are likely to be part of a routine. Similarly, the thickness of the edges encodes the probability of that edge. Thickness of edges from states to actions is given by the probability distribution of actions given states ($P(a|s)$), and represents the influence of each state on the choice of actions. The thickness of edges from actions to states is given by the probability of transition ($P(s'|s, a)$).

To layout the nodes, we sort the initial states from the demonstrated sequences by their frequency (D_s) in descending order. We then use a version of the depth-first search algorithm, starting from the initial state nodes, that traverses nodes by first iterating over edges in order from highest to lowest probabilities $P(a|s)$ and $P(s'|s, a)$. State nodes are never duplicated (*i.e.*, there is exactly one node in the layout for each state in the model), whereas action nodes are duplicated for each state.

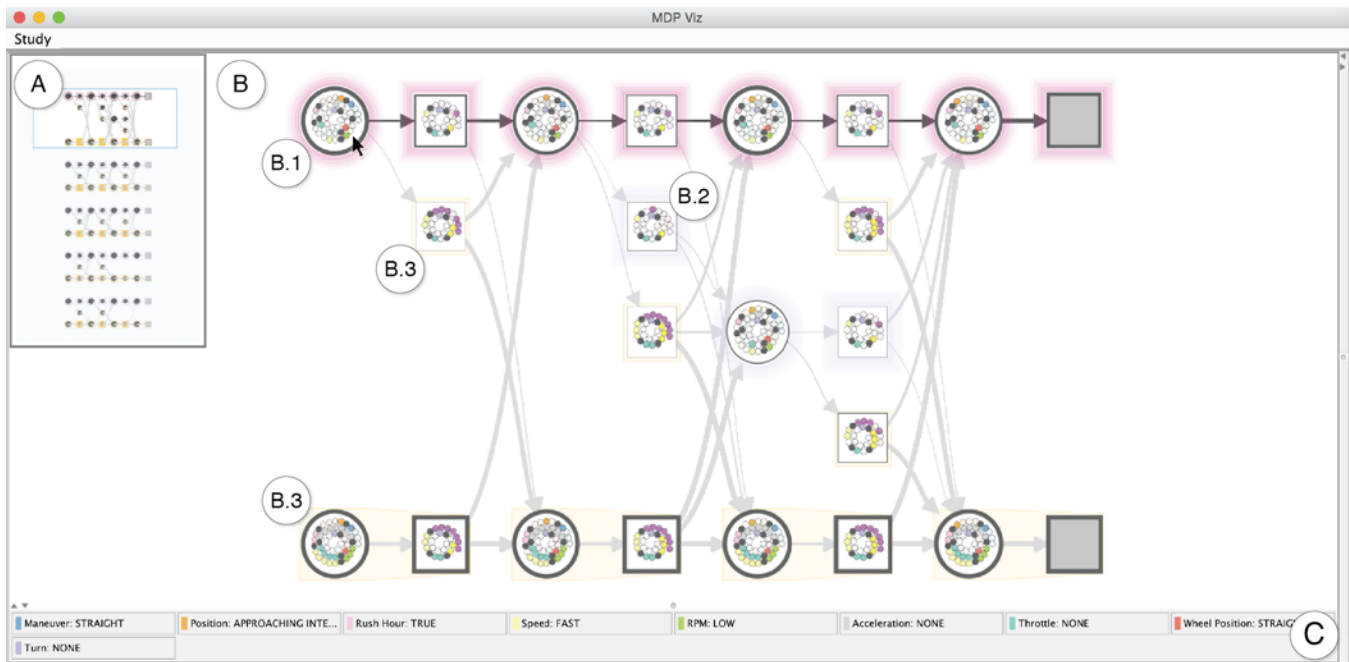


Figure 1. Study software user interface showing the main routine and one likely variation of non-aggressive drivers extracted using our approach: A) overview panel, B) the main display area containing subgraphs representing automatically extracted routine sequences of states (circles) and actions (squares), B.1) the user is hovering over a node to highlight extracted routine (nodes highlighted in purple and dark gray edges), B.2) an action that starts a variation from the main routine B.3) aggregate items representing possible extracted variations, and C) details panel showing information about visual elements on demand.

When the participant selects a data set and population, the tool provides the initial layout of routines extracted using our algorithm. This shows the most important information about the extracted routines. However, to further analyze the routine behavior, the participant must be able to explore the details of routine variations filtered out as aggregate nodes (Figure 1.B.3). For example, the participant might want to find which of the parents' routines include locations where they pickup and drop off their children.

Aggregated items contain valuable information about potential routine variations. For example, a child might go to her grandparents or their friend's house on Wednesdays after school; two variations on the same routine that occur with similar probability. To show all possible variations in an aggregate, the participant can click on the aggregate to expand its content. To mark an aggregated node as a variation of interest, the researcher can *pin* that aggregated node by clicking on it, thus removing it from all of its aggregate parents. When the researcher holds Alt-key and clicks on an aggregated node, this pins all the nodes on the most likely sequence of states and actions, determined by the probabilities of edges between the two. The pinned sequence, starting from the clicked node to the sequence end node, represents a routine variation. Pinned nodes are identified by a gray glow effect. All nodes that are part of the extracted routine are automatically pinned, and all other nodes are unpinned. Clicking on a pinned node unpins it, which returns the node into the aggregate.

To determine whether or not to pin the node, the researcher can review the features of individual and aggregate nodes by hovering over them. In addition to showing the details of individual nodes in the details panel (Figure 1.C), hovering over nodes shows relationships between different elements of the routine. Hovering over any node highlights the most likely routine path from an initial state to the collecting node that contains the hovered over node (Figure 1.B.1). This makes it easier to understand the routine states and actions in the area of interest.

Participants

To verify the routines, we have recruited 8 researchers (5 male and 3 female) that have had experience with machine learning and data mining, or have worked in the domain of activity recognition and human routine modeling. The participants included Ph.D. students, Postdoctoral fellows, Research Scientists, and Professors working or visiting at our local University. All participants had experience with machine learning, 2 with data mining, 4 with activity recognition, and 1 worked specifically on modeling human routine behavior. The participants were compensated \$25 for taking part in the study.

Method

When participants arrived at our lab, we briefed them on the purpose of the study and they signed a consent form. They then filled out a short questionnaire asking them about their occupation and experience with relevant research topics. We then demonstrated the visual tool to the participants and allowed them to practice using it for

approximately 20 minutes. Participants had to complete two tasks: 1) identify daily routine for a randomly chosen person and weekday from the daily routines data set, and 2) identify the differences between routines of non-aggressive and aggressive drivers from the driving data set. The first task took approximately 20 minutes, and the second task about an hour. Total study time was approximately between one and a half and two hours.

For the two tasks, we asked participants to identify routines and differences between the routines without presenting them with the ground truth. We did this to avoid biasing the participants towards trying to match the routines presented in the tool with what we might have told them is the correct answer. We then compared their answers with the ground truth from the previous literature to verify that the tool extracted the right and meaningful routine. Because the main purpose of the study was to validate the extracted routines and not evaluate the usability of the tool, participants could ask clarifying questions about the tool and the user interface at any point during the tasks.

Ground Truth

In the first task, we used the family daily commute routines model to understand the patterns of pickups and drop-offs. We compare the findings of our participants with the ground truth and discussion provided by Davidoff *et al.* [11]. In this data set, the ground truth represents self-reported daily commute routines for all family members in all families that took part in the study. Family members reported their location and the time they usually arrive and leave that location. Davidoff *et al.* [11] then manually annotated and confirmed the routines in the raw sensor data.

In the second task, we used the driving routines of aggressive and non-aggressive drivers that Hong *et al.* [18] identified in their data set. Hong *et al.* [18] used their intuition and expert knowledge of driving behaviors to separately compare the distributions of each sensor stream in the raw data to gain insight about aggressive driving styles. They found that aggressive drivers drive faster than non-aggressive drivers, and that they experience higher acceleration than non-aggressive driver (*i.e.*, they are more likely to press hard on the gas and brake pedals). Additionally, they found more variability in the behavior of aggressive drivers than non-aggressive drivers.

Results

Our results show that our approach extracts meaningful patterns of routine behavior. The participants were able to point out the patterns that form the high-level routines present in the ground truth for both tasks (*RQPat*). For the daily routines task, this means that they successfully listed the locations and times of the routines of people in the daily routine data set. However, two participants identified two separate patterns where the locations and times reported as part of the main daily routine did not correspond to the ground truth. After careful examination, we found that the participants wrongfully identified the actions as part of the

routines because the events were part of infrequent routine variations that the people in the original study [11] did not report in the ground truth. The algorithm correctly assigned low probabilities to those actions, but the participants did not notice this. This is likely an issue with the visualization rather than the algorithm itself, and is something that can be addressed with more training with the tool.

In addition to simply pointing to patterns that represented correct routines, participants also generated some insights for themselves. For example, all six participants that were presented with a parent's daily routine that contained a child pickup or drop-off specifically pointed out this activity. Also, three participants, that had a case where the parent drops off the child as part of his or her routine, but does not also pick the child up, correctly explained that the other parent was likely responsible for the pickup, without seeing the other parent's routines.

In the driving data set, participants pointed to the patterns that form the main routines (*RQPat*) and variations in driving behavior (*RQVar*) of both aggressive and non-aggressive drivers. All participants pointed to patterns that show that aggressive drivers are more likely to drive faster through intersections than non-aggressive drivers. Five participants showed the patterns of routine variations where aggressive drivers are likely to increase their throttle just before entering and leaving intersections. Participants pointed those out as the main differences between the two populations (*RQCom*). Two participants also pointed to the probabilities of routine variation patterns extracted using our approach that suggest that aggressive drivers are less consistent in their behavior than non-aggressive drivers. Participants likely drew their conclusions from the model, but might also have a preconceived notion that acceleration and speed are correlated with aggressive driving. However, even if our participants had preconceived notions, they could verify and document them using our model.

Although evaluation of the visualization tool was not our main goal, 2 participants mentioned that such a tool would help them explore and understand human routines. One participant, who studies routine behaviors, pointed out that the organization of routine patterns and variations helped him clarify his understanding of what constitutes routines and how they manifest themselves in people's activity.

DISCUSSION

Through our evaluation, we showed that our models trained using MaxCausalEnt algorithm [45] can extract patterns of routine behavior from demonstrated behavior logs. This is a novel application of an algorithm that was designed to predict human behavior. In our evaluation of the algorithm we found that its ability to predict routines from the two example data sets was sufficient for modeling routines. The ability of MaxCausalEnt algorithm to generalize from small sample sizes enabled it to beat the baseline in the daily routine data set. The performance of the algorithm was comparable with the first-order Markov Model in the

aggressive driving data set. This is likely because the training data happened to match the testing data well. However, this is not safe to assume in general case, and MaxCausalEnt’s decision-theoretic guarantee that it will not overfit the observed data make it a better choice for modeling routines than the first-order Markov Model.

Although not our main contribution, our visualization tool also helped us validate the ability of our approach to extract meaningful routines. The participants were able to explore context, actions, and the relationships between the two, to correctly identify the patterns of routines (*RQPat*) and their variations (*RQVar*) reported in the previous work. They pointed to these relationships to establish the differences between routines of the two driver populations (*RQComp*). Although we carefully designed our tool, our goal was not to formally evaluate its usability. We did not notice any usability issues that prevented participants from learning the elements of the model. We found that the participants knew how to progress towards understanding the routines.

Our results imply that researchers can use the patterns extracted using our approach to more quickly identify major aspects of routines by visually inspecting them, even after only short amount of training, compared to previous work. For example, Davidoff *et al.* [11] performed tedious manual labeling of routine and routine variation patterns in the raw data based on feedback from the participants before presenting the patterns on a timeline. Hong *et al.* [18] used their intuition and expert knowledge of driving behaviors to separately compare the distributions of each sensor stream in the raw data to gain insight about aggressive driving styles. Our participants had to only explore the patterns extracted using our approach.

The knowledge that the researchers gain about routine behaviors through exploring our models can inform the design of interventions that help people improve their routines. For example, the knowledge that aggressive drivers are likely to use higher throttles can inform the design of in-car systems that monitor the throttle and make the driver more aware of this aggressive behavior through subtle ambient notifications. Another advantage of our approach is the underlying MDP-based model, which can be used to power smart agents that automatically classify current behaviors and prescribe new actions that improve existing routines.

CONCLUSION AND FUTURE WORK

We presented a novel approach for modeling human routine behavior from behavior logs that explicitly models one of the most important aspects of routines: the causal relationship between the contexts and actions the people perform in those contexts. We demonstrated that our approach can be used to extract meaningful routine patterns from two different types of human behaviors. However, future work should explore how our approach can be used to model routines in other domains (*e.g.*, health, accessibility, software user interfaces).

We showed that routine models extracted using our approach can help researchers to identify routines and routine variations without having to manually search for those patterns in raw data. This is a step towards models that enable researchers to form scientific insight and generate knowledge about human behavior that will help them develop new theories about human routines. Having such knowledge can also inform the design of novel technologies that help people improve the quality of their routine behavior.

Our long-term goal is to help researchers in the process of generating knowledge that will support the design of technologies for routine behavior change. In this work, we presented a simple visual representation of the model extracted using our approach. However, in future work, we intend to develop a visual analytics [21] tool to offer a framework for visual data mining and understanding of human routine behaviors. We also intend to show how researchers and designers can use the insights they generate from our models to design technologies that help people improve their routines.

Machine understanding of human routine behaviors is also necessary to design and implement effective technologies that help people improve their routines. Our approach encodes patterns of routine behavior in a way that allows systems, such as smart agents, to classify, predict, and reason about human actions under the inherent uncertainty present in human behavior. For example, future smart agents can use our model to detect aggressive driving behavior and provide feedback to the driver based on the models of non-aggressive drivers. Such technologies can have a positive effect on society by making people healthier, safer, and more efficient in their routine tasks.

ACKNOWLEDGMENTS

This work was funded by NSERC (PGSD3-438429-2013) and NSF (CCF-1029549, IIS-1217929). The authors would like to thank Brian Ziebart for his valuable input regarding the MaxCasualEnt algorithm, Scott Davidoff and Jin-Hyuk Hong for their help in obtaining and understanding the two datasets used in our work, and Julian Ramos and Christine Bauer for discussions about human routine behavior.

REFERENCES

1. Pieter Abbeel and Andrew Y. Ng. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning (ICML '04)*. ACM, New York, NY, USA, 1-. <http://doi.acm.org/10.1145/1015330.1015430>
2. Wolfgang Aigner, Silvia Miksch, Bettina Thurnher, and Stefan Biffl. 2005. PlanningLines: Novel Glyphs for Representing Temporal Uncertainties and Their Evaluation. In *Proceedings of the Ninth International Conference on Information Visualisation (IV '05)*. IEEE Computer Society, Washington, DC, USA, 457-463. <http://dx.doi.org/10.1109/IV.2005.97>

3. American Automobile Association. 2009. Aggressive driving: Research update. *American Automobile Association Foundation for Traffic Safety*.
4. John R. Anderson, Daniel Bothell, Michael D. Byrne, Scott Douglass, Christian Lebiere, and Yulin Qin. 2004. An integrated theory of the mind. *Psychological review*, 111(4), 1036-1060. <http://dx.doi.org/10.1037/0033-295X.111.4.1036>
5. Mitra Baratchi, Nirvana Meratnia, Paul J. M. Havinga, Andrew K. Skidmore, and Bert A. K. G. Toxopeus. 2014. A hierarchical hidden semi-Markov model for modeling mobility data. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14)*. ACM, New York, NY, USA, 401-412. <http://doi.acm.org/10.1145/2632048.2636068>
6. Richard Bellman. 1957. A Markovian decision process. *Journal of Mathematics and Mechanics*, 6, 679-684.
7. Oliver Brdiczka, Norman Makoto Su, and James Bo Begole. 2010. Temporal task footprinting: identifying routine tasks by their temporal patterns. In *Proceedings of the 15th international conference on Intelligent user interfaces (IUI '10)*. ACM, New York, NY, USA, 281-284. <http://doi.acm.org/10.1145/1719970.1720011>
8. Andreas Bulling, Ulf Blanke, and Bernt Schiele. 2014. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Comput. Surv.* 46, 3, Article 33 (January 2014), 33 pages. <http://doi.acm.org/10.1145/2499621>
9. Paolo Buono, Aleks Aris, Catherine Plaisant, Amir Khella, and Ben Shneiderman. 2005. Interactive pattern search in time series. In *Electronic Imaging 2005*, 175-186. <http://dx.doi.org/10.1117/12.587537>
10. Christopher Collins, Sheelagh Carpendale, and Gerald Penn. 2007. Visualization of uncertainty in lattices to support decision-making. In *Proceedings of the 9th Joint Eurographics / IEEE VGTC conference on Visualization (EUROVIS'07)*, Ken Museth, Torsten Möller, and Anders Ynnerman (Eds.). Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 51-58. <http://dx.doi.org/10.2312/VisSym/EuroVis07/051-058>
11. Scott Davidoff, John Zimmerman, and Anind K. Dey. 2010. How routine learners can support family coordination. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 2461-2470. <http://doi.acm.org/10.1145/1753326.1753699>
12. Scott Davidoff, Brian D. Ziebart, John Zimmerman, and Anind K. Dey. 2011. Learning patterns of pick-ups and drop-offs to support busy family coordination. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 1175-1184. <http://doi.acm.org/10.1145/1978942.1979119>
13. Anind K. Dey. 2001. Understanding and Using Context. *Personal Ubiquitous Comput.* 5, 1 (January 2001), 4-7. <http://dx.doi.org/10.1007/s007790170019>
14. Nathan Eagle and Alex S. Pentland. 2009. Eigenbehaviors: identifying structure in routine. *Behav. Ecol. Sociobiol.*, vol. 63, no. 7, 1057-1066. <http://dx.doi.org/10.1007/s00265-009-0739-0>
15. Katayoun Farrahi and Daniel Gatica-Perez. 2012. Extracting Mobile Behavioral Patterns with the Distant N-Gram Topic Model. In *Proceedings of the 2012 16th Annual International Symposium on Wearable Computers (ISWC) (ISWC '12)*. IEEE Computer Society, Washington, DC, USA, 1-8. <http://dx.doi.org/10.1109/ISWC.2012.20>
16. Martha S. Feldman and Brian T. Pentland. 2003. Reconceptualizing organizational routines as a source of flexibility and change. *Administrative Science Quarterly*, 48 (1), 94-118. <http://dx.doi.org/10.2307/3556620>
17. Geoffrey M. Hodgson. 1997. The ubiquity of habits and rules. *Cambridge Journal of Economics*, 21(6), 663-684.
18. Jin-Hyuk Hong, Ben Margines, and Anind K. Dey. 2014. A smartphone-based sensing platform to model aggressive driving behaviors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 4047-4056. <http://doi.acm.org/10.1145/2556288.2557321>
19. Edwin T. Jaynes. 1957. Information theory and statistical mechanics. *Physical review*, 106(4), 620. <http://dx.doi.org/10.1103/PhysRev.106.620>
20. Jin Jing and Pedro Szekely. 2010. Interactive querying of temporal data using a comic strip metaphor. In *2010 IEEE Symposium on Visual Analytics Science and Technology (VAST)*, 163-170. <http://dx.doi.org/10.1109/VAST.2010.5652890>
21. Daniel A. Keim, Florian Mansmann, Jorn Schneidewind, and Hartmut Ziegler. 2006. Challenges in Visual Data Analysis. In *Proceedings of the conference on Information Visualization (IV '06)*. IEEE Computer Society, Washington, DC, USA, 9-16. <http://dx.doi.org/10.1109/IV.2006.31>
22. John Krumm and Eric Horvitz. 2006. Predestination: inferring destinations from partial trajectories. In *Proceedings of the 8th international conference on Ubiquitous Computing (UbiComp'06)*, Paul Dourish and Adrian Friday (Eds.). Springer-Verlag, Berlin, Heidelberg, 243-260. http://dx.doi.org/10.1007/11853565_15
23. Ian Li, Anind Dey, and Jodi Forlizzi. 2010. A stage-based model of personal informatics systems.

- In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '10). ACM, New York, NY, USA, 557-566.
<http://doi.acm.org/10.1145/1753326.1753409>
24. Nan Li, William Cushing, Subbarao Kambhampati, and Sungwook Yoon. 2014. Learning Probabilistic Hierarchical Task Networks as Probabilistic Context-Free Grammars to Capture User Preferences. *ACM Trans. Intell. Syst. Technol.* 5, 2, Article 29 (April 2014), 32 pages. <http://doi.acm.org/10.1145/2589481>
 25. Brian Y. Lim and Anind K. Dey. 2011. Investigating intelligibility for uncertain context-aware applications. In *Proceedings of the 13th international conference on Ubiquitous computing*(UbiComp '11). ACM, New York, NY, USA, 415-424.
<http://doi.acm.org/10.1145/2030112.2030168>
 26. Magnus S. Magnusson. 2000. Discovering hidden time patterns in behavior: T-patterns and their detection. *Behavior Research Methods, Instruments, & Computers*, 32(1), 93-110.
<http://dx.doi.org/10.3758/BF03200792>
 27. Neal Martin. 2008. Habit: the 95% of behavior marketers ignore. First edition. FT Press: NJ.
 28. Megan Monroe, Rongjian Lan, Hanseung Lee, Catherine Plaisant, and Ben Shneiderman. 2013. Temporal Event Sequence Simplification. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (December 2013), 2227-2236.
<http://dx.doi.org/10.1109/TVCG.2013.200>
 29. Andrew Y. Ng and Stuart J. Russell. 2000. Algorithms for Inverse Reinforcement Learning. In *Proceedings of the Seventeenth International Conference on Machine Learning* (ICML '00), Pat Langley (Ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 663-670.
 30. Wanda J. Orlikowski and JoAnne Yates. 2002. It's About Time: Temporal Structuring in Organizations. *Organization Science* 13, 6 (November 2002), 684-700.
<http://dx.doi.org/10.1287/orsc.13.6.684.501>
 31. Catherine Plaisant, Brett Milash, Anne Rose, Seth Widoff, and Ben Shneiderman. 1996. LifeLines: visualizing personal histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '96), Michael J. Tauber (Ed.). ACM, New York, NY, USA, 221-227.
<http://doi.acm.org/10.1145/238386.238493>
 32. Martin L. Puterman. 2014. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, Inc., New York, NY, USA.
 33. Jonathan C. Roberts. 2007. State of the Art: Coordinated & Multiple Views in Exploratory Visualization. In *Proceedings of the Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization* (CMV '07). IEEE Computer Society, Washington, DC, USA, 61-71.
<http://dx.doi.org/10.1109/CMV.2007.20>
 34. David L. Ronis, J. Frank Yates, and John P. Kirscht, 1989. Attitudes, decisions, and habits as determinants of repeated behavior. A.R. Pratkanis, S.J. Breckler, & A.G. Greenwald, Eds., Attitude structure and function, Lawrence Erlbaum: NJ, 213-239.
 35. Adam Sadilek and John Krumm. 2012. Far Out: Predicting Long-Term Human Mobility. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence* (AAAI '12), 814-820.
 36. David Shinar and Richard Compton. 2004. Aggressive driving: an observational study of driver, vehicle, and situational variables, *Accident Analysis & Prevention*, Volume 36, Issue 3, May 2004, 429-437.
[http://dx.doi.org/10.1016/S0001-4575\(03\)00037-X](http://dx.doi.org/10.1016/S0001-4575(03)00037-X)
 37. Richard Taylor. 1950. Purposeful and non-purposeful behavior: A rejoinder. *Philosophy of Science*, 17, 4. The University of Chicago Press on behalf of the Philosophy of Science Association, 327-332.
 38. Martin Wattenberg. 2002. Arc diagrams: Visualizing structure in strings. In *Proceedings of IEEE Symposium on Information Visualization* (INFOVIS 2002). IEEE, 110-116.
<http://dx.doi.org/10.1109/INFVIS.2002.1173155>
 39. Marc Weber, Marc Alexa, and Wolfgang Müller. 2001. Visualizing time-series on spirals. In *Proceedings of IEEE Symposium on Information Visualization* (INFOVIS 2001). IEEE, 7-14.
<http://doi.ieeecomputersociety.org/10.1109/INFVIS.2001.963273>
 40. Krist Wongsuphasawat, John Alexis Guerra Gómez, Catherine Plaisant, Taowei David Wang, Meirav Taieb-Maimon, and Ben Shneiderman. 2011. LifeFlow: visualizing an overview of event sequences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '11). ACM, New York, NY, USA, 1747-1756.
<http://doi.acm.org/10.1145/1978942.1979196>
 41. Jian Zhao, Fanny Chevalier, and Ravin Balakrishnan. 2011. KronoMiner: using multi-foci navigation for the visual exploration of time-series data. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '11). ACM, New York, NY, USA, 1737-1746.
<http://doi.acm.org/10.1145/1978942.1979195>
 42. Jian Zhao, Fanny Chevalier, Emmanuel Pietriga, and Ravin Balakrishnan. 2011. Exploratory analysis of time-series with chronolenses. In *IEEE Transactions on Visualization and Computer Graphics*, 17, 12. IEEE, 2422-2431.
<http://dx.doi.org/10.1109/TVCG.2011.195>

43. Brian D. Ziebart, Andrew L. Maas, Anind K. Dey, and J. Andrew Bagnell. 2008. Navigate like a cabbie: probabilistic reasoning from observed context-aware behavior. In *Proceedings of the 10th international conference on Ubiquitous computing (UbiComp '08)*. ACM, New York, NY, USA, 322-331. <http://doi.acm.org/10.1145/1409635.1409678>
44. Brian D. Ziebart, Nathan Ratliff, Garratt Gallagher, Christoph Mertz, Kevin Peterson, J. Andrew Bagnell, Martial Hebert, Anind K. Dey, and Siddhartha Srinivasa. 2009. Planning-based prediction for pedestrians. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009)*, 3931-3936. <http://dx.doi.org/10.1109/IROS.2009.5354147>
45. Brian D. Ziebart, J. Andrew Bagnell, and Anind K. Dey. 2010. Modeling interaction via the principle of maximum causal entropy. In *Proceedings of the 27th international conference on Machine learning (ICML '10)*. ACM, New York, NY, USA., 1247-1254.
46. Brian D. Ziebart. 2010. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. (Doctoral dissertation, Carnegie Mellon University).