Inferring and Learning Subgoal Sequences

Kaiser Asif KASIF2@UIC.EDU
University of Illinois at Chicago, 851 S. Morgan St., Chicago, IL 60607 USA

Brian D. Ziebart BZIEBART@UIC.EDU
University of Illinois at Chicago, 851 S. Morgan St., Chicago, IL 60607 USA

Introduction

Computational models for understanding human behavior are an important component for enabling robots to appropriately interact with people in shared environments. Recent techniques assume that behavior is goal-directed and infer the goals and future behaviors of people (Baker et al., 2006) to plan complementary robotic control (Ziebart et al., 2009; Kuderer et al., 2012) or learn imitative control policies (Verma & Rao, 2006; Michini et al., 2013).

This goal-based approach to learning predictive models of behavior works well when the boundaries between behaviors with different goals can be readily discerned. For example, with pedestrian motion, a person may spend a noticeable amount of time at an intended goal location before proceeding towards a next intended goal. However, some sequences of observed behavior may not have such clear demarcations. For example, a pedestrian may walk to a garbage can to discard an object and, without stopping, then walk to an intended location. Given only position information, the change in goals that occurs at that point cannot be explicitly recognized. Learning a goal-directed model of behavior without having appropriately recognizing these subgoals will lead to poorly trained models and less accurate predictions.

Previous techniques identify subgoals (sequences of goals) based on the state visitation frequencies (McGovern & Barto, 2001) or graph properties of the (discrete) decision process (Menache et al., 2002; Şimşek et al., 2005). These techniques require many observed trajectories and/or are not applicable in continuous state spaces. We discuss the relationships of our approach with Bayesian change-point detection and trajectory segmentation in detail later in the paper.

In this work, we investigate the subgoal inference problem using Bayesian probabilistic methods to infer subgoals from trajectory likelihood functions. We present

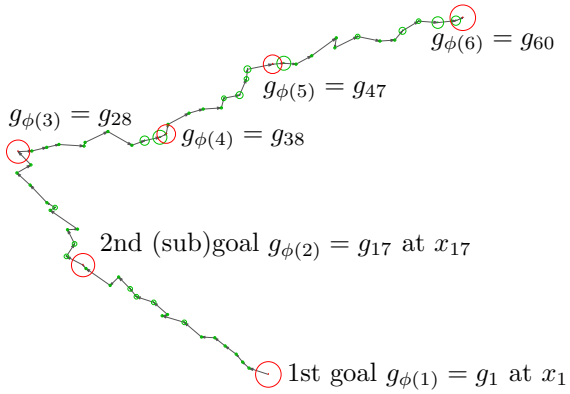


Figure 1. A trajectory of states (connected by directed edges) and posterior goal probabilities (circle areas in proportion to posterior probability) with actual goals in red.

an efficient quadratic time algorithm for inferring the posterior of a sequence of goals and the marginals of individual subgoals and describe its applicability for learning hierarchical models of behavior. We evaluate the accuracy of our inference approach in recovering latent subgoals from a synthetic dataset, and also apply it to a real dataset of pedestrian trajectories.

The subgoal inference task

We focus on the problem of identifying the subgoals that motivate observed sequences of behavior, as shown in Figure 1. Though we focus on location-based behavior in this paper, the technique is general for any sequences of behavior that can be represented as a decision process. We denote the complete trajectory of states as $x_{1:T}$ ($x_t \in \mathcal{X}$). We assume that some subset of these states are (sub)goal locations which the pedestrian intended to reach. The vector $g_{1:T}$ ($g_t \in \{0, 1\}$) indicates whether each state is a subgoal location or not and the position of active subgoals. We denote using $g_{\phi(k)}$ the kth active goal. More formally, the

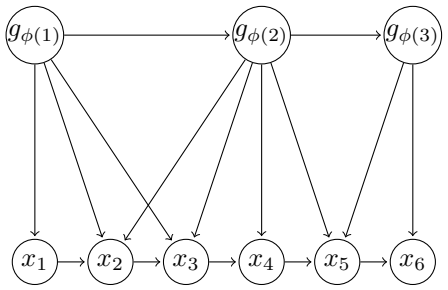


Figure 2. Bayesian network representing our likelihood function.

subgoal inference task is any of the following:

- To (probabilistically) determine which of the states are subgoals via the posterior,

$$P(g_{1:T}|x_{1:T}) = \frac{P(x_{1:T}|g_{1:T})P(g_{1:T})}{P(x_{1:T})}, \quad (1)$$

  which factors according to Bayes' theorem as shown;

- To obtain the marginal probability of a subgoal, $P(g_t = 1|x_{1:T})$, or subsequent subgoals, $P(g_{\phi(k)}, g_{\phi(k+1)}|x_{1:T})$; or

- To find the most probable subgoal sequence (maximum a posteriori estimate):

$$g^*_{1:T} = \operatorname*{argmax}_{g_{1:T}} P(g_{1:T}|x_{1:T}). \quad (2)$$

We assume that the likelihood function for a trajectory between two consecutive goals, denoted

$$P(x_{\phi(k):\phi(k+1)}|g_{\phi(k)}, g_{\phi(k+1)}), \quad (3)$$

is known. We also assume that a prior probability distribution over goals factors according to a known Markov chain distribution,

$$P(g_{1:T}) = P(g_{\phi(1)}) \prod_{k=1}^{K-1} P(g_{\phi(k+1)}|g_{\phi(k)}), \quad (4)$$

which is decomposed according to Figure 2. With slight abuse of notation, we let $P_\phi(g_j|g_i)$ represent the conditional probability that g_j is the next active goal following g_i from Eq. (4) and let $P_\phi(x_{i:j}|g_i, g_j)$ represent the trajectory probability between consecutive subgoals g_i and g_j from Eq. (3). The first and the last states are assumed to be goals: $g_1 = g_T = 1$, $\phi(1) = 1$, $\phi(K) = T$.

An efficient subgoal inference algorithm

Previous work on goal-based model learning for robotics has employed Markov chain Monte Carlo (MCMC) to perform approximate subgoal inference for robotics application (Michini et al., 2013). We present an efficient polynomial time algorithm for exact subgoal inference within our model setting in this section.

Theorem 1. *Given the likelihood functions of sub-trajectories between pairs of goals, the goal probability marginals can be efficiently computed in $\mathcal{O}(T^2)$ time using dynamic programming:*

$$P(g_t = 1|x_{1:T}) = \frac{\alpha(t)\beta(t)}{\alpha(T)}, \quad (5)$$

where the $\alpha(t)$ and $\beta(t)$ terms are recursively obtained as follows:

$$\alpha(i) = \begin{cases} 1 & i = 1 \\ \sum_{k=1}^{i-1} \alpha(k) P_\phi(g_i|g_k) P_\phi(x_{k:i}|g_k, g_i) & i > 1 \end{cases}$$
$$\beta(i) = \begin{cases} 1 & i = T \\ \sum_{k=i+1}^{T} \beta(k) P_\phi(g_k|g_i) P_\phi(x_{i:k}|g_i, g_k) & i < T \end{cases} \quad (6)$$

The method is general for any goal-conditioned likelihood function and goal probability distribution corresponding to the independence properties of Figure 2. Though, in general, $\mathcal{O}(T^3)$ may be required to generate these likelihood evaluations, predictive inverse optimal control (Ziebart et al., 2009) techniques may provide likelihood evaluations more efficiently (Vernaza & Bagnell, 2012).

Learning hierarchical models of behavior

Subgoal inference is an important sub-problem of learning models of goal-directed behavior when the goals are not explicitly labeled (or directly recognizable from other context). In this section, we integrate our subgoal inference routine into a learning framework for hierarchical models of behavior. We assume that trajectory datasets do not have annotated subgoal locations or, at best, partial annotation. Our approach uses the expectation-maximization (EM) algorithm to infer estimates of the latent subgoal probabilities and then updates the prior subgoal distribution and the goal-directed trajectory likelihood function.

Let, $Q(i, j)$ be the expectation of the i-th and j-th trajectory points being consecutive subgoals. We can use the following equations to estimate the probability

distribution of $P_\phi(x_{i:j}|g_i, g_j)$,

E-step:

$$Q(i,j) = \frac{\alpha(i)\beta(j)P_\phi(g_j|g_i)P_\phi(x_{i:j}|g_i,g_j)}{\alpha(T)} \quad (7)$$

M-step:

$$\Theta_x^* = \underset{\Theta_x}{\text{argmax}} \sum_{i,j} Q(i,j) \log P(x_{i:j}|g_i, g_j; \Theta_x). \quad (8)$$

For estimating the prior subgoal distribution, the M-step is:

$$\Theta_g^* = \underset{\Theta_g}{\text{argmax}} \sum_{i,j} Q(i,j) \log P(g_j|g_i; \Theta_g). \quad (9)$$

Convergence to a local optima is guaranteed by sequentially applying the E-step and M-step (Dempster et al., 1977).

Hierarchical models can be learned using the EM algorithm by treating the goal sequences learned in a single-layer model (as described previously) as position sequences for a higher-level goal sequence distribution. Thus, instead of learning a prior distribution over subgoals, $P(g_{1:T})$, the distribution over subgoals is conditioned on a higher-level sequence of hyper-subgoals $P(g_{1:T}|g'_{1:T})$ with a prior distribution $P(g'_{1:T})$ (or conditioning on an even higher-level set of subgoals, $g''_{1:T}$).

Experiments

We evaluate our subgoal inference approach in two settings: using a synthetic dataset for which the true subgoals are known; and a real pedestrian trajectory dataset with unknown subgoals.

Synthetic dataset

For our synthetic dataset, we created goals and trajectories in two-dimensional Euclidean space using a simple model. First, we generated six subgoal points according to Gaussian distributions (centered at the previous subgoal point with variance of 10):

$$x_{\phi(k+1)}|x_{\phi(k)} \sim \text{Normal}\left(x_{\phi(k)}, \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix}\right) \quad (10)$$

Next, we generate trajectories between each pair of consecutive subgoals. The number of trajectory points between two consecutive subgoals, n_k, is based on the Euclidean distance between the subgoals,

$$n_k = \lceil 3\ ||x_{\phi(k+1)} - x_{\phi(k)}|| \rceil \quad (11)$$

with approximately three trajectory points for every unit of distance between subgoal points. The trajectory points themselves are each Gaussian distributed around the line connecting consecutive subgoals:

$$x_{\phi(k)+i} \sim \text{Normal}\left(x_{\phi(k)} + \frac{x_{\phi(k+1)} - x_{\phi(k)}}{n_k} i, \begin{bmatrix} \sigma_x & 0 \\ 0 & \sigma_x \end{bmatrix}\right). \quad (12)$$

We vary the covariance value σ_x in our experiments.

Pedestrian trajectory dataset

We employ a previously collected and studied dataset of pedestrian trajectories of tracked movements through a laboratory environment (Ratliff et al., 2009; Ziebart et al., 2009). In those previous investigations of this dataset, anomalous trajectories were discarded and only a subset of 166 trajectories were used to train and evaluate the methods. We focus our attention on rationalizing those anomalous trajectories as being motivated by sequences of subgoals.

Prior probabilities and likelihoods

For our synthetic dataset, we compute the trajectory likelihood using the generative distribution described in Eq.(12). As the relative positions of consecutive goals is distributed according to a two-dimensional, zero-mean Gaussian, the distribution over distances between goals is a (transformed) Chi-distribution of degree two. We construct our prior distribution over consecutive goals by evaluating this Chi-distribution at the spacing points of the trajectory points and normalizing.

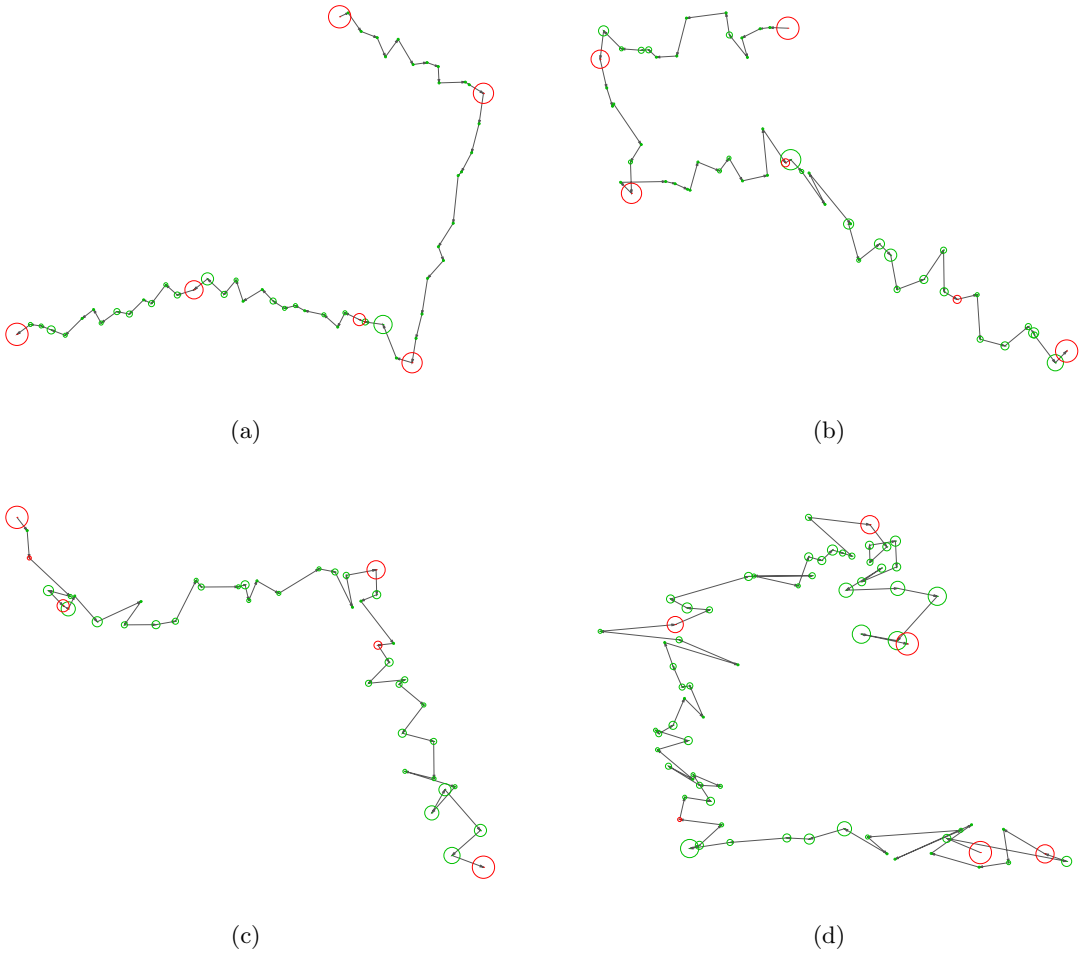
For our real dataset, we employ the prior work's smoothed prior goal distribution for our subgoal prior (Ziebart et al., 2009) and maximum entropy inverse reinforcement (Ziebart et al., 2008; 2009) learning distribution over trajectories between subsequent subgoals:

$$P(x_{\phi(k):\phi(k+1)}|g_{\phi(k)}, g_{\phi(k+1)}) = \frac{e^{-\sum_{i=\phi(k)}^{\phi(k+1)} \text{cost}(x_i)}}{\sum_{x' \in \Xi_{g_{\phi(k)}, g_{\phi(k+1)}}} e^{-\sum_{i=\phi(k)}^{\phi(k+1)} \text{cost}(x'_i)}}, \quad (13)$$

where the set of all paths from $g_{\phi(k)}$ to $g_{\phi(k+1)}$ is represented as $\Xi_{g_{\phi(k)}, g_{\phi(k+1)}}$. This model corresponds to an undirected conditional probabilistic graphical model as shown in Figure 3.

Synthetic dataset results

We generated 100 samples for each of the variance values (deviation from the connecting line) of {0.01, 0.05, 0.1, 0.2}. For precision and recall evaluations, we assume a point to be a goal if its posterior probability is greater than 0.5. The result (Figure 4) shows that



(a) (b) (c) (d)

Figure 5. Synthetic trajectories with inferred and actual subgoals. The area of the circles are proportional to the posterior probabilities. Red circles are the actual goals, green circles are all other points in the trajectory. (a) is with trajectory generating variance 0.01, (b) is with variance 0.05, (c) with 0.1 and (d) is with 0.2 variance

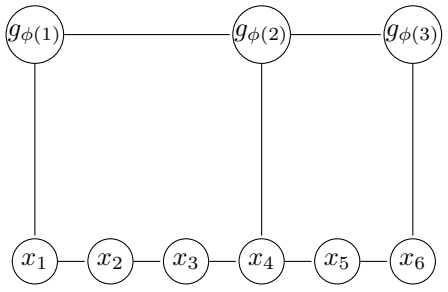


Figure 3. Random field probabilistic graphical model used for the likelihood function of the pedestrian trajectory dataset.

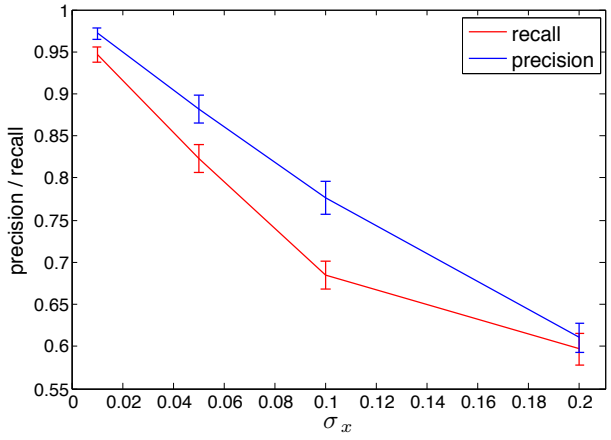


Figure 4. Correctness measurement with respect to different distribution variance of trajectory points.

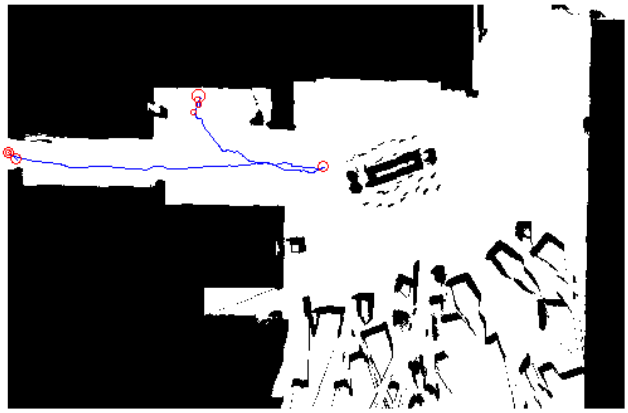


Figure 6. A pedestrian trajectory from the entrance to a table/garbage can location to the left portion of the laboratory. High-probability inferred subgoals (greater than 10% posterior probability) are displayed as red circles with areas in proportion to posterior probability.

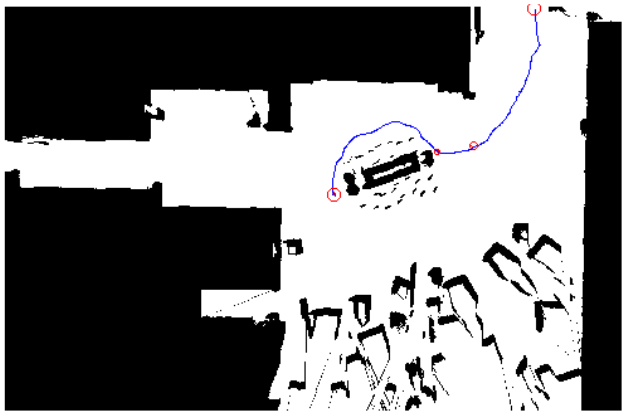


Figure 7. A pedestrian trajectory from the upper-right portion of the laboratory and around the table area. High-probability inferred subgoals (greater than 10% posterior probability) are displayed as red circles with areas in proportion to posterior probability.

the prediction loses its accuracy when the trajectory points are more deviated from the straight line.

Due to the noise in the data, the inference routine is sometimes misled (Figure 5(d)), but the likelihood function also prevents it from choosing all extreme points as subgoals, which would lead to a larger number of errors.

Pedestrian trajectory dataset results

The results of our subgoal inference procedure on two pedestrian trajectories are shown in Figure 6 and Figure 7. In the first figure, a garbage can is located within arm's reach of the right-most inferred subgoal location and may explain the trajectory. Additional subgoals inferred near the beginning and end of the trajectory are believed to be due to tracking software errors. An inverse reinforcement learning model that views this trajectory as being directed towards a single goal (and not at least two subgoals) will have great difficulty rationalizing the trajectory and estimating an appropriate cost function. In the second figure, the pedestrian chooses a seemingly sub-optimal trajectory around the table location (in comparison to many other trajectories in the dataset). The subgoal inference method places moderate posterior probability of the trajectory actually being directed to intermediate subgoals along the way.

Related Work

Subgoal recognition has recently been investigated for robotics applications. An imitation learning technique for unmanned aerial vehicles (UAVs) learns a sequence of significant subgoals that adequately define demonstrated behavior (Michini et al., 2013) within a Bayesian non-parametric framework. A sampling algorithm is employed to approximately calculate the marginal posterior probability of each particular subgoal. The resulting learned subgoals are then employed within a UAV controller to autonomously imitate previously demonstrated behavior. In contrast, our algorithms provide exact inference in quadratic time.

Bayesian change-point detection (Fearnhead, 2006) is a related technique for inferring that different subsequences of time series data are generated from different probability distributions. At timestep t, either the datapoint x_t is generated from the same distribution that generated data $x_\tau, \ldots, x_{t-1}$, (for each $\tau \in \{1, \ldots, t-1\}$) or it is generated from a new parametric distribution with unknown parameters assumed to be drawn from a known prior distribution. Using Bayes' theorem, a posterior distribution over possible durations of the current subsequence of data and model parameters from each subsequence is maintained in $\mathcal{O}(t)$ time per timestep or $\mathcal{O}(T^2)$ time for the entire sequence. This change-point detection approach has been employed for skill tree learning from demonstration (Konidaris et al., 2012).

Our approach, in contrast, assumes a known likelihood function (and provided evaluations of that likelihood function between pairs of potential subgoals) that is conditioned on the next subgoal. Thus, at time-step t, for each $\tau < t$, there are $T - t + 1$ possible para-

metric models to consider (one for each next subgoal position, $t+1, t+2, \ldots, T$) and update if following the Bayesian change-point perspective. Thus, a straightforward application of Bayesian change-point detection would entail $\mathcal{O}(T^3)$ total time.

Dynamic programming algorithms for trajectory segmentation (Mann et al., 2002) are more similar to our approach. They operate by assuming that each sub-trajectory segment is generated from e.g., a polynomial curve with Gaussian error. Given the best fit polynomial for each sub-trajectory segment ($\mathcal{O}(T^2)$ total sub-trajectories), the optimal trajectory segmentation is obtained in $\mathcal{O}(T^2)$ time complexity. Rather than obtaining an independent model for each sub-trajectory segment, our approach attempts to learn sub-trajectory models with shared parameters. Thus, we provide a posterior distribution given the sub-trajectory segment likelihoods for incrementally improving the likelihood function using the expectation-maximization algorithm.

Conclusion

In this paper, we presented a method for inferring and learning when faced with sequences of behavior that are motivated by latent subgoals. Our approach uses a quadratic-time dynamic programming algorithm for inference and the expectation-maximizaton algorithm for learning. We evaluated our approach on a synthetic dataset to show the relationship between noise in the trajectory samples and resulting precision/recall of subgoals. We also applied the approach to real pedestrian trajectories and showed examples where we believe it correctly identifies subgoals that would help improve learned predictie inverse optimal models.

References

Baker, C., Tenenbaum, J., and Saxe, R. Bayesian models of human action understanding. *Advances in Neural Information Processing Systems*, 18:99, 2006.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

Fearnhead, Paul. Exact and efficient bayesian inference for multiple changepoint problems. *Statistics and computing*, 16(2):203–213, 2006.

Konidaris, George, Kuindersma, Scott, Grupen, Roderic, and Barto, Andrew. Robot learning from demonstration by constructing skill trees. *The International Journal of Robotics Research*, 31(3):360–375, 2012.

Kuderer, Markus, Kretzschmar, Henrik, Sprunk, Christoph, and Burgard, Wolfram. Feature-based prediction of trajectories for socially compliant navigation. In *Proc. of Robotics: Science and Systems (RSS)*, 2012.

Mann, Richard, Jepson, Allan D, and El-Maraghi, Thomas. Trajectory segmentation using dynamic programming. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 1, pp. 331–334. IEEE, 2002.

McGovern, Amy and Barto, Andrew G. Automatic discovery of subgoals in reinforcement learning using diverse density. In *Proceedings of the International Conference on Machine Learning*, pp. 361–368, 2001.

Menache, Ishai, Mannor, Shie, and Shimkin, Nahum. Q-cut-dynamic discovery of sub-goals in reinforcement learning. In *Machine Learning: ECML 2002*, pp. 295–306. Springer, 2002.

Michini, Bernard, Cutler, Mark, and How, Jonathan P. Scalable reward learning from demonstration. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

Ratliff, Nathan, Ziebart, Brian, Peterson, Kevin, Bagnell, J. Andrew, Hebert, Martial, Dey, Anind K., and Srinivasa, Siddhartha. Inverse optimal heuristic control for imitation learning. In *Proc. AISTATS*, pp. 424–431, 2009.

Şimşek, Özgür, Wolfe, Alicia P, and Barto, Andrew G. Identifying useful subgoals in reinforcement learning by local graph partitioning. In *Proceedings of the International Conference on Machine learning*, pp. 816–823. ACM, 2005.

Verma, Deepak and Rao, Rajesh. Goal-based imitation as probabilistic inference over graphical models. *Advances in neural information processing systems*, 18:1393–1400, 2006.

Vernaza, Paul and Bagnell, Drew. Efficient high dimensional maximum entropy modeling via symmetric partition functions. In *Advances in Neural Information Processing Systems 25*, pp. 584–592, 2012.

Ziebart, Brian D., Maas, Andrew, Bagnell, J. Andrew, and Dey, Anind K. Maximum entropy inverse reinforcement learning. In *Proc. AAAI Conference on Artificial Intelligence*, pp. 1433–1438, 2008.

Ziebart, Brian D., Ratliff, Nathan, Gallagher, Garratt, Mertz, Christoph, Peterson, Kevin, Bagnell, J. Andrew, Hebert, Martial, Dey, Anind K., and Srinivasa, Siddhartha. Planning-based prediction for pedestrians. In *Proc. Intelligent Robots and Systems*, pp. 3931–3936, 2009.

Appendix

Proof of Theorem 1. We begin by computing the numerator of Eq. (1), which is the joint probability of goal g_t being active and the trajectory.

$$
\begin{aligned}
P(g_t = true, x_{1:T}) &= \sum_{g_{-t}} P(g_{1:T}, x_{1:T}) \\
&= \sum_{g_{1:t-1}} \sum_{g_{t+1:T}} P(g_{1:t-1}) P(g_t|g_{1:t-1}) P(g_{t+1:T}|g_t) P(x_{1:t}|g_{1:t}) P(x_{t:T}|g_{t+1:T}) \\
&= \underbrace{\Big(\sum_{g_{1:t-1}} P(g_{1:t-1}) P(x_{1:t}|g_{1:t}) P(g_t|g_{1:t-1}) \Big)}_{\alpha(t)} \underbrace{\Big(\sum_{g_{t+1:T}} P(g_{t+1:T}|g_t) P(x_{t:T}|g_{t+1:T}) \Big)}_{\beta(t)} \\
&= \alpha(t)\beta(t)
\end{aligned} \tag{14}
$$

Where,

$$
\alpha(t) \triangleq \sum_{g_{1:t-1}} P(g_{1:t-1}) P(g_t|g_{1:t-1}) P(x_{1:t}|g_{1:t}) \tag{15}
$$

Expanding $P(x_{1:t}|g_{1:t})$ by the last two consecutive goals and using goal sequence notations we can write,

$$
\begin{aligned}
&= \sum_{\phi(k-1) \in \{1,\ldots,t-1\}} \sum_{g_{1:\phi(k-1)-1}} P(g_{1:\phi(k-1)}) P_\phi(g_t|g_{\phi(k-1)}) P(x_{1:\phi(k-1)}|g_{1:\phi(k-1)}) P_\phi(x_{\phi(k-1):t}|g_{\phi(k-1)}, g_t) \\
&\quad \text{where, } g_t \text{ is the } k\text{th goal and } g_{\phi(k-1)} = k-1\text{th goal,} \\
&\quad g_{1:\phi(k-1)} = \text{the goal sequence before } g_t \ (k\text{th goal}), \\
&\quad x_{1:\phi(k-1)} = \text{the trajectory from starting point to } k-1\text{th goal,} \\
&\quad P_\phi(g_j|g_i) = \text{the conditional probability that } g_j \text{ is the next active goal following } g_i \text{ and} \\
&\quad P_\phi(x_{i:j}|g_i, g_j) = \text{trajectory probability between consecutive goals } g_i \text{ and } g_j \\
&= \sum_{\phi(k-1) \in \{1,\ldots,t-1\}} P_\phi(g_t|g_{\phi(k-1)}) P_\phi(x_{\phi(k-1):t}|g_{\phi(k-1)}, g_t) \sum_{g_{1:\phi(k-1)-1}} P(g_{1:\phi(k-1)}) P(x_{1:\phi(k-1)}|g_{1:\phi(k-1)}) \\
&= \sum_{\phi(k-1) \in \{1,\ldots,t-1\}} P(g_t|g_{\phi(k-1)}) P_\phi(x_{\phi(k-1):t}|g_{\phi(k-1)}, g_t) \ \alpha(\phi(k-1)) \text{ , by definition of } \alpha \text{ from Eq.(15)} \\
&= \sum_{i=1}^{t-1} \alpha(i) P_\phi(g_t|g_i) P_\phi(x_{i:t}|g_i, g_t) \text{ , replacing } \phi(k-1) \text{ by } i
\end{aligned} \tag{16}
$$

And similarly,

$$
\beta(t) \triangleq \sum_{g_{t+1:T}} P(g_{t+1:T}|g_t) P(x_{t:T}|g_{t+1:T}) \tag{17}
$$

Expanding by the first pair of goals in the sequence, we can write

$$
\begin{aligned}
&= \sum_{\phi(k+1) \in \{t+1,\ldots,T\}} \sum_{g_{\phi(k+1)+1:T}} P_\phi(g_{\phi(k+1)}|g_t) P(g_{\phi(k+1)+1:T}|g_{\phi(k+1)}) P_\phi(x_{t:\phi(k+1)}|g_t, g_{\phi(k+1)}) P(x_{\phi(k+1):T}|g_{\phi(k+1)+1:T}) \\
&= \sum_{\phi(k+1) \in \{t+1,\ldots,T\}} P_\phi(g_{\phi(k+1)}|g_t) P_\phi(x_{t:\phi(k+1)}|g_t, g_{\phi(k+1)}) \sum_{g_{\phi(k+1)+1:T}} P(g_{\phi(k+1)+1:T}|g_{\phi(k+1)}) P(x_{\phi(k+1):T}|g_{\phi(k+1)+1:T}) \\
&= \sum_{\phi(k+1) \in \{t+1,\ldots,T\}} P_\phi(g_{\phi(k+1)}|g_t) P_\phi(x_{t:\phi(k+1)}|g_t, g_{\phi(k+1)}) \ \beta(\phi(k+1)) \text{ , by definition of } \beta \text{ from Eq.(17)} \\
&= \sum_{i=t+1}^{T} \beta(i) P_\phi(g_i|g_t) P_\phi(x_{t:i}|g_t, g_i) \text{ , replacing } \phi(k+1) \text{ by } i
\end{aligned} \tag{18}
$$

Also, since the two end points are always goals, $\alpha(1) = \beta(T) = 1$. This property and Eq. (16) and Eq. (18) together defines the Eq. (6).

By definition of $\alpha(t)$ and $\beta(t)$, $\alpha(T)$ and $\beta(1)$ are the probability functions that are summed over all possible goals in the trajectory. Therefore,

$$\alpha(T) = \beta(1) = \sum_{g} P(g_{1:T}, x_{1:T}) \tag{19}$$

So we have

$$P(g_t = 1|x_{1:T}) = \frac{\sum_{g_{-t}} P(g_{1:T}, x_{1:T})}{\sum_{g} P(g_{1:T}, x_{1:T})} = \frac{\alpha(t)\beta(t)}{\alpha(T)} \tag{20}$$

To compute all $\alpha(t)$ and $\beta(t)$, it requires $\mathcal{O}(T)$ time for each one (using Eq. (16) and Eq. (18)) and there are T number of each of the terms, thus the time complexity is $\mathcal{O}(T^2)$. □