Plan, Activity, and Intent Recognition (PAIR) Tutorial

AAAI 2026

Reuth Mirsky, Sarah Keren, Christopher Geib

Speakers

Reuth Mirsky
Tufts University

Sarah Keren
Technion

Christopher Geib
Charles River Analytics

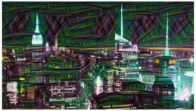
Table of contents

Past Events

PAIR 2017 – San Francisco

PAIR 2018 – New Orleans

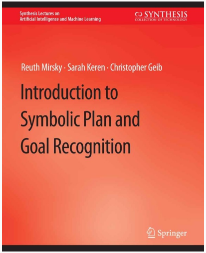
PAIR 2019 – Honolulu



PAIR 2020 – New York City

PAIR 2021 + tutorial

PAIR 2026 – Singapore



Intro to PAIR 2021

Historical Overview

We were there (almost) at the very beginning.

- 1956 - Dartmouth conference occurs (coins the Term AI)
- 1959 - Newell, Shaw, Simon build the General problem-solver
- 1960 - Plans and the Structure of Behavior. Miller, Galanter, and Pribram "In this survey we were especially fortunate in having at our disposal a large mass of material, much of it still unpublished, that Miller had obtained from Allen Newell, J. C. Shaw, and Herbert A. Simon..."
- 1971 - STRIPS, Fikes and Nilsson.
- 1978 - The plan recognition problem: An intersection of psychology and artificial intelligence. Schmidt, Sridharan, Goodson
- ... and don't get me started on pattern recognition and HMMs ('30s and '40s)

Schmidt et. al's Def.

- "The problem of plan recognition is to take as input a sequence of actions performed by an actor and to infer the goal pursued by the actor and also to organize the action sequence in terms of a plan structure. This plan structure explicitly describes the goal-subgoal relations among its component actions."

But it was harder than we thought.

- Like many of the other (now) sub-fields of AI, we realized there was more than one problem.
- At least three different problems that were at one time or another called plan recognition.
  - Activity Recognition
  - Goal Recognition
  - Plan Recogntion

Definition:

- INPUT: A sequence of noisy sensor inputs over time.
- OUTPUT: A unique label for each temporal subsequence.

Central problem:

Dealing with noise in the input observation stream.

Alternative characterization:

Classification/Labeling of noisy temporal observations.

Sometimes called "Behavior recognition"

Example:

- Video segmentation of football plays. (e.g. passing, clearing, throw-in, etc...)

Goal Recognition

Definition:

- INPUT: An ordered sequence of discrete symbolic input tokens.
- OUTPUT: A unique label (perhaps with a probability) for each temporal subsequence.

Central problem:

Dealing with evidence for multiple conflicting hypothesis.

Alternative characterization:

Classification/Labeling temporal observations where each observation can contribute to many possible labels.

Example:

- Identifying computer user goals from observing their actions. (e.g. searching web, starting a new document, confused, etc...)

Plan Recognition

Definition:

- INPUT: An ordered sequence of discrete symbolic input tokens.
- OUTPUT: Complex structure capturing plan being executed. Potentially including abstract tasks that have been done and which are yet to do and traditionally the goal of the plan.

Central problem:

Combining sequences of lower level observations into larger structured patterns. (probabilistic or not)

Alternative characterization:

Temporal pattern matching, sequence matching.

Example:

- Identify the plan and goal of cyber intruders and their progress though a network.

Chris' Digression: "Intent Recognition"

Don't use this term. Nothing good comes from it.

If you read it, ask yourself what they are actually doing.

By this people DON'T mean a plan, action, goal, or state.

They might mean:

- Civilians (non-AI researchers): Ineffable magic that differentiates human's and synthetic agent's actions.
- Some philosophers: A separate pro-attitude towards a plan denoting a commitment to its execution.
- Other philosophers: A mental state in which an agent believes a sequence of actions will bring about a state they desire and believes that they will execute those actions to that end.
- Military: What the Sr. Officer wanted to have happen.

This is NOT everything you should know about these papers and past work!

Kautz and Allen[1]: PLAN REC

- Domain: Cooking.
- Approach: Minimal graph covering based on a preexisting plan library.
- Core contribution: Plan libraries and formalization.
- Limitation: Bias for minimal number of goal and not probabilistic

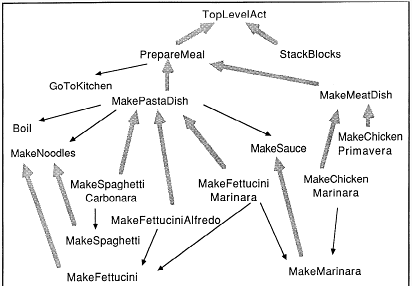


Figure 1: Action Hierarchy

[1] Henry Kautz and James F. Allen. Generalized plan recognition. In *Proceedings of the National Conference on Artificial Intelligence*, pp. 32-38, 1986.

Carberry[2] and Littman [3] : PLAN REC? GOAL REC?

- Domain: Dialog, Discourse, Natural Language Understanding.
- Approach: Logical inference using the situation calculus.
- Core contribution: Realization of how much inference we actually do in language.
- Limitation: Cost of encoding and inference

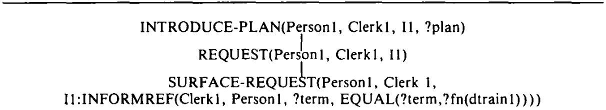


Figure 8. Chaining produces an intermediate plan recognition structure.

[2] S. Carberry. *Plan Recognition in Natural Language Dialogue.* ACL-MIT Press Series in Natural Language Processing. MIT Press, 1990.

[3] D. Littman and J. Allen, A plan recognition model for subdialogues in conversation, *Cognitive Science*, vol. 11(2), pp. 163-200, 1987.

Villain[4]: PLAN REC

- Domain: None really
- Approach: Parsing formal plan grammars.
- Core contribution: Plan recognition as parsing.
- Limitation: No actual system provided.
- Limitation: An acyclic hierarchy does not contain any recursive plan definitions, and has a finite yield.

[4] M. Vilain. Getting serious about parsing plans: A grammatical analysis of plan recognition. In *Proceedings of National Conference on Artificial Intelligence*, pp190-197, 1990.

Goldman and Charniak[5]: PLAN REC

- Domain: Story understanding.
- Approach: Dynamically assembly of Bayes nets.
- Core contribution: The problem of universal quantification and unbound vars in stories.
- Limitation: Limited by Bayesian methods of the time, and building Bayes nets dynamically.

[5] R. Goldman and E. Charniak. Probabilistic text understanding. In *Statistics and Computing*, Vol 2, pp. 105-114, 1992.

Pynadath and Wellman[6]: PLAN REC

- Domain: Driving/Lane Selection.
- Approach: Probabilistic State Dependant Grammar to specialized probabilistic inference
- Core contribution: Formalizing the problem in a grammar looked at a dynamic Bayes net.
- Limitation: Then built their own probabilistic algorithm.

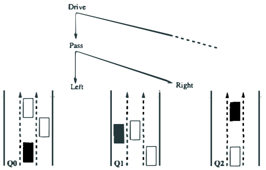


Figure 2: Simple PSDG parse tree from traffic domain.

[6] D. Pynadath and M. Wellman, Accounting for context in plan recognition with application to traffic monitoring, *Proceedings of UAI-95*, pp. 472-481, 1995.

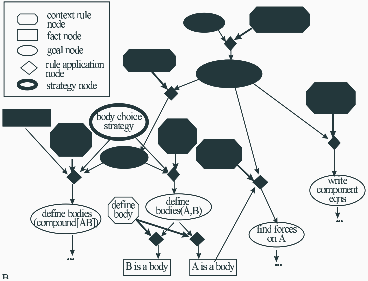
Horvits et al[7]: GOAL REC

- Domain: Software assistive systems
- Approach: Bayes nets.
- Core contribution: Goal recognition in a real system.
- Limitation: Limited by Bayesian methods of the time. And Bayes nets of our time. :-)

[7] E. Horvitz, J. Breese, D. Heckerman, D. Hovel, and K. Rommelse, The Lumiere project: Bayesian user modeling for inferring the goals and needs of software users, *Proceedings of UAI-98*, pp. 256-265, 1998.

Conati et al[8]: GOAL REC

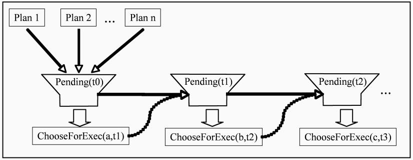
- Domain: Educational agents.
- Approach: Bayes nets.
- Core contribution: Explicit modeling of incorrect plans.
- Limitation: Bayesian models: propositional and scale.



[8] C. Conati, A. Gertner, K. VanLehn, and M. Druzdzel, On-Line student modeling for coached problem solving using Bayesian networks, *Proceedings of User Modeling-97*, pp. 231-242, 1997.

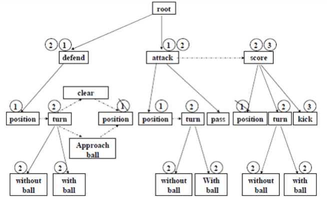
Geib and Goldman[9]: PLAN REC

- Domain: Synthetic domains.
- Approach: Parsing of probabilistic plan recognition as parsing.
- Core contribution: Efficient grammars for parsing, multiple concurrent goals, pending sets.
- Limitation: Required building the complete set of parses.



[9] C. Geib and R. Goldman. A probabilistic plan recognition algorithm based on plan tree grammars, *Artificial Intelligence*, vol 173(11), pp. 1101-1132, 2009.

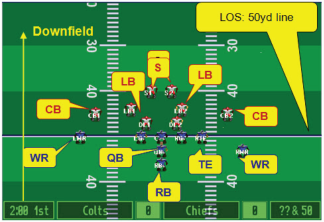
- Domain: RoboCup
- Approach: Marker passing over packed parse trees.
- Core contribution: More efficient plan recognition as parsing.
- Limitation: Multiple instances of the same plan.



[10] D. Avrahami-Zilberbrand and G. Kaminka, Fast and complete symbolic plan recognition, *Proceedings of IJCAI-05*, pp. 653-658, 2005.

Laviers et al[11]: ACTIVITY REC

- Domain: Video games (RUSH 2008 football).
- Approach: Support vector machines for classification
- Core contribution: Real-world deployment fast enough to make a difference.
- Limitation: New plays?



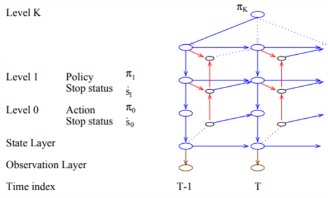
[11] K. Laviers, G. Sukthankar, D. Aha, M. Molineaux, and C. Darken, Improving Offensive Performance Through Opponent Modeling, *Proceedings of AIIDE*, pp.58-63, 2009.

Ramirez and Geffner[12]: GOAL REC

- Domain: IPC domains.
- Approach: Plan recognition as planning.
- Core contribution: Use of planning algorithms.
- Limitation: Initial work was actually doing goal recognition.

[12] M. Ramírez, and H. Geffner, Plan recognition as planning. in *Proceedings of IJCAI*, pp. 1778-1783, 2009

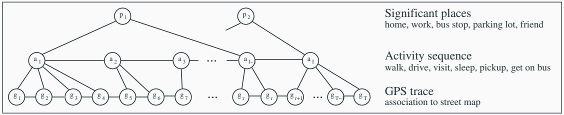
- Domain: 2D navigation.
- Approach: Hierarchical Hidden Markov Models.
- Core contribution: Using HMMs at multiple levels to actually do plan recognition
- Limitation: Fully ground models.



[13] H. Bui, S. Venkatesh, and G. West. Policy recognition in the Abstract Hidden Markov Model, *Journal of Artificial Intelligence Research*, vol 17, pp. 451-499, 2002.

Liao, Fox, and Kautz [14]: ACTIVITY REC

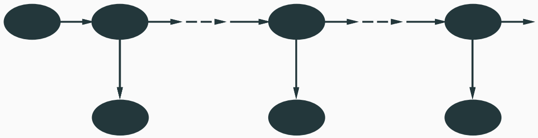
- Domain: Daily activity tracking. (2D tracking)
- Approach: Hierarchical conditional random fields.
- Core contribution: Using HCRF, and real GPS data
- Limitation: location based...



[14] L. Liao, D. Fox, and H. Kautz. Hierarchical conditional random fields for GPS-based activity recognition, *Proceedings of the 12th International Symposium of Robotics Research (ISRR)*, 2005.

A Small Nod to the Markov Decision Process People

- In a sense they were here first (like the '50s).
- If you have a Hidden Markov Model:
  - *Filtering*: $P(X_t|e_{1:t})$ predicting the current hidden state.
  - *Prediction*: $P(X_{t+k}|e_{1:t}k > 0$ predicting the next hidden state.
- What if the hidden state captured the possible plan states?



Formulating a Recognition Problem

Formulating a recognition problem

In this part of the tutorial we will focus on the following question:

What are the elements that need to be specified to define a recognition problem ?

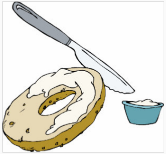
As a running example we will use the human-robot collaboration setting by Levine and Williams (ICAPS'14 and JAIR'18)

Alice is making breakfast for herself with the help of her trusty robot. The team is either making coffee (for which Alice uses a mug, and for which the coffee beans need to be grounded) or getting some juice (for which Alice uses a glass, and oranges need to be pressed). To eat, the team is either making a bagel with cream cheese or getting some cereal and milk.










Elements of a Recognition Problem

- Environment
- Acting agent (actor)
- Recognition system (recognizer)
- Actor-recognizer relationship

Our focus is on recognition of a single agent. Recognition in a multi-agent setting is very interesting but beyond the scope for today.

Environment E

A description of the dynamics of the setting in which the actor operates, including all aspects that dictate the possible agent behaviours.

Also referred to as the **domain theory**.

Can be described as a tuple $\langle S, I, A, T, \mathcal{G} \rangle$

- The state space S
  - Often, a set of features F is used to describe a state
  - In continuous domains these features are numeric-valued
- The set of possible initial states I
- The set of actions A(s) that can be performed at each state $s \in S$
  - Deterministic/ stochastic actions
  - Temporal actions
- The transition function T
  - Deterministic: $T : S \times A \times S \to S$
  - Stochastic: $T : S \times A \times S \to [0, 1]$
- The set of possible goals $\mathcal{G}$

We may also have constraints (e.g. temporal constraints) that need to be respected

Environment

In continuous domains, actions transition from one state to another via paths through the state space, rather than through discrete states.

In non-deterministic domains, instead of a plan, the actor follows a **policy** - which is a mapping from states to actions.

An environment induces a set of Π of paths / policies that represent the set of possible behaviors in the environment.

Environment

In our running example:

- A state specifies the position of the objects (e.g. whether the cup is on the table), the execution status of the different sub-tasks, etc.
- The actions represent the activities that can be performed by the human or the robot (e.g., pour coffee).
- In our description, actions are deterministic. In a probabilistic version of this problem, an action may fail with some probability.

The Actor

Formalizing the actor requires enumerating the assumptions made by an observer regarding how an agent with a specific objective will choose to act in an environment.

The actor's actions may be influenced by many factors:

- its familiarity with the environment (possibly reflected by its sensor model),
- its capabilities and preferences (e.g., can the actor compute an optimal plan?),
- its relationship to the observer,
- and more.

The Actor

Generally, there are three types of relationships between the actor and the observer discussed in the literature.

1. **Keyhole recognition:** [Kautz and Allen, 1986] - the actor is unaware or indifferent to the recognition of their plans and goals.
2. **Adversarial recognition:** the actor is hostile to the inference of its plans and goals.
3. **Intended recognition:** the actor explicitly acts with the intent that its plans and goals are easy to infer

The Actor

When defining the actor, we need to specify the assumptions made about how an agent with a specific goal chooses to behave in a given environment

- Note that the actor's behavior may be influenced by its relationship to the recognizer, which we will discuss later on. For now, we are only characterizing the actor's capabilities.

Recognizer (Recognition System)

The formulation of the actor specifies how the recognizer expects the actor to behave w.r.t each goal/ action / activity.

Also important to specify:

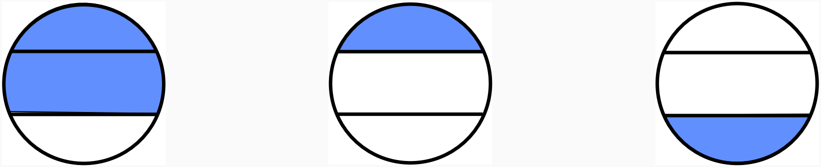
- Observability- How does the recognizer perceive the actor's behavior? What is the recognizer's sensor model
- Objective- What is the recognizer's objective ?
- Possible Intervention- Can the recognizer interact with the actor or affect its behavior?

Recognizer's Observability

- Most of the work we will cover today maps raw sequences of data from the recognizer's sensors to a sequence of symbols: $O_{rec} : E \rightarrow \vec{o}$
- Typically defined as a mapping from actions / states to observation tokens. $O_{rec} : A \rightarrow O$ or $O_{rec} : S \rightarrow O$
- The observation sequence is the entity that is analyzed.

Three types of Recognition

- Plan recognition
- Goal recognition
- Activity recognition



Recognizer's Objective

The recognition task

- Typically, the recognizer wants to recognize the actor's goal / plan / activity as soon as possible.
- The recognition task can be generally characterized via:
  - $P(G|\vec{o})$ for goal recognition, where $G \in \mathcal{G}$ is the goal and $\vec{o}$ is the perceived observation sequence.
  - $P(\pi|\vec{o})$ for plan recognition, where π is a complete plan.
  - $P(a|\vec{o})$ for activity recognition, where a is an activity.
- As a special case, the mappings above can be deterministic.
- In some of the works, the objective is to find the most probable goal / plan / activity. Others assume multiple goals could be pursued at the same time.

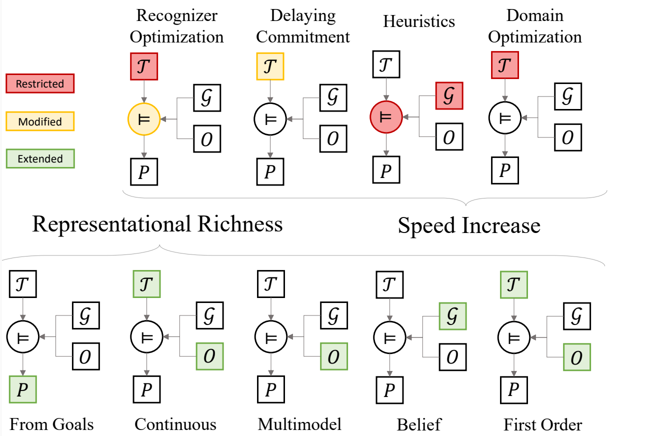
Checklist

- ✓ What are the dynamics of the environment ?
- ✓ What are the assumptions made on agent behavior ?
- ✓ What is the language used to represent agent behavior ?
- ✓ What is the relationship to the recognizer ?
- ✓ How is the agent behavior perceived ?
- ✓ What are the possible interventions ?
- ✓ What is the recognition objective ?

How to Make a New Contribution

Very few papers present a whole new representation for recognition tasks.

Many contributions are founded on an existing representation and improve it, either by *enhancing representational richness* or *improving processing speed*.



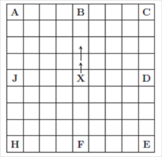
Solution Approaches

Solution Types

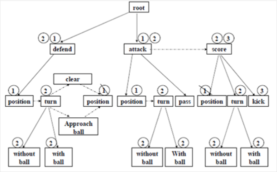
Plans ($\langle s_0, a_0, s_1, a_1 \ldots s_{n-1}, a_{n-1}, s_n \rangle$) transition from state to state can be mapped using a linear sequence of actions.
Prominent approaches: deterministic planning, CSP.

Hierarchies ($T_i \rightarrow T_{i,0}, \ldots, T_{i,k}$) focuses on environments where plans are constructed hierarchically.
Prominent approaches: Hierarchical planning, parsing.

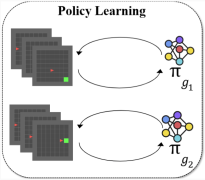
Policies ($\Pi : A \rightarrow N$) focuses on stochastic environments where execution might not be deterministic.
Prominent approaches: Stochastic planning, RL, supervised learning.



Ramírez and Geffner 2010

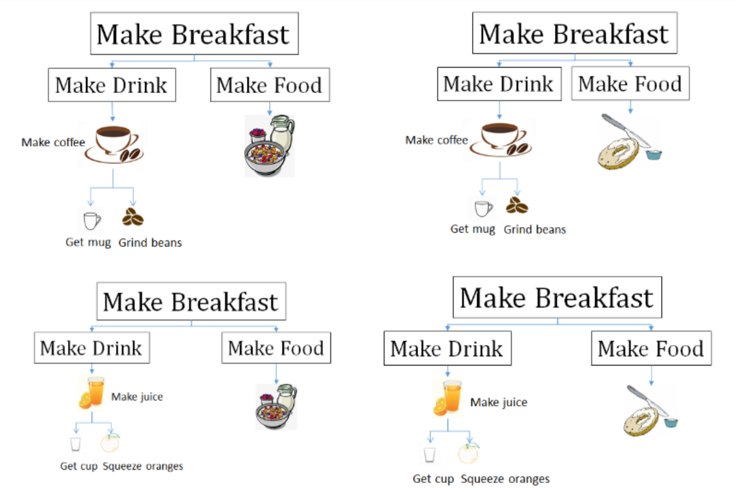


Avrahami-Zilberbrand and Kaminka 2009



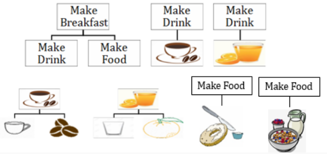
Amado et al. 2022

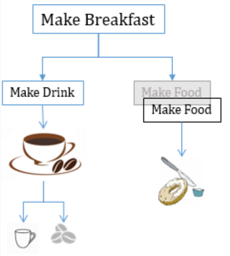
Explicit Symbolic Approaches (Representative Example: Parsers)

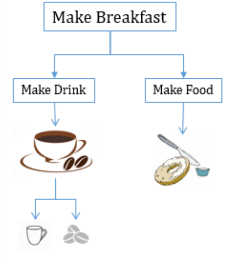


PHATT : Probabilistic Hostile Agent Task Tracker [Geib and Goldman, 2009]

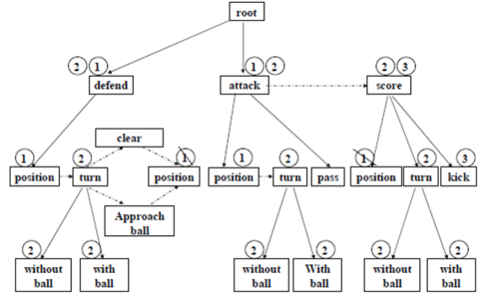
- Input 1: plan libraries as a set of recipes with partial ordering (Make Breakfast $\rightarrow$ Make Drink, Make Food | ϕ)
- Input 2: observation sequence ($\langle$  ,  $\rangle$)
- Output: probabilistically weighted set of explanations
- In Figures: Set of recipes; Combining leftmost trees; Explanation



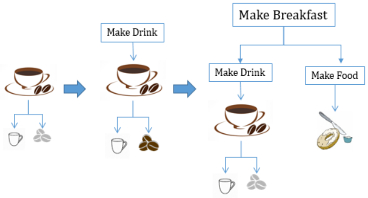




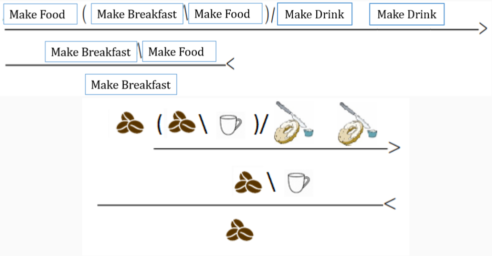
- Uses a single structure to represent the plan library
- Each observation adds marks on nodes it can be mapped to
- Upon request, can answer "where are you now?" (current state query) or "what is the path you took?" (history state query)



- SLIM combines top-down and bottom-up parsing
- Bottom up: At each step, grow the tree up to the first stage that still has missing observations.
- Top Down: To reach a root node, treat each *fragment* of a tree as a node and combine them in the PHATT-style



- Combines compactness and expressibility
- Using Combinatory Categorial Grammars (CCGs) and probability
- Instead of Make Breakfast $\rightarrow$ Make Drink, Make Food | ϕ, represent: Goal-Breakfast := (Make Breakfast \ Make Food)/Make Drink



Technical Resource 1: The PlanRec Hub

Figure 1: This is also where you will be able to find these slides.

Implicit Symbolic Approaches (Representative Example: Deterministic planners)

```
(:action make-coffee
    :parameters (?a - agent ?b - object)
    :condition  (and (at start (on-table ?b)))

    :effect     (and (at end (not (on-table ?b)))
                     (at end    (coffee-made))))
```

```
(:action pour-coffee
    :parameters (?a - agent ?m - object)
    :condition  (and (at start (coffee-made))
                     (at start (on-table ?m)))

    :effect     (and (at end (drink-ready))))
```

Achieve drink-ready

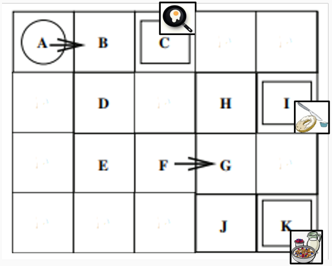
```
(:goal
    (and (food-ready)
         (drink-ready))))
```

```
(:action toast
    :parameters (?a - agent ?b - object)
    :condition  (and (at start (on-table ?b)))

    :effect     (and (at end (toasted ?b))))
```

```
(:action add-spread
    :parameters (?a - agent ?b ?c - object)
    :condition  (and (at start (on-table ?b))
                     (at start (on-table ?c))
                     (at start (toasted ?b)))

    :effect     (and (at end (food-ready))))
```

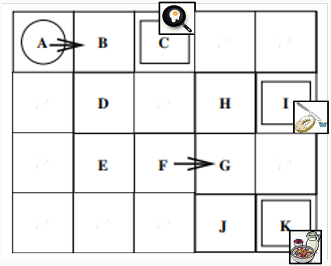
Achieve food-ready

Plan Recognition as Planning [Ramirez and Geffner, 2009, 2010]



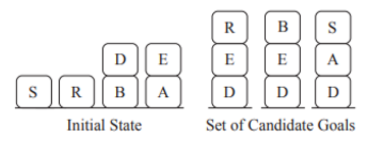
- Possible Goals (G): $\{at(C)\}, \{at(I)\}, \{at(K)\}$
- Observations (O): arrows
- $\forall g_i \in G$: $L(g_i \mid O) = C_i(O) - C_i(\neg O)$
  - $C_i(O)$ - the cost of reaching g_i while going through O
  - $C_i(\neg O)$ - the cost of reaching g_i without going through O
- $p(g_i \mid O) \cong \frac{1}{e^{\beta \cdot L(g_i \mid O)} + 1}$

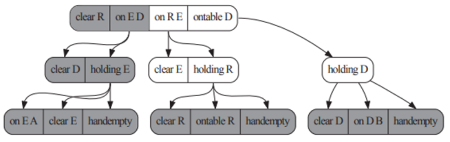
- Comparing to the k-best plans (or diverse plans) for each goal $g_i \in G$
- Reasons about noisy and missing observations
- Given O and g_i, the cost of plan π that meets g_i and satisfies O is: $cost_{g_i,O}(\pi) = cost(\pi) + b_1 M_{g_i,O}(\pi) + b_2 N_{g_i,O}(\pi)$ where $M_{g_i,O}(\pi)$ is number of missing obs and $N_{g_i,O}$ noisy obs



Landmark-based Heuristics for Goal Recognition [Pereira et al., 2017]

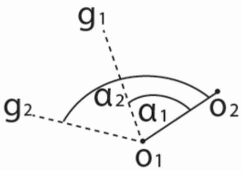
- Uses landmarks to improve runtime
- Heuristic 1: Estimate proximity to each goal (what is the ratio between achieved and not-achieved landmarks)
- Heuristic 2: Add weights to landmarks according to their uniqueness





Heuristic Online Goal Recognition in Continuous Domains [Vered and Kaminka, 2017]

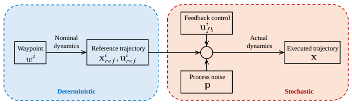
- Enhancing PRaP to continuous domains
- Proposes two heuristics inspired by mirroring neurons:
  - RECOMPUTE - recomputes new plans only if the new observations seem to change the plan significantly
  - PRUNE - prunes unlikely goals (reduces $|G|$)

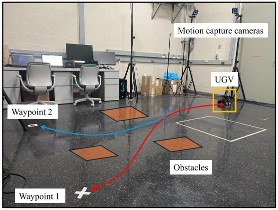


/ Online Waypoint Recognition of Controlled Agents in Uncertain Environments [Guo et al., 2025]

An alternative underlying continuous representation: instead of relying on a high-level abstraction of the environment and of its dynamics, **online waypoint recognition (OWR)** incorporates knowledge about the dynamic models into the analysis of the observed agent behavior.

- **Input:** The dynamics model of the robot and a set of possible waypoints
- **Output:** A Kalman filter is used to perform waypoint recognition at high frequency.





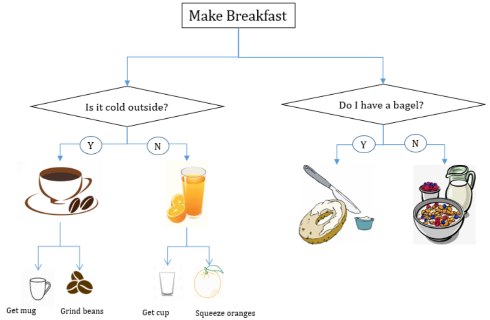
Technical Resource 2: Masterclass on Plan Recognition as Planning

Figure 2: Thanks to Felipe Meneguzzi for sharing this resource with us

Why not just use an LLM to solve this?

Existing approaches:

- Build explicit plan libraries
- Build implicit plan libraries
- Build policies directly



Goal Recognition using Off-the-Shelf Process Mining Techniques [Polyvyanyy et al., 2020]

- Builds an explicit plan library: all likely trajectories per goal
- Construct graph using off-the-shelf data mining techniques — generate *process nets*
- Goal recognition over alignments: how similar is the observed trajectory to each of the process model?

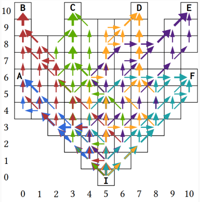


Figure 3: Example walks

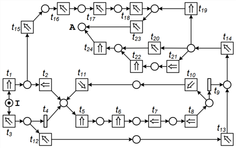
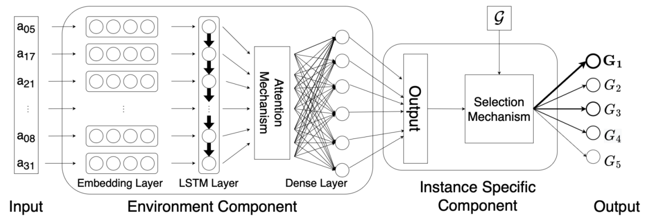


Figure 4: Process graph

Goal recognition as a deep learning task: The GRNet approach [Chiari et al., 2023]

Learning an implicit plan library, GRNet has two main components:

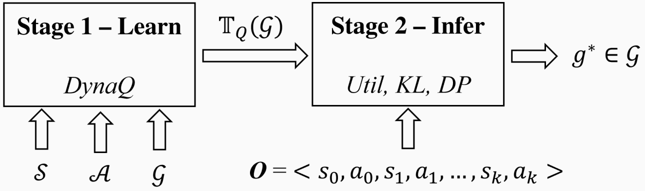
- Environment: answers for each action sequence in the training set "*does the fluent f true at the end?*". The output vector is the list of all fluents
- Instance-specific: given a new observation sequence, pass through the environment component and count the number of fluents that are true for each goal



Model-free, in the sense that the policies per-goal are learned directly instead of the environment or transition model:

- Learning: train a policy for each of the goals out of a goal set
- Inference: use distance measures to quantify the likelihood of each goal. E.g. MaxUtil using Q-learning:

$$MaxUtil(Q_g, O) = \sum_{i \in |O|} Q_g(s_i, a_i)$$



Technical Resource 3: The GRLib Repository

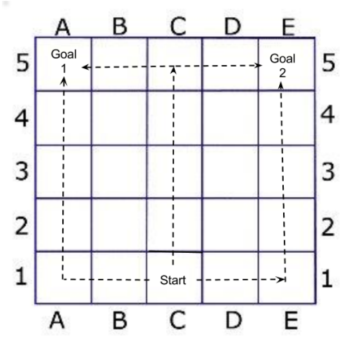
Figure 5: Goal Recognition as RL variants, based on the Gymnasium API.

Goal Recognition Design [Keren et al., 2014]

When designing an environment, can we adjust it so recognition will be easier?

Not a recognition task: An offline rather than an online task: compute *Worst Case Distinctiveness (WCD)* and minimize it

- **Input:** A PDDL of an environment and a set of goals
- **Output:** A modified PDDL with minimal WCD

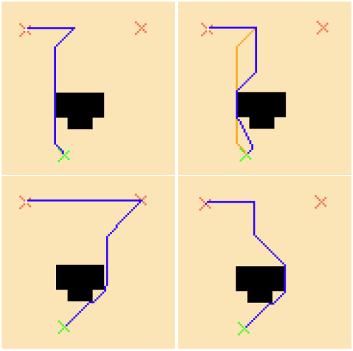


Deceptive Path-Planning [Masters and Sardina, 2017]

How to choose a path to hide your goal for as much as possible?

Not a recognition task: Playing the role of the actor rather than the observer: compute *Last Deceptive Point (LDP)* and maximize it

- **Input:** A PDDL of an environment and a set of goals, including the true goal
- **Output:** A path that deceives the observer

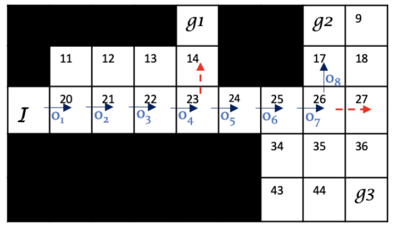


Towards Explainable Goal Recognition Using Weight of Evidence (WoE): A Human-Centered Approach [Alshehri et al., 2025]

Can we provide a counterfactual explanation for the observer's choice of some goal over another?

Not a recognition task: Instead of inferring the goal of the actor, providing an explanation for this choice

- **Input:** A GR task
- **Output:** Explanation for the chosen goal



Also:

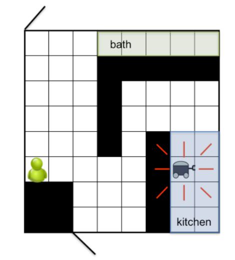
- Generating Impact and Critique Explanations of Predictions made by a Goal Recognizer [Junior et al., 2025]

Active Goal Recognition [Shvo and McIlraith, 2020, Amato and Baisero, 2019]

If the observer can act to change its sensing, how can it optimize its actions to improve recognition?

Not a recognition task: Instead of inferring the goal of the actor, planning to minimize ambiguity.

- **Input:** A GR task with a current state and a set of potential actions
- **Output:** Action to take by the observer to improve recognition



Also:

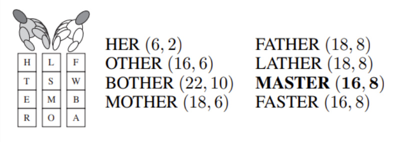
- A Penny for Your Thoughts: The Value of Communication in Ad Hoc Teamwork [Mirsky et al., 2020]

Concurrent plan recognition and execution for human-robot teams [Levine and Williams, 2014]

How to leverage successful recognition into teamwork?

Not a recognition task: Instead of inferring the goal of the actor, planning to improve task completion.

- **Input:** A GR task with a current state and a set of potential actions
- **Output:** Observer action to take to improve teamwork



Also:

- Integration of planning with recognition for responsive interaction using classical planners [Freedman and Zilberstein, 2017]
- Too many cooks: Bayesian inference for coordinating multi-agent collaboration [Wu et al., 2021]

Open Challenges

Open Challenges

- Recognition of agents pursuing more than one goal.
- Recognition from observations that mapped to more than one plan.
- Recognition of plan executed by a team.
- Recognition using LLMs
- Learning-related: sim2real, meta-learning of goal sets [Elhadad et al., 2026].

References

A. Alshehri, A. Abdulrahman, H. Alamri, T. Miller, and M. Vered. Towards explainable goal recognition using weight of evidence (woe): A human-centered approach. *Journal of Artificial Intelligence Research*, 82:2535–2594, 2025.

L. Amado, R. Mirsky, and F. Meneguzzi. Goal recognition as reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 9644–9651, 2022.

C. Amato and A. Baisero. Active goal recognition. *arXiv preprint arXiv:1909.11173*, 2019.

D. Avrahami-Zilberbrand and G. A. Kaminka. Fast and complete symbolic plan recognition. In *IJCAI*, pages 653–658, 2005.

M. Chiari, A. Gerevini, F. Percassi, L. Putelli, I. Serina, and M. Olivato. Goal recognition as a deep learning task: The grnet approach. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, 2023.

O. Elhadad, O. Morrissey, and R. Mirsky. General dynamic goal recognition using goal-conditioned and meta reinforcement learning. In *The 25th International Conference on Autonomous Agents and Multi-Agent Systems*, 2026.

R. Freedman and S. Zilberstein. Integration of planning with recognition for responsive interaction using classical planners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

C. W. Geib. Delaying commitment in plan recognition using combinatory categorial grammars. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2009.

C. W. Geib and R. P. Goldman. A probabilistic plan recognition algorithm based on plan tree grammars. volume 173, pages 1101–1132. Elsevier, 2009.

J. Guo, S. Surve, Z. He, S. Ferrari, and S. Keren. Online waypoint recognition of controlled agents in uncertain environments. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12600–12606. IEEE, 2025.

J. d. S. F. Junior, I. Zukerman, E. Makalic, C. Paris, and M. Vered. Generating impact and critique explanations of predictions made by a goal recognizer. In *Proceedings of the 18th International Natural Language Generation Conference*, pages 554–575, 2025.

S. Keren, Y. Gal, and E. Karpas. Goal recognition design. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, 2014.

S. Levine and B. Williams. Concurrent plan recognition and execution for human-robot teams. In *Proceedings of the international conference on automated planning and scheduling*, volume 24, pages 490–498, 2014.

T. Masters and S. Sardina. Deceptive path planning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

R. Mirsky and Y. K. Gal. Slim: Semi-lazy inference mechanism for plan recognition. In *IJCAI International Joint Conference on Artificial Intelligence*, volume 2016, pages 394–400, 2016.

R. Mirsky, W. Macke, A. Wang, H. Yedidsion, and P. Stone. A penny for your thoughts: The value of communication in ad hoc teamwork. In *29th International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 254–260. International Joint Conferences on Artificial Intelligence, 2020.

R. Pereira, N. Oren, and F. Meneguzzi. Landmark-based heuristics for goal recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

A. Polyvyanyy, Z. Su, N. Lipovetzky, and S. Sardina. Goal recognition using off-the-shelf process mining techniques. In *Proceedings of the 19th international conference on autonomous agents and multiAgent systems*, pages 1072–1080, 2020.

M. Ramirez and H. Geffner. Plan recognition as planning. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1778–1783, 2009.

M. Ramirez and H. Geffner. Probabilistic plan recognition using off-the-shelf classical planners. In *AAAI Conference on Artificial Intelligence*, volume 24, pages 1121–1126, 2010.

M. Shvo and S. A. McIlraith. Active goal recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9957–9966, 2020.

S. Sohrabi, A. Riabov, and O. Udrea. Plan recognition as planning revisited. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3258–3264, 2016.

M. Vered and G. A. Kaminka. Heuristic online goal recognition in continuous domains. In *International Joint Conference on Artificial Intelligence 2017*, pages 4447–4454. Association for the Advancement of Artificial Intelligence (AAAI), 2017.

References v

S. A. Wu, R. E. Wang, J. A. Evans, J. B. Tenenbaum, D. C. Parkes, and M. Kleiman-Weiner. Too many cooks: Bayesian inference for coordinating multi-agent collaboration. *Topics in Cognitive Science*, 13(2):414–432, 2021.