Dynamic probabilistic logic models for effective task-specific abstractions in RL

Harsha Kokel
Dynamic probabilistic logic models for effective task-specific abstractions in RL

Harsha Kokel
Nikhilesh Prabhakar
Sriram Natarajan
Arjun Manoharan
Balaraman Ravindran
Erik Blasch
Prasad Tadepalli

With support from DARPA, AFOSR, ARO, NSF & USDA-NIFA, and RBCDSAI
Raedt et al. 2016; Raedt et al. 2020
Parkinson’s disease prediction

Drug-Drug Interactions

Chemical Entities of Biological Interest (ChEBI)

Social Networks

Cohort of Pregnant Women (nuMoM2b)

Collaborative Problem Solving

Dhami et al. 2017; Dhami et al. 2022; Karanam et al. 2022; Mathur et al. 2023; Dhami et al. 2017; Das et al. 2022; Ramanan et al. 2021; Kaur et al. 2020; Kokel et al. 2021
Cohort of Pregnant Women (nuMoM2b)

Chemical Entities of Biological Interest (ChEBI)

Collaborative Problem Solving

Dhami et al. 2017; Dhami et al. 2022; Karanam et al. 2022; Mathur et al. 2023; Dhami et al. 2017; Das et al. 2022; Ramanan et al. 2021; Kaur et al. 2020; Kokel et al. 2021
How to facilitate generalizable, effective and efficient learning with human guidance?
Relational domains

Non-IID domains with varying # objects and heterogeneous relations.
Abstract Representations

Konidaris, G., 2019; Li et al 2006
Given: Relational sequential decision-making domain
To do: Learn an efficient agent that
• is compositional
• can handle varying # of objects
• can generalize to different tasks
• can support task-specific representations
• can handle multi-modal data
RePReL

- Plan the sequence of high level subgoals and learn to execute each subgoal at lower level

Grounds and Kudenko 2008; Yang et al. 2018; Jiang et al. 2019; Eppe et al. 2019; Illanes et al. 2020; Lee et al. 2020; Mitchener et al. 2022; Lyu et al. 2019; Goel et al. 2022; Planning and RL workshop
RePReL

Goal directed relational MDP:
\(<S, A, P, R, \gamma, G>\)

Dietterich 1999
Definition 3. The subgoal RMDP $M_o$ for each operator $o$ is defined by the tuple $\langle S, A, P_o, R_o, \gamma \rangle$ consisting of states $S$, actions $A$, transition function $P_o$, reward function $R_o$, and discount factor $\gamma$. State and Actions remain same as the original RMDP. The reward function $R_o$ and transition probability distribution function $P_o$ are defined as follows:

$$R_o(s, a, s') = \begin{cases} t_R + R(s, a, s') & \text{if } s' \in \beta(o) \text{ and } s \notin \beta(o) \\ 0 & \text{if } s' \in \beta(o) \text{ and } s \in \beta(o) \\ R(s, a, s') & \text{otherwise} \end{cases}$$

$$P_o(s, a, s') = \begin{cases} 0 & \text{if } s \in \beta(o) \text{ and } s' \notin \beta(o) \\ 1 & \text{if } s \in \beta(o) \text{ and } s' \in \beta(o) \\ P(s, a, s') & \text{otherwise} \end{cases}$$

with $R(s, a, s')$ indicating the reward function from the original GRMDP definition. $t_R$ is a fixed terminal reward.
RePReL

• Plan the sequence of high level subgoals and learn to execute each subgoal at lower level

• Advantage:
  • Compositionality
  • Task specific state representations

• Dynamic First Order Conditional Influence (D-FOCI) statements to obtain task-specific abstract representations

Kokel et al ICAPS 2021
D-FOCI

First Order Conditional Influence (FOCI) statements

\[
\text{if } \langle \text{condition} \rangle \text{ then } \langle \text{influent} \rangle \text{ QINF } \langle \text{resultant} \rangle
\]

Dynamic FOCI statements

\[
[\text{subgoal:}] \; \langle \text{influent} \rangle ^{[+1]} \rightarrow \langle \text{resultant} \rangle
\]

Natarajan, Tadepalli, Dietterich, and Fern 2008
D-FOCI as Dynamic PLMs

![Diagram showing Bayesian Network (BN) and Dynamic Probabilistic Logic Models (PLM, PRM, BLP, LBN, DAPER)]

Koller and Friedman 2009; Getoor and Tasker 2007; Raedt et al. 2016
D-FOCI example

\[
\{\text{action, taxi}_\text{at}(X)\} \xrightarrow{+1} \text{taxi}_\text{at}(X) \quad (3a)
\]

\[
\text{pick}(P) : \{\text{action, taxi}_\text{at}(X), \text{at}(P, Y), \}
\]

\[
\text{in}_\text{taxi}(P) \xrightarrow{+1} \text{in}_\text{taxi}(P) \quad (3b)
\]

\[
\text{pick}(P) : \{\text{in}_\text{taxi}(P)\} \xrightarrow{} \text{Reward} \quad (3c)
\]

\[
\text{drop}(P) : \{\text{at}_\text{dest}(P)\} \xrightarrow{} \text{Reward} \quad (3d)
\]

\[
\text{drop}(P) : \{\text{at}(P, X), \text{dest}(P, D), \text{at}_\text{dest}(P)\}
\]

\[
\quad \xrightarrow{} \text{at}_\text{dest}(P) \quad (3e)
\]

\[
\text{drop}(P) : \{\text{action, taxi}_\text{at}(X), \text{at}(P, Y), \}
\]

\[
\text{in}_\text{taxi}(P) \xrightarrow{+1} \text{at}(P, K) \quad (3f)
\]
### Abstraction

**Depth 1 unrolling:**
1. Find a substitution that grounds relevant D-FOCI statements that have reward on RHS
   
   \[
   \text{pick}(p_1): \text{in}_{\text{taxi}}(p_1) \rightarrow \text{Reward} \\
   \rho = \{P/p_1\}
   \]
2. Collect LHS in relevant literals set \(\hat{s}\)
   
   \[
   \hat{s} \leftarrow \{\text{in}_{\text{taxi}}(p_1)\}
   \]

**Depth 2 unrolling:**
1. Find a substitution that grounds relevant D-FOCI statements that have a relevant literal on RHS
   
   \[
   \text{pick}(P): \{\text{action, taxi}_{\text{at}}(l_3), \text{at}(p_1, r), \text{in}_{\text{taxi}}(p_1)\} \rightarrow \text{in}_{\text{taxi}}(p_1) \\
   \rho = \{P/p_1, X/l_3, Y/r\}
   \]
2. Collect LHS in set \(\hat{s}\)
   
   \[
   \hat{s} \leftarrow \{\text{in}_{\text{taxi}}(p_1), \text{action, taxi}_{\text{at}}(l_3), \text{at}(p_1, r)\}
   \]

**Depth 3 unrolling:**
1. Ground applicable D-FOCI statements
   
   that have a relevant literal \(\rho\) on RHS
   
   \[
   \{\text{action, taxi}_{\text{at}}(l_3)\} \rightarrow \text{taxi}_{\text{at}}(l_3) \\
   \text{pick}(p_1): \{\text{action, taxi}_{\text{at}}(l_3), \text{at}(p_1, r), \text{in}_{\text{taxi}}(p_1)\} \rightarrow \text{in}_{\text{taxi}}(p_1) \\
   \rho = \{P/p_1, X/l_3, Y/r\}
   \]
2. Collect LHS in set \(\hat{s}\)
   
   \[
   \hat{s} \leftarrow \{\text{in}_{\text{taxi}}(p_1), \text{action, taxi}_{\text{at}}(l_3), \text{at}(p_1, r)\}
   \]

---

**Given:**
- D-FOCI statements from Equation 3
- state \(s = \{\text{at}(p_1, r), \text{taxi}_{\text{at}}(l_3), \text{dest}(p_1, d1), \neg\text{at}_{\text{dest}}(p_1), \neg\text{in}_{\text{taxi}}(p_1), \text{at}(p_2, b), \neg\text{at}_{\text{dest}}(p_2), \neg\text{in}_{\text{taxi}}(p_2)\}\)
- grounded option \(\rho\): \(\text{pick}(P) \{P/p_1\}\)

**Output:** A set of relevant state literals: \(\hat{s}\)
RePReL Learning

- Initialize buffers
- Get high level plan
- For each subgoal
  - Loop till the subgoal is achieved or # steps exceeds
    - Get the abstract state
    - Get the policy for that subgoal
    - Take a step and observe reward, next state
    - Add <S, A, R, S> to the buffer
- Update the subgoal policy using samples from the buffers

Kokel et al. 2021a; Kokel et al. 2021b
Hybrid Deep RePReL

Initial state

Symbolic Planner

subgoals

Abstraction Reasoner

Input pre-processor

Merge

RL agents

Environment

D-FOCI

Manhaeve et al. 2018; Kokel et al. 2022
Given: Relational sequential decision-making domain

To do: Learn an efficient agent that
  • is compositional
  • can handle varying # of objects
  • can generalize to different tasks
  • can support task-specific representations
  • can handle multi-modal data
Experiments

- **Domains**
  - Office World
  - Craft World
  - Relational Taxi
  - Relational Box World
  - Fetch Pick and Place

- **Baselines**
  - Tabular RL
  - Deep RL (DDQN, PPO, SAC)
  - Hierarchical RL (options framework)
  - Planner + RL (Taskable RL)
  - Deep Relational RL (ReNN)

Illanes et al. 2020; Li et al 2020;
Sample Efficiency

![Graph showing sample efficiency with steps in environment on the x-axis and episode reward on the y-axis. Lines represent different learning algorithms: RePReL, trl, hrl, and ql.]

![Graph showing average reward with time on the x-axis and reward on the y-axis. Lines represent different learning algorithms: RePReL, TRL, HDQN, and DQN.]
Sample Efficiency

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>▲</td>
<td>Agent</td>
</tr>
<tr>
<td>⋄</td>
<td>Furniture</td>
</tr>
<tr>
<td>Ⓔ</td>
<td>Coffee machine</td>
</tr>
<tr>
<td>⚪</td>
<td>Mail room</td>
</tr>
<tr>
<td>⚫</td>
<td>Office</td>
</tr>
</tbody>
</table>

A, B, C, D Marked locations

![Graph showing sample efficiency](image)

![Graph showing average reward](image)
Task Transfer

![Diagram with a grid and symbols]

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>△</td>
<td>Agent</td>
</tr>
<tr>
<td>*</td>
<td>Furniture</td>
</tr>
<tr>
<td>🚛</td>
<td>Coffee machine</td>
</tr>
<tr>
<td>💌</td>
<td>Mail room</td>
</tr>
<tr>
<td>📎</td>
<td>Office</td>
</tr>
</tbody>
</table>

A, B, C, D Marked locations

Tabular

Deliver coffee

Deliver mail
Task Transfer

CRAFT WORLD

Get wood and iron

Make stick
(at workbench, needs wood)
Varying # of objects

Transport 2 passengers

Transport 3 passengers
Varying # of objects

- Open lock
- Collect key and open lock
- Collect key and open 2 locks

Graphs showing episode reward vs. steps in environment for different methods: RePReL, trl, RePReL+T, trl+T.
Varying # of objects
Multi modal

transport one passenger
Transport two passengers
Transport three passengers
Multi modal

- Make Bread
- Build a house
- Break a rock
Given: Relational sequential decision-making domain
To do: Learn an efficient agent that
• is compositional
• can handle varying # of objects
• can generalize to different tasks
• can support task-specific representations
• can handle multi-modal data
Summary

- Combined a symbolic planner with RL agents
- Provide a batch learning algorithm
- Demonstrate **sample efficiency**, that is significant reduction in the number of steps required for the model to learn an optimal policy for the task
- Demonstrate **efficient generalization** over number of objects
- Provide hybrid approach for structured and unstructured data
- Most importantly, the framework is planner agnostic and RL algorithm agnostic
Future work

- Refine the D-FOCI statements
- Relax downward refinement
- Partial observability and uncertainty over states
- Boolean task algebra style compositions
Questions?
References

StarAI
Raedt, L.D., Dumančić, S., Manhaeve, R., & Marra, G. (2020). From statistical relational to neuro-symbolic artificial intelligence. IJCAI.

Starling lab: Parkinson’s Patient

Starling lab: Cohort of pregnant women

References

Starling lab: Drug-Drug Interaction

Starling lab: ChEBI

Starling lab: Social Network

Starling lab: Games
References

Abstraction

Planning + RL
Lyu, D., Yang, F., Liu, B. and Gustafson, S., 2019, July. SDRL: interpretable and data-efficient deep reinforcement learning leveraging symbolic planning. In AAAI.
References

Taxi domain

FOCI

Graphical Models

Neural Predicate

RePreL and HDRePreL
References


Baselines


Capacity loss in Deep RL

Lyle, C., Rowland, M. and Dabney, W., 2022. Understanding and preventing capacity loss in reinforcement learning. ICLR.
THANKS