Context Tree Maximizing Reinforcement Learning (CTMRL)

March 2, 2012

Phuong Nguyen\textsuperscript{1,2} (PhD student)
Joint work with Peter Sunehag\textsuperscript{1} and Marcus Hutter\textsuperscript{1,2}
Australian National University\textsuperscript{1}, National ICT Australia\textsuperscript{2}
Reinforcement Learning (RL) Approach to Artificial Intelligence (AI)

RL is an area of AI in which the agent learns a task through interactions with the environment.
Problem formulation

\[ h_t = o_0a_1o_1r_1o_2r_2a_2 \ldots o_tr_t \]
\[ a_t = \text{Agent}(h_t) \]
\[ o_{t+1}r_{t+1} = \text{Environment}(h_t, a_t) \]

- **General Reinforcement Learning (GRL) Problem**: find the agent function to maximize the total reward given that the environment’s model and states are both unknown
  - **Example**:

  ![Diagram](image)

  - **Special case**: Markov Decision Processes (MDP) where observations are states of the environment, \( s_t = o_t \)
Feature Reinforcement Learning - ΦMDP

\[ h_t = a_1 o_1 r_1 o_2 r_2 a_2 \ldots o_t r_t \]
\[ a_t = \text{Agent}(h_t) \]
\[ o_{t+1} r_{t+1} = \text{Environment}(h_t a_t) \]
\[ s_t = \Phi(h_t) \]
\[ \text{Cost}(\Phi|h_n) = \text{CL}(r_{1:n}|s_{1:n}, a_{1:n}) + \text{CL}(s_{1:n}|a_{1:n}) \]
Context tree maximizing for binary sequence prediction

- Context tree source

\[ S = \{0, 01, 11\} \]
\[ \theta_{01} = P(\text{next\_bit} = 1|\text{current\_context} = s = 01) \]

- Binary sequence prediction problem: find the optimal context tree given a history \( x_{1:n} = 010010 \ldots 01 \) and an initial context \( x_{1-D:0} = 0100 \ldots 1 \)

Cost function

\[
\min_{S \subseteq C_D} \left[ \log \frac{1}{P_c(x_{1:n}|x_{1-D:0}, S)} + \Gamma_D(S) \right]
\]

\[
P_c(x_{1:n}|x_{1-D:0}, S) = \prod_{s \in S} P_e(a_s, b_s)
\]

* Based on MDL (Minimum Description Length) principle, and arithmetic coding*
Iterative procedure to find the optimal tree

Maximized probability

\[ P_{m,s}^D := 2^{-\Gamma_{D-d}(S_{m,s}^D)} \prod_{u \in S_{m,s}^D} P_{e}^{a_{us}, b_{us}} \]

\[ = \max_{U \subseteq C_{D-d}} 2^{-\Gamma_{D-d}(U)} \prod_{u \in U} P_{e}^{a_{us}, b_{us}} \]

\[ P_{m,s}^D := \begin{cases} \frac{1}{2} \max(P_{e}(a_{s}, b_{s}), P_{m,0s}^D P_{m,1s}^D) & \text{for } 0 \leq l(s) < D \\ P_{e}(a_{s}, b_{s}) & \text{for } l(s) = D \end{cases} \]
Iterative procedure to find the optimal tree

Maximizing set $S^D_{m,s}$

$$S^D_{m,s} := \begin{cases} 
S^D_{m,0s} \times 0 \cup S^D_{m,1s} \times 1 & \text{if } P_e(a_s, b_s) < P^D_{m,0s} P^D_{m,1s}, \\
\{\epsilon\} & \text{and } 0 \leq l(x) < D \\
\text{else}
\end{cases}$$

Theorem [Willems et al, 2000]: $S^D_{m,\epsilon}$ is the optimal solution of the cost function for binary sequence prediction
Context Tree Maximizing Reinforcement Learning (CTMRL)

- **Instance Context Tree**: context tree with the instance set
  \[
  \mathcal{X} = \{x^1, x^2, \ldots, x^{\mathcal{|X|}}\} = \{aor : a \in \mathcal{A}, o \in \mathcal{O}, r \in \mathcal{R}\}
  \]
  \[
  (x_t = a_{t-1}o_tr_t \text{ is the instance at time } t)
  \]

- **Cost function**

\[
\begin{align*}
\min_{S \subset \mathcal{C}_D} & \left[ \log \frac{1}{P_c(s_{1:n}r_{1:n}|a_{1:n}, h_0)} + \Gamma_D(S) \right] \\
= & \min_{S \subset \mathcal{C}_D} \left[ \sum_a \sum_s \log \frac{1}{P_e^{x|sa}} + \Gamma_D(S) \right]
\end{align*}
\]

where \( S \) is the state set of some instance context tree,
\( s_i = \Phi_S(h_i), i = 1, n \); and \( P_e^{x|sa} \) is the block probability of all instances
Context tree maximizing iterative procedure

Maximizing probability

\[
P^D_{m,s} := 2^{-\Gamma_{D-d}(S^D_{m,s})} \prod_{a \in A} \prod_{u \in S^D_{m,s}} P^x_{e|usa}
\]

\[
= \max_{\mathcal{U} \subseteq \mathcal{C}_{D-d}} 2^{-\Gamma_{D-d}(\mathcal{U})} \prod_{a \in A} \prod_{u \in \mathcal{U}} P^x_{e|usa}
\]

\[
P^D_{m,s} := \begin{cases} 
\frac{1}{2} \left( \max_{a \in A} \left( \prod_{a \in A} P^x_{e|sa}, \prod_{i} P^D_{m,x^i|s} \right) \right) & \text{if } 0 \leq l(s) < D \\
\prod_{a \in A} P^x_{e|sa} & \text{if } l(s) = D
\end{cases}
\]
Context tree maximizing iterative procedure

- Maximizing state set

\[ S_{m,s}^D := \begin{cases} \bigcup_{x^i} S_{m,x^i s}^D \times x^i & \text{if } \prod_{a \in A} P_{e|s}^{x|sa} < \prod_{a \in A} \prod_{i} P_{m,x^i s}^D, \\
\{\epsilon\} & \text{and } 0 \leq l(s) < D \\
\text{else} & 
\end{cases} \]

- Theorem [direct extension]: \( S_{m,\epsilon}^D \) is the optimal solution of the CTM-GRL cost function

- Problematic in estimating the multivariate block probability

\[ P_{e|s}^{x|sa} := P_{e|s}^{x|sa}(n_{x^1}, n_{x^2}, \ldots, n_{x|\mathcal{I}|}) \]
CTM-GRL: binarization and factorization

- The primary purpose of binarization is to overcome the estimation problem in $P_e^x|sa$.

- Binarize observations, actions, rewards of a history. Each instance is represented in binary form as $x = aor = a[1 \ldots l_a]o[1 \ldots l_o]r[1 \ldots l_r] = ap = ap[1 \ldots l_p]$. Consider the set of models $M = (M_1, \ldots, M_{l_p}) \in C_D \times \ldots \times C_{D+l_p-1}$

\[
\text{Cost}(M|h_n, h_0) = \log \frac{1}{P_c(h_n|a_0:n-1, h_0, M)} + \sum_{i=1}^{p} \Gamma(M_i)
\]

\[
= \sum_{i=1}^{p} \left[ \sum_{t=1}^{n} \log \frac{1}{P_c(p_t[i]|h_t^i, h_0, M_i)} + \Gamma(M_i) \right]
\]

where $h_t^i = h_{t-1}a_{t-1}p_t[1 \ldots i-1]$. 

Context Tree Maximizing Reinforcement Learning (CTMRL)
1. Generate a random history $h$

2. Learn (update) $l_p$ binary CTMs based on history $h$ ($h'$ from the second iteration)

3. Join learnt contexts from each of the CTMs to form AOCT $\mathcal{T}$

4. Compute frequency estimates of state transition and reward probabilities of the MDP model $\hat{M}$ based on states induced from tree $\mathcal{T}$ and history $h$

5. Use AVI to find an estimate of optimal action values $\hat{Q}$ based on $\hat{M}$

6. (Optional) Evaluate the current optimal policy induced from $\hat{Q}$

7. $Q \leftarrow \hat{Q} + \frac{R_{\text{max}}}{1-\gamma}$ [[Optimistic Initialization]]

8. $h' \leftarrow \text{Q-learning}(Q, S^\mathcal{T}, A, \text{Environment}, n_i)$

9. $h \leftarrow [h, h']$

10. Repeat 2-9

11. $\hat{Q}' \leftarrow \text{Q-learning}(Q, S^\mathcal{T}, A, \text{Environment}, n_q)$

12. $\pi^*(s) \leftarrow \arg\max_a \hat{Q}'(s, a)$ for all $s \in S^\mathcal{T}$
Related work

- ΦMDP
- MC-AIXI-CTW
- U-tree
- Active-LZ
Results - small domains

- Cheese maze

![Cheese Maze Diagram]

Context Tree Maximizing Reinforcement Learning (CTMRL)
Results - small domains

- Kuhn poker

Kuhn Poker

- CTMRL
- PhiMDP
- MC-AIXI
- U-Tree
- Active-LZ
- Optimal

Graph showing the average reward per cycle for Kuhn poker over different experience (cycles) with various algorithms.
Results - small domains

- Extended tiger

![Extended Tiger](image)

- Other small domains: tiger, gridworld
Results - large domain

Modifications of CTMRL algorithm for large domains:

- Deletion of CTM trees after each learning loop (saving memory)
- Adding of unseen scenarios (dealing with huge observation space)
- Running Q-learning for a long time after the learning loop (solving MDPs with a large state space)
Results - large domain

- Pacman

**Partially Observable Pacman**

![Graph showing learning curves for CTMRL and MC-AIXI over experience (cycles).]
Results - large domain

- Pacman

**Partially Observable Pacman**

![Graph showing comparison between CTMRL and MC-AIXI in terms of average reward over time.](image)
Conclusion

- CTMRL is competitive with the state of the art MC-AIXI-CTW in terms of learning and superior to other competitors.

- Compared to MC-AIXI-CTW, CTMRL is dramatically more efficient in both computation time and memory.
Thank you!