Continual and Real-time Learning for Modeling Combat Identification in a Tactical Environment

Ying Zhao, Naval Postgraduate School, Monterey, CA, USA
Nate Derbinsky, Northeastern University, Boston, MA, USA
Lydia Y. Wong, Jonah Sonnenshein, Metron, Inc., San Diego, CA, USA
Tony Kendall, Naval Postgraduate School, Monterey, CA, USA

Problem Domain Definitions

Common Tactical Air Picture (CTAP) Process

- Collects and analyzes data from a vast network of sensors and platforms to make better decisions for defensive purposes.
- Provides situational awareness to decision-makers.

Combat Identification (CID)

- Locates and identifies critical airborne objects as friendly, hostile or neutral with high precision.

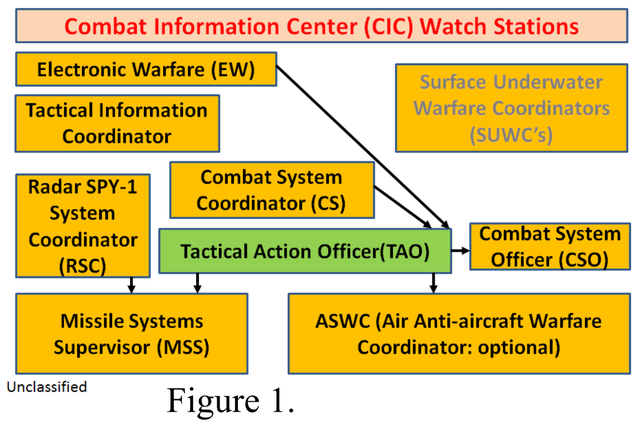

Figure 1.

- **Core question: Are machine learning (ML) and artificial intelligence (AI) systems able to learn and use the existing knowledge models for better and timely decision making for CID?**
  - Learn from the historical data with ground truth
  - Learn from the feedback of a human operator
  - Learn from the cross-validation of big data
  - Learn from the delayed ground truth after actions taken

Methodology: Combining Reinforcement Learning in Soar (Soar-RL) with Lexical Link Analysis (LLA)

Soar: Open source cognitive architecture, developed by University of Michigan, integrates reinforcement learning to modify action-selection knowledge represented as rules.
Lexical Link Analysis: Text/data mining method to discover initial correlations and rules

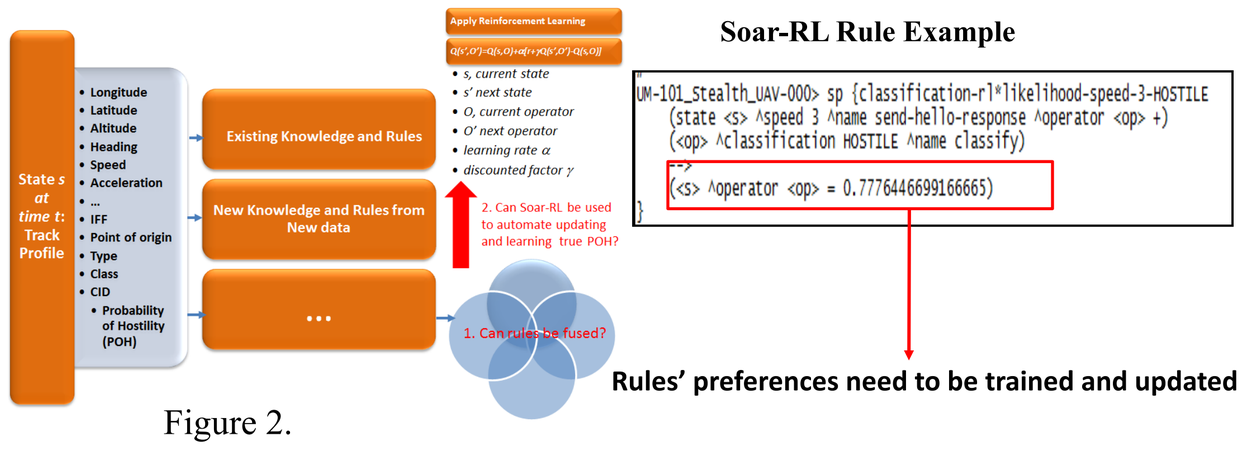

Figure 2.

As shown in Equation (1),Soar-RL is implemented in a typical RL implementation involving a recursive formula that is widely accepted in the RL research and literature. Since we only consider an on-policy setting or SARSA, $Q(s_{t+1}, a) = 0$ in Equation (1). Therefore, $Q(s_{t+1}, a_{t+1})$ is updated continuously for each time point and immediate reward r.

$$Q(s_{t+1}, a_{t+1}) = Q(s_t, a_t) + \alpha[r + \gamma \max_{a \in A} Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (1)$$

(https://soar.eecs.umich.edu/downloads/Documentation/SoarManual.pdf: Page 135)

Data from Naval Simulation System

Figure 3.

- Monte Carlo simulation of Naval platforms in a tactical scenario where the BLUE TAO tries to predict the probability of hostility of an airborne object based kinematic features such as altitude, speed, heading, and the changes in each feature
- Each scenario generated according to a different random seed number specified in NSS. Four random seeds (10,11, 1256,23576). Each different random seed generates a slightly different set of track data.
- Total 2690 tracks and 449 (16.7%) tracks are hostile.

Soar-RL/LLA Learning Results

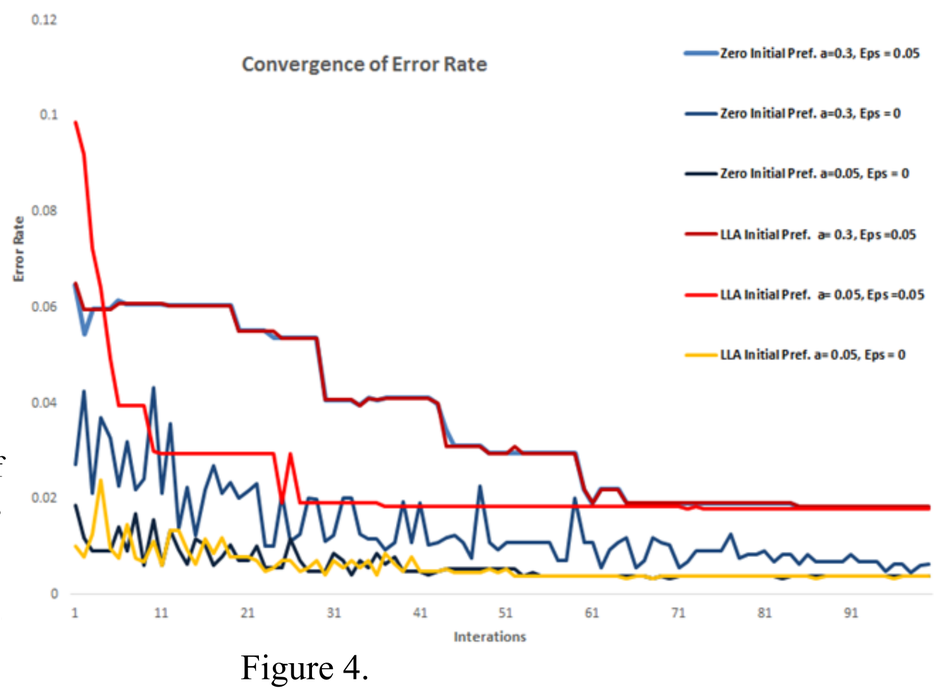

Figure 4.

Compared with Other Methods

Table 1: Classification Errors Comparison

	J48	LR	NB	Soar-RL 1	Soar-RL 2
Train (seed=10)	0.26%	0.93%	1.19%	2.8%	1.9%
Test (seed=11)	0.24%	0.75%	0.92%	1.4%	0.6%
Test (seed=1256)	1.24%	1.02%	1.17%	1.9%	1.8%
Test (seed=23576)	0.63%	1.32%	1.56%	2.2%	1.2%

Figure 4 shows classification error rates for two Soar-RL settings (1. initial preferences are computed from LLA; 2. initial preferences are zeros) decrease with more iterations when varying the parameters of Soar-RL (learning-rate alpha and epsilon).

Table 1 compares classification error rates for two Soar-RL settings with a few other supervised batch learning methods such as decision trees (i.e. J48), logistic regression (LR), and Naive Bayes (NB) from the tool Weka.

Continual and Real-time Learning Requirement: An ML/AI assistant in a tactical environment needs to be trained initially and then continues to learn and adapt to human operators' feedback or new data.

Figure 5. Soar-RL learns (and the error rate decreases) as the scenario progresses from the beginning set of track points (1-539) to the last set of track points (2154-2690).

Conclusions

- Soar-RL/LLA is a continual learning method for CID.
- The Soar-RL/LLA methodology ensures the following aspects of continual learning requirements for potential further and future studies:
  - Resistance to catastrophic forgetting – new learning of Soar-RL does not destroy performance on previously seen data;
  - Bounded system size – the Soar-RL model's capacity is fixed as reflected in the number of preferences is fixed, the system uses its capacity intelligently and converges to an estimated maximum of future reward;
  - No direct access to previous experience – while the model can remember a limited amount of experience, Soar-RL does not have direct access to past tasks or the ability to rewind the environment -- a potential benefit of Soar that could be incorporated in the future version.

Acknowledgements and Disclaimer: Thanks to the Naval Research Program at the Naval Postgraduate School for the support of the research project. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied of the U.S. Government.

Poster at the Workshop of Continual Learning (https://neurips.cc/Conferences/2018/Schedule?showEvent=10910 and https://sites.google.com/view/continual2018) held at the Thirty-second Conference on Neural Information Processing Systems (NeurIPS, 2018, https://neurips.cc/), Canada, Montreal, December 7, 2018.