Liu, Anji

E-mail: anjiliu219@gmail.com   **Phone Number:** +1 (424) 559-4740
Address: Room 211, 516 Glenrock Ave., Los Angeles, CA 90024

Education

Beihang University, School of Automation Science and Electrical Engineering   Sep. 2015 – Jun. 2019

➢ **Bachelor of Engineering in Automation**
➢ **General GPA:** 3.86/4.00, **Rank:** Top 2/175
➢ **Standardized Test Scores: TOFEL**: 103 (Speaking 21), **GRE**: 324
➢ **Honors:** Excellent Graduate in Beijing, China; Outstanding undergraduate thesis;
     Merit Student (ranked 3/120) 2016-2017; Major Scholarship Award 2015-2018;
     Excellent Student Scholarship from Aviation Industry Corporation of China 2016-2017
➢ **Key courses:** Single Variable Calculus: 99; Multivariable Calculus: 100;
     Principles of Automation Control (I): 97; Principles of Automation Control (II): 98;
     Applied Discrete Mathematics: 93; Probability and Statistics A: 100

Experiences

University of California, Los Angeles (UCLA)   Aug. 2019 – Present

➢ **Title:** Visiting Researcher
➢ **Research topic:** Tractable inference on probabilistic graphical models, and its application on reinforcement learning.

Seattle AI Lab, Kwai Inc.   Dec. 2018 – Jul. 2019

➢ **Title:** Machine Learning Intern
➢ **Project:** User pass rate prediction system for a level-oriented mobile game "joy city". The system is powered by a game AI using reinforcement learning (RL) and Monte Carlo tree search (MCTS) that achieves master-level performance in the mobile game.

Papers

[1] **Liu A**, Chen J, Yu M, Zhai Y, Zhou X, Liu J. Watch the Unobserved: A Simple Approach to Parallelizing Monte Carlo Tree Search. In ***Eighth International Conference on Learning Representations (ICLR)***, 2020. [pdf]
Oral full presentation, acceptance rate 48/2594 = 1.8%
[2] **Liu A**, Laili Y. Balance Gate Controlled Deep Neural Network. ***Neurocomputing 320: 183-194***, 2018. [pdf]
[3] **Liu A**, Liang Y, Van den broeck G. Off-Policy Deep Reinforcement Learning with Analogous Disentangled Exploration. In ***Thirty-fourth AAAI conference on Artificial Intelligence workshop on Reinforcement Learning for Games (RLG)***, 2020.
[4] **Liu A**, Liang Y, Van den broeck G. Off-Policy Deep Reinforcement Learning with Analogous Disentangled Exploration. In ***Nineteenth International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)***, 2020. Status: **under review**
[5] **Liu A**, Laili Y. Finding Good State Representations for Reinforcement Learning. ***IEEE Transacions on Cybernetics***, 2018. Status: **revising paper**

Research Experience

Research on Model-based Reinforcement Learning   Sep. 2019 – Present

Statistictical and Relational Artificial Intelligence (StarAI) Lab, UCLA   *Advisor: Guy Van den Broeck*

➢ Solving the environment model's inaccuracy problem in model-based reinforcement learning with tractable probabilistic inference models. The main idea is that by propagating environment model's uncertainty over multiple time steps, we can both improve multi-step prediction and be able to quantify the uncertainty, which

allows principled tradeoff between using environment model for planning and using on-policy methods.

Research on Exploration Strategy of Continuous Control Tasks   Jan. 2019 – Aug. 2019

Statistical and Relational Artificial Intelligence (StarAI) lab   *Advisor: Guy Van den Broeck*

➢ Disentangled the policy learning and exploration process in continuous control reinforcement learning tasks by separately design two policies that maximize their respective objectives, which offers policy optimality as well as effective exploration. The resultant model outperforms state-of-the-art approaches in various benchmarks [3, 4].

Research on Parallelizing Monte Carlo Tree Search   Jan. 2019 – Aug. 2019

Seattle AI Lab, Kwai Inc.   *Advisor: Ji Liu, Jianshu Chen*

➢ Designed a novel parallelization algorithm for Monte Carlo Tree Search as well as a practical system implementation to achieve linear speed up while suffering negligible performance degradation. The key idea is to make use of the information from unobserved samples to guide the node selection process in the tree search. This algorithm has been successfully deployed in a level-oriented mobile game to predict user pass rate, significantly reducing the game design cycle [2].

Research on Improving State Representations for Reinforcement Learning   Jul. 2017 – Sep. 2017

Simulation Center of Automation College at Beihang University   *Advisor: Yuanjun Laili*

➢ Built a learning model to extract the location of moving objects in visual RL tasks automatically without the need for supervision or labeled samples, which provide high-level state-representations that accelerate reinforcement learning significantly [5].

Research on Improving Fully Connected Neural Networks   Jan. 2017 – May. 2017

Simulation Center of Automation College at Beihang University   *Advisor: Yuanjun Laili*

➢ Studied the fundamentals of neural networks and programmed a neural network package from scratch in MATLAB. The package contains commonly used layers in neural networks and it is easy to modify.

➢ Proposed a balance gate controlled neural network that out-performs traditional fully-connected neural networks in both regression tasks and classification tasks [1].

Practical Projects

User pass rate prediction system powered by game AI   Dec. 2018 – Jul. 2019

➢ Designed an auto-testing system for the "joy city" mobile game using reinforcement learning (RL) and Monte Carlo tree search (MCTS) to replace human testers. It is used to help level designers to fine-tune the game design to meet the expected difficulty before the game is released online.

➢ Achieved **fast** (~20 minutes compared to hours for human testers) and **accurate** (e.g., predict user pass rate of levels with ~8% MAE, compared to 15-20% for human testers) auto-testing results.

Improving Heartbeat Detection Algorithm   Jan. 2018

➢ Modified an ECG data processing package to improve its detection accuracy.

➢ Made some improvements: reduced misdetection of R wave, decreased the false detection of ST wave, and able to detect high frequency and low frequency noise robustly.

SIFT Parallelization   Oct. 2016

➢ Parallelized the SIFT (Scale Invariant Feature Transform) algorithm using CUDA.

Skills

Programming languages: skilled in Python, Julia, C, C++, MATLAB