

Chang Ye

347-320-1201 | yooceii19960620@gmail.com | <https://yooceii.github.io> | github.com/yooceii

EDUCATION

New York University

Master of Science in Computer Science. 3.778/4.0

New York, NY

Aug. 2018 - May 2020

Dalhousie University

Joint Program in Computer Science. 3.44/4.3

Halifax, NS, Canada

Aug. 2017 - April 2018

Zhejiang University of Technology

Bachelor of Engineering in Software Engineering. 3.3/4.0. Ranking: 24/186

Hangzhou, ZJ, China

Aug. 2014 - Jun 2017

RESEARCH INTEREST

- Neuroscience and psychology inspired RL.
- Causal Representation learning in RL.

PUBLICATION (*=EQUAL CONTRIBUTION)

- Y. Yao*; C. Ye*; J. He; GF. Elsayed. Teacher-generated pseudo human spatial-attention labels boost contrastive learning models. *SVRHM Workshop@ NeurIPS*, 2022. [paper]
- H. Shengyi; R.F.J. Dossa; C. Ye and J. Braga. CleanRL: High-quality Single-file Implementations of Deep Reinforcement Learning Algorithms. *Journal of Machine Learning Research (JMLR)*, 2022. [paper, code]
- C. Ye; A. Khalifa; P. Bontrager and J. Togelius. Rotation, Translation, and Cropping for Zero-Shot Generalization. *IEEE Conference on Games (CoG)*, 2020. (39.9% Acceptance Rate) [paper, code, presentation]
- C. Ye*; G. Mittal*; Y. ruksachatkun*; L. Cui*. SeqG(SC)AN: SeqGAN baseline for Grounded SCAN. DS-GA 1016 Final Project. [paper, code, Course Website]
- C. Ye and M. Heywood. Uniform Cost Search in Procedural Content Generation for Angry Bird Games. **Honour Thesis**. [paper, code]

RESEARCH & PROJECTS EXPERIENCE

Boost self-supervised representation learning model with pseudo-spatial human attention

Aug 2022 - present

- Trained the model with different type of saliency distance function such as **EMD**.
- Implemented linear evaluation and image retrieval experiments for analyzing trained model's performance.
- Analyzed the model performance by calculating **mCE**, **mAP** and **PR** curve.

CleanRL open source project

July 2020 - present

- Reimplemented the **Deep Q-learning** algorithm in **JAX** and achieved the same level performance compared to the original implementation.
- Add envpool support to the **random network distillation** algorithm and reduced the runtime to **30%** of the original OpenAI VecEnv Gym environment version.
- Coauthored the technique paper and it's accepted by **JMLR** in software track.
- Reviewed major PRs and helped release the 1.0.0 version of the lib.

Consecutive-step curiosity diff as intrinsic reward for exploration in reinforcement learning

July 2020 - Jan 2021

- Reimplemented the **random network distillation** algorithm in PyTorch and achieved state-of-the-art performance. The code is contributed to the Github repository: [CleanRL](#) which has around **1.7k** stars on GitHub.
- Designed a new curiosity reward based on the difference between RND's curiosity reward in 2 consecutive steps.
- Experimented with different forms of scaling function and analyzed the intrinsic reward at the different states in game.

Policy-Dynamics Value function (PD-VF) extension

Sep 2020 - Jan 2021

- Designed a transformer encoder that takes rewards, states and actions as input, and a feed-forward network decoder that takes states and actions as input and outputs rewards. Trained network by using the ℓ_2 error of predicted rewards and real rewards as the objective function.
- Performed an ablation study by training PPO agents conditioned on learned embeddings.

Imitation Learning through watching videos

July 2019 - Aug 2019

- Implemented an tree-search algorithm by adopting the **Monte Carlo tree search** methods to play game zelda.
- Took the play-traces from tree-search agent and trained a embedding that maintains pixel sequence information by using temporal distance between 2 frames as the objective function.
- Adopted the A2C algorithm and used cosine distance between the trained embedding and current state's embedding as an extra reward to train agent on game zelda.

Representation learning for reinforcement learning

Feb 2020 - March 2020

- Designed an algorithm that learns a suitable representation while performing policy gradient by incorporating **SimCLR** structure into the standard **A2C** algorithm.

- Created an asynchronous version that learns representations simultaneously and added more data augmentation options

Human-face recognition and mosaic

Jan 2016 – May 2016

- Utilized **histogram equalization** in **OpenCV** to solve the low contrast problem that causes the face recognition algorithm unable to recognize the face features. The face recognition rate improved **30%** by adopting that technique.
- Took function from **Dlib** to extract face features and used these features to designed a module that can efficiently locate the multiple facial features on multiple faces and convert them to contours for Gaussian blur purpose.
- Designed the whole backend pipeline, implemented in **C++** under the **Qt** platform and ran backend unit tests. Built the system that is able to process up to **40** frames per sec with ignorable latency. The system achieved a **95%** detection rate in the real-time video under a low-light environment.

EXPERIENCE

Research Intern

Jan 2019 - Jul 2021

New York University, Game Innovation Lab, Supervisor: Julian Togelius

New York, NY

- Designed and executed the **MAP-Elites** algorithm to explore the generalization in policy space. Created t-SNE plots to visualize the search space.
- Conducted various research projects on **evolutionary algorithms** and **reinforcement learning**, and presented the results during the lab meeting.

Google LLC

California, United States

Software Engineer

August 2021 - Present

- Designed a system to automatically validate and alert cloud compute resource supply forecast signals in Golang by using RPC and internal toolstack that runs on multiple data centers.
- Migrated the supply forecast system to new data sources and improved the system runtime by **40%**.
- Joined the **Google brain** team as a 20%er and worked on human attention inspired representation learning model.

TEACHING

Course Assistant

New York University

New York, NY

- CS-GY 6943 AI for Game. Instructor: Julian Togelius (2020 Spring)
- ECE-GY 6143 Introduction to Machine Learning. Instructor: Anna choromanska (2019 Fall)

TECHNICAL SKILLS

Languages: Python, Go, Java, C++, C, C#, Shell, SQL, Haskell, Matlab

Software & Tools: PyTorch, JAX, Tensorflow, OpenAI Gym, OpenAI Baseline, Numpy, Scipy, Git, L^AT_EX, .Net, Vim, Tmux