Arth Shukla

Redwood City, CA | https://arth.website | https://www.linkedin.com/in/arth-shukla/ | (650) 850-9097 | arthshukla03@gmail.com

EDUCATION

University of California, San Diego September 2021 – June 2025
Bachelor of Science in Mathematics-Computer Science GPA – 4.0
Relevant Coursework
Completed: Design and Analysis of Algorithms, Advanced Data Structures, Optimizations for Data Science I and II, Computer Organization and Systems Programming, Data Structures and Object-Oriented Design, Computer Science and Object-Oriented Programming - Java, Networks and Digital Communications, Data Warehousing to Big Data, Statistical Methods and Probability, Enumerative Combinatorics, Abstract Algebra I and II, Linear Algebra, Multivariable Calculus, Vector Calculus, Differential Equations
Ongoing in Spring 2023: Supervised Machine Learning, Theory of Computation, Intro to Computational Statistics, Synthesis 100

EXPERIENCE

Personal AI Projects: https://github.com/arth-shukla; https://wandb.ai/arth-shukla/projects
Technologies Used: Development: Pytorch (Torch. TorchVision, Datasets, Dataloaders, Cuda), OpenAI Gym, HuggingFace Transformers Library, TensorFlow, Keras, Gensim; Concepts: Reinforcement Learning (PPO, DDQN, DQN), NLP (Transfer Learning, Embeddings, Attention); Tools/Technologies: WandB, BERT/DistilBERT Pretrained, Conda
Reinforcement Learning

- Use Pytorch to make PPO Agent (w/ entropy regularization, advantage normalization, early stop w/ approx. KL Div, GAE, etc) and Gymnasium to consistently beat Mario level 1-1 in under 2000 episodes of training: GitHub: https://github.com/arth-shukla/ppo-mario
- Use Pytorch to implement DDQN from *Human-level control through deep reinforcement learning*, Deep Mind 2015 in Gymnasium to beat Mario level 1-1 in under 13000 episodes of training: https://github.com/arth-shukla/ddqn-mario
- Use Pytorch to make simple PPO Agent and Gymnasium to consistently beat CartPole in under 140 episodes of training: https://github.com/arth-shukla/ppo-gym-cartpole

Natural Language Processing

- Use HuggingFace Transformers library to fine-tune DistilBERT model (transfer learning) trained on Stanford Question-Answer 2.0 (SQuAD 2.0) to answer a question given some context (article, paragraph, etc): https://github.com/arth-shukla/squad2.0-bert-question-answer
- Use Pytorch and HuggingFace to fine-tune DistilBERT model (trainsfer learning) to classify and approximate sentiment for Stanford Sentiment140 1.4-million Tweet Dataset: https://github.com/arth-shukla/sentiment140-bert-transfer-learning
- Use TensorFlow Keras to build LSTM and CNN and use Gensim to refit GLoVE word embeddings for IMDB Review Sentiment Classification: https://github.com/arth-shukla/gensim-embedding-training-imdb

Computer Vision

- Use Pytorch (Datasets and Dataloaders) and TorchVision to implement *PointNet: Deep learning on point sets for 3d classification and segmentation*, Qi et al. 2017 for part segmentation on chair point clouds: https://github.com/arth-shukla/pointnet-part-segmentation

ACM AI's Element.AI Competition https://github.com/acmucsd/Element.AI
I led development and organization of Element.AI, an $8000 competition at UCSD with 200 participants.
Technologies Used: Element.AI: Python, Conda, Java, Maven, OpenAI Gym, PettingZoo, PyGame, Jackson, Squid, Bash
Lead Developer July 2022 – Feb 2023

- Use Python with Conda and PettngZoo ParrallelEnv to create multi-agent gym environment as the core competition environment
- Use Java with Maven and Jackson to create Java sdk for 45 participants (~22.5% of all participants)
- In coordination with UCSD ITS, use Squid proxy, IPTables and bash scripts to create instructor tools, allowing us to enable/revoke access to wifi, whitelist sites, enable/revoke access to files, and in general control the competition accounts with granularity, both targeted and en masse
- Write proposals and attend meetings to obtain $8000 in sponsorships, attracting 200 participants (limited primarily by the number of UCSD linux lab machines) with hundreds of submissions

ACM AI UCSD
ACM AI is UCSD's largest AI student org which fosters a community for those interested in AI and research.
Technologies Used: AI/ML Workshops and Projects: Python, PyTorch, TensorFlow, Google Colab; Web Development: TypeScript, React, LESS

Board – President May 2023 – Present

- Lead Operations, Competitions, Dev, and Marketing teams to develop competitions and workshops, and revamp our forward-facing resources (website, GitHub, ACM AI Wiki)

Board – Director of Operations May 2022 – May 2023

- Lead team of 7 event leads in creation of competitions, workshops, and socials related to ML/AI
- Coordinate with marketing and development teams to market events and create competitions

Board – Event Lead January 2022 – May 2022

- Ideate, create, and host workshops on NLP, deep learning, and ML topics
- Develop and organize competitions run by ACM AI (100-200 submissions on average)
- Coordinate with different parts of ACM AI Board (marketing, social, and other event leads) to host workshops and social events
- Mentor intermediate and beginner ACM Projects teams in developing AI/ML projects
- Promote Diversity, Equity, and Inclusion by planning URM-focuses events, implementing inclusive Discord server management, and working to make AI more accessible

ACM Projects – Machine Learning Engineer September 2021 – January 2022

- Create model to convert human faces to Cat-Human hybrid using DCGAN, PatchGAN, CycleGAN, and StyleGAN
- Coordinate with team of 3 front- and back-end devs to implement model into user-friendly tool

Bittner Development Group

Bittner Development Group is an education technology company which creates interactives and products for clients like Norton, Barnes and Noble, SparkNotes, Thames and Hudson, and more.

Technologies Used: Web Development: React, SCSS, Node.js; DevOps: GitLab, Git, WSL; Scripting and Automation: TypeScript, JavaScript, Java; Development Standards: Web Content Accessibility Guidelines (WCAG) 2.1 AAA, Aria Authoring Practices Guide (APG), Norton Design System; Processes: Agile Methodology

Software Engineering Intern January 2021 – Present

- Develop component library and enterprise web application 'Interactive Builder' using React and SCSS
- Manage DevOps and CI/CD pipeline using GitLab
- Web development, QA, and devops of over 10 education interactive projects in React
- Manage and train two interns to complete projects using React and SCSS, GitLab, Git, and WSL
- Code projects in compliance of web content accessibility with WCAG
- Propose, lead, and develop internal and for-client automation projects using Node and native JavaScript

Media Group Intern November 2019 – January 2021

- Independent project involving automation of over 10 spreadsheet and data entry tasks using Java improving efficiency by over 90%
- Code questions into over 15 textbooks on Norton's online textbook platform, PCAT, using HTML

Personal Web Development Projects: https://github.com/arth-shukla

Technologies Used: Web Development: TypeScript, React, Rollup, Jest, Webpack, Storybook, SCSS; DevOps: Netlify, Git, GitHub Pages, GitHub Packages; Development Standards: WCAG 2.1 AAA, Aria Authoring Practices Guide (APG)

Independent Developer January 2022 – Present

- Create personal website using React Typescript, SCSS, and Material UI, accessible by WCAG 2.1 AA standard, and publish to custom domain using Netlify: https://arth.website
- Code mobile-compatible Dice Roller web app with React and SCSS, accessible by WCAG 2.1 AA standards and publish to GitHub Pages: https://arth-shukla.github.io/dice-roller
- Create Icon Library with React TypeScript and SCSS, publish to GitHub packages: https://github.com/arth-shukla/arth-components; code demos and documentation: https://arth-shukla.github.io/my-icons-documentation

SKILLS

- **Programming Languages** – Python, TypeScript, JavaScript, Node, React, Java, Ruby, Bash, SCSS, LESS, HTML5 and CSS3, C, C++, R
- **Packages and Libraries** – Pytorch, Keras, HuggingFace, PettingZoo, Rollup, Jest, Webpack, Storybook, PyGame, Jackson
- **Development Standards** – Open AI Gym, PettingZoo AEC Environment and Parallel Environment, Web Content Accessibility Guidelines (WCAG) 2.1 AAA, Aria Authoring Practices Guide (APG)
- **Programs and Software** – Netlify, Git, GitHub, GitLab, Conda, Maven, Visual Studio Code, Conda, Maven, WSL, Storybook, Android Studio, Matlab, Microsoft Office, Microsoft Excel, Microsoft Powerpoint
- **Languages** – Fluent in English and French, Spoken Hindi