Jeffrey Wu

Email me@wuthejeff.com
Website wuthejeff.com
Github user WuTheFWasThat
Google Scholar user x4JAvwMAAAAJ

Summary: Machine learning researcher and full-stack engineer interested in keeping the world safe as technologies such as machine learning become more powerful.

Work Experience

- **OpenAI** Research engineer — Aug 2018 – Present
  Focused on aligning models to human values. Researched language modeling (including training GPT-2, a state-of-the-art model in 2019) and reward modeling (training models that wrote summaries better than human-written ones) – more details in Research section. Also led applied efforts to integrate human-feedback techniques, training the instruct-series of models which were deployed in our API product with overwhelmingly positive response.

- **Google Research** Software engineer — Oct 2016 – Aug 2018
  Built general infrastructure (data pipelines, libraries, custom Tensorflow ops, a fun Lisp-like DSL) for supporting models for personalization from cross-product user history (basically learning giant embedding spaces for all users, Chrome pages, Youtube videos, etc). Experimented with RNN models to replace bag-of-words models, and helped launch news feed personalization experiments. In 20% time, studied properties of generalization error.

- **Terminal.com** Founding engineer — Jan 2013 – Oct 2016
  Building cloud-based container infrastructure, for scientific computing and online education. Helped design and implement many core systems across the stack and oversaw their security and scalability. Saw company grow from 2 to 12, and managed a small team of engineers. Interfaced with clients, including Crunchbase, Stanford University, Codecademy, and Udacity. Company was sold to Udacity.

- **Probabilistic Computing Project** Master's student — Nov 2011 – Jan 2013
  Implemented a probabilistic programming language. Explored a new Gibbs sampling algorithm to make inference more efficient in very general settings. Work presented [at NIPS 2012 probabilistic programing workshop]. [Source code] and [thesis].

Research

- **Large-scale language model training** I trained GPT-2 (blog) and the early iterations of GPT-3 (NEURIPS 2020 best paper award), scaling up OpenAI's largest models from 110M parameters to over 6B parameters. I also helped extensively with making efficient large-scale training infrastructure, implementing an evaluation suite, and building a web UI.

- **Reinforcement learning from human preferences** I then switched my efforts to learning from human feedback, and have been a primary contributor to research direction on the Alignment team. I joined midway and helped with analysis in Fine-Tuning Language Models

from Human Preferences (blog). After migrating from Tensorflow to Pytorch and making some methodological improvements I suggested, we worked on Learning to Summarize with Human Feedback (NEURIPS 2020) (blog), where we trained models that produced summaries better than human-written ones. I was involved in nearly every aspect of this paper, and I think our results were very strong and somewhat underappreciated. Most recently, I have been working on yet to be published results on book-length summarization.

- **Miscellaneous** I also helped out on some miscellaneous projects, such as iGPT (ICML 2020) and Scaling Laws for Neural Language Models.

Selected Side Projects

- **Vimflowy** Vim inspired outlining tool with many features. [Source] (Typescript) and [Demo].
- **Hanabi simulation** Game engine for simulating hanabi strategies, and state of the art bots. Cited in [DeepMind/Brain paper] and subsequently [interviewed for WSJ]. [Source] (Rust).
- **Send A Damned Message** A simple puzzle game written to learn ReasonML. (Play here!)
- **tapystry** A small library for handling side effects in python, inspired by redux-saga [source]
- **plotserver** A small app for plotting results of ML jobs [source]

Education

Massachusetts Institute of Technology — **Cumulative GPA**: 4.8/5
B.S. in Mathematics, **B.S.** in Computer Science — May 2012
M.Eng. in Computer Science — January 2013

Skills

- Deep learning (especially language modeling, RL, reward learning)
- Machine learning frameworks (Tensorflow, pytorch)
- Front end, e.g. React frameworks
- Mathematics (2006-2008 USAMO, 2010 Putnam top 200)
- CS theory (e.g. complexity theory, cryptography)
- Algorithms and distributed systems design
- Devops, e.g. AWS/GCP, linux, containers, Kubernetes
- Keeping an eye on the big picture
- Acting with integrity