Hashem Elezabi

(240) 708-3081 | hashem@stanford.edu | hashemelezabi.github.io | hashemelezabi | hashemelezabi

Education

Stanford University — *Stanford, CA*

M.S. in Computer Science (with Distinction in Research) | GPA: 3.95 — *Expected Jun 2024*

B.S. in Electrical Engineering (with Honors), Minor in Mathematics | GPA: 3.77 — *Jun 2022*

Coursework: [AI] *Machine Learning, Deep Learning for Computer Vision, NLP with Deep Learning, ML with Graphs, Deep Generative Models, Deep Reinforcement Learning.* [Math] *Formal and Mathematical Logic, Graph Theory, Applied Matrix Theory, Abstract Algebra, Real Analysis.* [Systems] *Parallel Computing, Operating Systems, Computer Architecture, Compilers, Data-Intensive Systems, Digital System Design, Database Systems.*

Honors & Awards

- **CS224N 2024 Outstanding Project Report |** Led project on improving LLM reasoning with a neurosymbolic approach, won among 509 students.
- **2022-23 Apple-Stanford Masters Scholarship |** 1 of 3 Stanford M.S. students in EE/CS chosen for this highly selective 1-year scholarship.
- **2022-23, 2021-22 Stanford School of Engineering Dean's Coterminal Fellowship |** This selective award covers a year of M.S. degree tuition.

Experience

Stanford Artificial Intelligence Lab *(ai.stanford.edu)* — *Stanford, CA*

Research Assistant, Stanford Vision and Learning Lab — *Jan 2024 - Present*

- Building a robust semantic evaluator of synthetic scenes using a GPT-4V agent. Fine-tuning a vision-language model to improve spatial reasoning.
- Worked on improving LLM-based question-answer generation for long-form video understanding via human feedback and prompt engineering.

Stanford Pervasive Parallelism Lab *(ppl.stanford.edu)* — *Stanford, CA*

Research Engineer — *Jan 2023 - Dec 2023*

- Trained graph neural networks (GNNs) to predict the TPU runtime of AI models as part of a NeurIPS 2023 competition for improving ML compilers.
- Fine-tuned two language models (DistilGPT and CodeGen-350M) with RLHF with different reward functions, towards improving LLM coding ability.

Apple Inc. — *Cupertino, CA*

Software Engineering Intern, SoC Performance — *Jun 2022 - Dec 2022*

- Developed new features in C++ performance models and ran simulations for improving the efficiency of Apple's iPhone and Mac chips.
- *Led new, cross-team effort* developing algorithms for efficiently analyzing hardware traces and bandwidth patterns to improve SoC performance.

NVIDIA Corporation — *Santa Clara, CA*

Software Engineering Intern, Deep Learning Library Performance — *Sep 2021 - Dec 2021*

- Contributed to internal APIs for new architectural features used for delivering efficient deep learning primitives as part of the Fast Kernels team.
- Integrated ~1000 new automated tests for NVIDIA's Hopper GPU architecture into Jenkins pipelines, and caught several software bugs.

Gridspace *(gridspace.com)* — *Los Angeles, CA*

Machine Learning Engineering Intern — *Jun 2020 - Sep 2020*

- Implemented and trained generative speech AI models in TensorFlow based on cutting-edge research for audio speech enhancement.
- Built a full AI pipeline, including complex data processing stages, and used it to enhance some of Gridspace's audio recordings.

Stanford Future Data Systems Lab — *Stanford, CA*

Undergraduate Researcher — *Jun 2017 - May 2018*

- Wrote optimized parallel code in Python and C++ for efficiently processing large (>1TB) seismic time series data for earthquake detection.
- Contributed to >100x speedup of algorithm, enabling discovery of >6K new earthquakes. Results published at VLDB, top database conference.

Selected Projects

Combining LLMs with a Z3 symbolic solver to improve their reasoning ability on AR-LSAT [paper, poster]

- Proposed and implemented a new framework, *Prototype-then-Refine*, that improves the ability of LLMs to generate correct logic programs using LLM-based *prototypers* and *refiners*. These logic programs are then fed to an external Z3 SAT solver. Our framework almost matches executable rate of generated logic programs with GPT-4 (32.61%) using the much cheaper GPT-3.5 (32.47%).

Vision-language model for converting diagrams to source code [paper, poster]

- Created a dataset of images of synthetic slides with diagrams and used it to fine-tune a DEtection TRansformer (DETR) object detection model for common diagram shapes. Achieved average precision of 89% on test data, significantly outperforming a baseline DETR.

Predicting prices of self-storage units using multi-modal data [paper, poster]

- Trained linear regression, neural network, and decision tree models on *geography embeddings* created by fusing tabular features (e.g. unit size) with unsupervised vector representations created by convolving random patches with satellite images. Achieved R^2 score of 0.75 on test data.

Parallel renderer in CUDA

- Wrote a parallel renderer in C and CUDA that draws overlapping colored circles efficiently. Designed algorithm that performs local computations in GPU shared memory, avoiding costly data transfer and dramatically improving performance. Solution beat reference time by up to >100x.

Teaching

- Teaching Assistant | EE 180: Computer Architecture (Jan - Mar 2024)
- Section Leader | Stanford CS 106A *Code in Place* (Apr - May 2020)

Skills

Languages Python, C/C++, Java, JavaScript, CUDA, SQL, Verilog, HTML, CSS, Matlab, LaTeX

Technologies PyTorch, NumPy, Git, Unix/Linux, TensorFlow, Apache Spark, HuggingFace, Pandas, Docker, MapReduce, ReactJS, Jira, Tableau