# Hashem Elezabi

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

## Education

**Stanford University** *Stanford, CA*

M.S. in Computer Science, AI Track | GPA: 3.91 *Expected Jun 2024*

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

**Coursework:** *Machine Learning, NLP with Deep Learning, Deep Learning for Computer Vision, Modern Algorithms, Data-Intensive Systems, ML with Graphs, Deep Generative Models, Parallel Computing, Operating Systems, Compilers, Computer Systems Architecture, Database Systems, Design & Analysis of Algorithms, Data Structures (Advanced), Mining Massive Datasets, Deep Reinforcement Learning, Decision Making Under Uncertainty.*

## Honors & Awards

- **2022-23 Apple-Stanford Masters Scholarship** | 1 of 3 Stanford M.S. students in EE/CS chosen for this highly selective 1-year scholarship.
- **2021-22 Stanford School of Engineering Dean's Coterminal Fellowship** | This selective award covers a year of M.S. degree tuition.

## Experience

**Stanford Pervasive Parallelism Lab** *Stanford, CA*

AI Research Engineer *Jan 2023 - Present*

- Trained graph neural networks (GNNs) to predict the TPU runtime of AI models given a computation graph, to improve ML compilers.
- **Fine-tuned large language models (LLMs)** pretrained for code generation to try to improve correctness of generated code via RLHF.

**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 a new, cross-team effort to apply advanced algorithms and data-driven processes** for extracting useful insights from hardware data.
- Designed and implemented algorithms for efficiently processing memory traces and analyzing 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 latest Hopper GPU** architecture into Jenkins pipelines, and caught several software bugs.
- Developed novel algorithm in Python for automatically testing various deep learning kernels to identify exact configurations that lead to 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.

## Projects

**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.

**Question-answering system with retrieval-augmented generation and the ChatGPT API**

- Built a program using Stanford's DSP library that retrieves relevant context passages from a ColBERTv2 Wikipedia index and prompts `gpt-3.5-turbo` to answer the question given the context passages. Improved F1 score from a baseline of $0.34$ to $0.51$ on a challenging dataset.

**Predicting prices of self-storage units using multi-modal data *(CS229 - Machine Learning)*** [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.

**MIPS processor and pipelined Sobel filter hardware accelerator *(EE180 - Digital Systems Architecture)***

- Implemented a 5-stage pipelined MIPS processor in Verilog, including hazard detection and data forwarding. Ran it on a real FPGA.
- For extra credit, implemented a pipelined datapath for a Sobel edge detector hardware accelerator in Verilog. Improved FPS by 96.6%.

**Parallel renderer in CUDA *(CS149 - Parallel Computing)***

- 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.

## Selected Publications

- *(VLDB '18)* **Locality-Sensitive Hashing for Earthquake Detection: A Case Study of Scaling Data-Driven Science.** (bit.ly/34fCIgT)
  Kexin Rong, Clara Yoon, Karianne Bergen, **Hashem Elezabi**, Peter Bailis, Philip Levis, Gregory Beroza.

## Skills

| | |
|---|---|
| **Languages** | Python, C/C++, Java, JavaScript, CUDA, SQL, Verilog, HTML, CSS, Matlab, LaTeX |
| **Technologies** | Git, Unix/Linux, PyTorch, TensorFlow, NumPy, Apache Spark, HuggingFace, Pandas, Docker, MapReduce, ReactJS, Jira, Tableau |
| **Areas** | Parallel computing, deep learning, data science, code optimization, computer vision, computer architecture, distributed systems, LLMs |