Eashan Gupta

University of Illinois, Urbana-Champaign
+1-217-979-0079 eashang2@illinois.edu linkedin.com/in/eagupta eash3010.github.io/

Education

University of Illinois Urbana-Champaign **2021-2023 (Expected by 05/23)**
Master of Science in Computer Science (with thesis) *GPA: 4.0/4.0*

Indian Institute of Technology Bombay **2016-2020**
Bachelor of Science in Computer Science with Honours *GPA: 9.13/10*

Technical Skills & Coursework

Languages: C++, C, Golang, Python, P4, Bash, Racket/Scheme, Haskell, Prolog, MIPS, SQL, Java
Technologies: Kubernetes, GitHub, Keras, Jupyter Notebooks, MATLAB, Simulink, Android Studio, Jenkins, PyTorch
Coursework: Advanced Operating Systems, Distributed Systems, Machine Learning for Signal Processing, Efficient & Predictive Vision, Knowledge-driven Natural Language Generation, Advances in Intelligent and Learning Agents, Advanced Machine Learning, Functional Programming Languages, Web Search & Information Retrieval, Digital Image Processing

Work Experience

Nutanix, Bengaluru | *Software Developer, Karbon/MSP team* **July 2020 - July 2021**
- Used **Kubernetes** to deploy microservices on a Hyper-converged Infrastructure using virtual machines
- Worked to support the Karbon platform on **VMware's hypervisor ESX** other than Nutanix's own hypervisor - AHV
- Added multiple features to the Karbon controller like migration to **CoreDNS** on k8s upgrade, network segmentation for efficient traffic handling, redacting logs, and tracking metrics using **Prometheus** and middlewares

Tower Research Capital, Gurgaon | *Core Engineering Intern* **May-July 2019**
- Automated the performance **testing platform** for the software processing the order book data broadcast
- Empirically investigated patterns in performance on using **cache allocation technology** with different configurations

Research Experience

Improving bounds of Policy Iteration Algorithm **February-August 2020**
Guide: Prof. Shivaram Kalyanakrishnan | Research Project *IIT Bombay*
- Proved exponentially better upper bounds for the number of steps taken by Policy Iteration Algorithm (**PI**) to determine the optimal policy in deterministic Markov Decision Processes (**DMDPs**) by bounding number of path-cycles in a digraph
- Conducted various empirical experiments on lower order AUSOs to observe the family of randomized PI

Towards validation of RTL passes of the GCC compiler **January-June 2020**
Guides: Prof. Amitabha Sanyal & Prof. Supratik Chakraborty | Bachelor's Thesis *IIT Bombay*
- Analysed the various Register Transfer Language (**RTL**) optimization passes in **GCC-4.7.2** and implemented a **block-by-block** validation technique to validate program transformations done by the passes
- Realized obligations based on the return values, heap memory and function calls of programs in the **Z3 Theorem Prover** tool to prove semantic equivalence between different control flow graphs (**CFGs**)

Optimized DL GPU Task Scheduling for NVIDIA Jetson TX2 **[GitHub] | Aug-Dec 2021**
Guide: Prof. Tianyin Xu *University of Illinois Urbana-Champaign*
- Implemented GPU task scheduling algorithms for deep learning inference models based on greedy longest chains and load balancing in **PyTorch**; Improved performance of Nimble by upto 16% on TX2, a popular embedded AI systems hardware

Key Projects

- **Self-Supervised Embedding-based Speech Emotion Recognition**: Trained a **Siamese NN** to distinguish emotions between 2 input speech samples with test accuracy 82% on the **CREMA-D** speech dataset, with upto 54% accuracy for unseen classes. Used the model as an embedding to learn emotion classifier with 81% accuracy
- **Abstract Interpretation and Program Verification**: Used domain specific techniques and fixed point analysis to implement **congruence** and **array** abstract domains for integration into the **CAnalyzer** tool
- **Reduction in Games played on recursion schemes**: Intern with Prof. Roland Meyer at TU Braunschweig, Germany; Worked on the reduction of parity games to safety games played on higher order recursion schemes (**HORS**), using similar results on reduction in games played on collapsible pushdown automata (**CPDA**)
- **Strong Password Generation**: Devised methods to evaluate a password based on the metrics of **guessability** and **memorability** and used them to compare the generative models developed
- **Monadic Parser**: Modernised the parser implementation for core language in **Haskell** using Monads
- **Handwriting synthesis**: Devised an algorithm to train an LSTM and an encoder-decoder model in an **adversarial** fashion; Used it to string letters smoothly to form complete handwritten words