Saurabh Raje

(+1) 801-706-7032 | saurabh.raje@utah.edu | smr97.github.io | smr97 | Saurabh Raje | saurabhmraje

Research Statement

I am a second year PhD student studying computer science. My research is in the domain of high performance computing (HPC). Currently, I am developing compiler methods to optimise chains of sparse tensor contractions. Prior to this, I have worked on scaling up sparse tensor decomposition algorithms. Some of my previous research has been on accelerating training and inference of deep neural networks - specifically in NLP and graph-based learning. Other than hand-tuning kernels, I have also worked on domain specific languages (DSL) to generate fast code for CPUs and GPUs.

Education

University of Utah — *Salt Lake City, USA*
PhD in Computer Science and Engineering — *Aug. 2021 - Present*
Advised by Prof. Saday Sadayappan

Birla Institute of Technology and Science, Pilani — *Pilani, India*
Bachelor of Engineering — *Aug. 2015 - Dec. 2018*
Major: Computer Science

Publications and Patents

[1] V. T. Chakaravarthy, A. R. Choudhury, S. Goyal, S. M. Raje, Y. Sabharwal, and A. Verma. Input ordering neural network decomposition, Mar. 24 2022. *US Patent* App. 17/026,589.

[2] V. T. Chakaravarthy, S. S. Pandian, **Raje, Saurabh**, and Y. Sabharwal. On optimizing distributed non-negative tucker decomposition. In *Proceedings of the ACM International Conference on Supercomputing (**ICS**)*, pages 238–249, 2019.

[3] V. T. Chakaravarthy, S. S. Pandian, **Raje, Saurabh**, Y. Sabharwal, T. Suzumura, and S. Ubaru. Efficient scaling of dynamic graph neural networks. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, **SC** '21, New York, NY, USA, 2021. Association for Computing Machinery.

[4] S. Goyal, A. R. Choudhury, **Raje, Saurabh**, V. Chakaravarthy, Y. Sabharwal, and A. Verma. PoWER-BERT: Accelerating BERT inference via progressive word-vector elimination. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning (**ICML**)*, volume 119 of *Proceedings of Machine Learning Research*, pages 3690–3699, Virtual, 13–18 Jul 2020. PMLR.

[5] S. Islam, S. Balasubramaniam, P. Goyal, A. Sultana, L. Bhutani, **Raje, Saurabh**, and N. Goyal. A rapid prototyping approach for high performance density-based clustering. In *2019 IEEE International Conference on Data Science and Advanced Analytics (**DSAA**)*, pages 260–269. IEEE, 2019.

[6] A. Kannan, A. Roy Choudhury, V. Saxena, **Raje, Saurabh**, P. Ram, A. Verma, and Y. Sabharwal. Hyperaspo: Fusion of model and hyper parameter optimization for multi-objective machine learning. In *2021 IEEE International Conference on Big Data (**Big Data**)*, pages 790–800, 2021.

[7] S. E. Kurt, **Raje, Saurabh**, A. Sukumaran-Rajam, and P. Sadayappan. Sparsity-aware tensor decomposition. In *2022 IEEE International Parallel and Distributed Processing Symposium (**IPDPS**)*, pages 952–962, 2022.

[8] S. M. Raje, S. Goyal, A. R. Choudhury, Y. Sabharwal, and A. Verma. Accelerating inference of neural network models via dynamic early exits, Nov. 10 2022. *US Patent* App. 17/307,501.

[9] V. Saxena, A. Kannan, S. M. Raje, P. Ram, Y. Sabharwal, and A. Verma. Multi-objective automated machine learning, June 9 2022. *US Patent* App. 17/115,673.

[10] **Raje, Saurabh**, A. Goel, S. Sharma, K. Aggarwal, D. Mantri, and T. Kumar. Development of on board computer for a nanosatellite. *68th International Astronautical Congress (IAC)*, 2017.

[11] **Raje, Saurabh**, S. Vaderia, N. Wilson, and R. Panigrahi. Decentralised firewall for malware detection. In *2017 International Conference on Advances in Computing, Communication and Control (ICAC3)*, pages 1–5. IEEE, 2017.

[12] Y. Xu, **Raje, Saurabh**, A. Rountev, G. Sabin, A. Sukumaran-Rajam, and P. Sadayappan. Training of deep learning pipelines on memory-constrained gpus via segmented fused-tiled execution. In *Proceedings of the 31st ACM SIGPLAN International Conference on Compiler Construction*, **CC** 2022, page 104–116, New York, NY, USA, 2022. Association for Computing Machinery.

Work Experience

University of Utah
Salt Lake City, Utah

Doctoral Researcher

Accelerating Sparse Linear Algebra

August 2019 - Present

- Currently working on new representations for sparse tensors with domain specific patterns.
- In collaboration with Pacific Northwest National labs, this research aims to accelerate quantum chemistry simulations.
- Co-developed a novel implementation for sparse-tensor decomposition (**SpTL**).
- **SpTL** reduces data movement and load imbalance to beat the state-of-the-art run-time.
- Co-developed a system to train convolutional neural networks (CNN)s on large images (20,000 x 20,000).
- This effectively tiles the dataflow through CNNs to enable processing of massive images on a single GPU system.

IBM Research
Delhi, India

Research Engineer

Accelerating AI

August 2019 - August 2021

- Worked with the model compression team to make AI faster.
- Designed *PowerBERT*, a new model that is up to **4.5x faster** than **BERT** for inference.
- This work was published in **ICML'20**, and was integrated into IBM OneNLP product stack.
- Implemented a new method to train massive Graph Neural Networks faster using supercomputers (published at **SC'21**)
- Implemented novel representations for sparse tensors. This was used to accelerate the tucker decomposition algorithm.
- Co-invented 4 **patents** on model compression techniques and multiobjective optimisation.

ETH Zurich
Zurich, Switzerland

Scientific Assistant

Compilers for Deep Learning

March 2019 - August 2019

- Accelerated the training of Deep Neural Networks using the **DACE** language developed in-house.
- DACE is a domain specific language for HPC workloads that uses a novel Stateful Dataflow Graph (SDFG) based Intermediate Representation.
- Wrote a Tensorflow frontend for DACE that parses a TF computation graph to build a DACE SDFG.
- Added a pattern based compiler transformation on the IR to reduce GPU kernel calls and repetitive memory access.
- Achieved at-par performance for ResNet-50 in comparison to Tensorflow and CuDNN.

INRIA
Grenoble, France

Bachelor Thesis

Middleware for Parallel Programming

September 2018 - February 2019

- Developed **Kvik**: a task based middleware in the **Rust** language.
- **Kvik** makes sequential code run in parallel without significant changes, by creating independent tasks.
- In particular, it provides tunable task splitting strategies that can be composed with each other.
- Wrote the fastest parallel merge sort using **Kvik** (2.5x faster than Intel TBB for 50 threads).

Honors & Awards

2021 **Winner,** Patent Plateau Award - IBM India Research Lab

2021 **Winner,** Outstanding Technical Achievement Award - IBM India Research Lab

2020 **Winner,** Distinguished Paper Award - IBM India Research Lab

2017 **Winner,** Mercedes Benz Hack.Banglore 2018

Presentations

Mobile World Congress 2018
Barcelona, Spain

Invited by Daimler AG to present our winning hackathon prototype

February 2018

Skills

Languages Rust, Python, C, C++, Java

Frameworks PyTorch, Tensorflow, Caffe, CuDNN, Git

HPC Libraries openMPI, openMP, Intel TBB