Suhas Jayaram Subramanya

Research Fellow
Microsoft Research India
Advisors: Dr. Harsha Simhadri, Dr. Praneeth Netrapalli

December, 2018
t-sujs@microsoft.com | j.s.suhas@live.in
Webpage : suhasjs.github.io
Github : www.github.com/suhasjs

Education

Indian Institute of Technology Madras, Chennai, India
Bachelor of Technology (B.Tech) in Computer Science and Engineering, *Jul '13 - Jul '17*
GPA: 9.32/10, Department Rank - 3 (Top 10%)

Research Interests

Systems for Machine Learning, Large-Scale Machine Learning, Distributed Systems, Reinforcement Learning

Publications

BLAS-on-flash: An Efficient Alternative for Large Scale ML Training and Inference?
Suhas Jayaram Subramanya, Harsha Simhadri, Srajan Garg, Anil Kag, Venkatesh Balasubramanian
16th USENIX Symposium on Networked Systems Design and Implementation (NSDI), Boston, 2019.

BLAS-on-flash: an alternative for training large ML models?
Suhas Jayaram Subramanya, Srajan Garg, Harsha Simhadri
SysML Conference, Stanford, 2018.

Exploration for Multi-task Reinforcement Learning with Deep Generative Models
Sai Praveen Bangaru, **Suhas Jayaram Subramanya**, Balaraman Ravindran
NIPS Deep Reinforcement Learning Workshop, Barcelona, 2016.

Work Experience

Microsoft Research India
Research Fellow — *Jul '17 - Jul '19,* ***Bangalore***

Working on cost-effective scalable machine learning impacting production pipelines for topic-modeling, extreme multi-label learning, deep relevance model training, and approximate nearest neighbor serving.

Google India
Software Engineering Intern, Strategic Technologies team — *May - Jul '16,* ***Bangalore***

Developed annotation metrics and production pipelines to understand efficacy of personalization re-rankers.

Hyperverge Technologies Inc.
Algorithms Engineer — *Jan - Dec '15,* ***Chennai***

Developed tools for album clustering and captioning using scene, timeline and geographic information obtained from a collection of photos and state-of-the-art scene labeling algorithms.

Teaching Experience

Introduction to Machine Learning
Instructor: Prof. Balaraman Ravindran — *Jan - Apr '17,* ***IIT Madras***

Teaching Assistant on a MOOC hosted on NPTEL with over 6000 registered students. Course contents now archived to allow others to take the course at their own pace.

Research Projects

BLAS-on-flash: An Efficient Alternative for Large Scale ML Training and Inference?

Advisor: Dr. Harsha Simhadri *Aug '17 - Jun '18,* ***Microsoft Research***

Conceptualized, designed and developed a framework to express matrix-based algorithms as dynamic computation graphs with nodes performing compute on a small subset of input data. Developed a high-performance runtime to execute these graphs on flash-resident data within a memory budget. Implementations of algorithms with a wide range of compute-communication ratios achieve performance parity with their in-memory variants at a substantially lower memory footprint, making 10x scalability in input sizes feasible.

Navigation for Muti-task Reinforcement Learning

Advisor: Prof. Balaraman Ravindran *Aug - Nov '16,* ***IIT Madras***

This project explores the problem of navigation in a distribution of maze-like environments with POMDP-like behaviours. We use a Variational Autoencoder in conjunction with a Gaussian Restricted Boltzmann Machine to model the agent's belief over the environment distribution and incentivise the agent to reduce uncertainty in the belief. Rollouts are used for exploration with Q-learning as the core learning algorithm.

Fast, Production-grade k-ANN Systems on Flash Storage

Advisors: Dr. Harsha Simhadri, Dr. Ravishankar Krishnaswamy, Dr. Amit Deshpande *Ongoing,* ***Microsoft Research***

Exploring approaches to develop a production-quality k-ANN serving system using SSDs to index and serve web-scale datasets (100B+ vectors) with a target of 10B vectors per node.

Parallelization of DNN Training on Web-scale Corpora

Advisors: Dr. Harsha Simhadri, Dr. Praneeth Netrapalli *Ongoing,* ***Microsoft Research***

Exploring utility of ensembles of DNNs in faster training of deep relevance models on web-scale datasets (2B+ train points). Efforts are focused on efficient learning of ensembles and an efficient distillation procedure to reduce end-to-end train time.

Continuous Control for Simulated Creatures using Hierarchy of Policies

Advisor: Prof. Balaraman Ravindran *Jan - Jun '17,* ***IIT Madras***

Explored a hierarchical approach to continuous control inspired by the Encapsulation-Syllabus-Pandemonium (ESP). With influences from Feudal Reinforcement Learning, DDPG, and A3C, policies are organized in a hierarchy to learn increasingly abstract *sub-routines* through a programmer-defined curriculum.

Learning Input Conditional Language Models for Natural Language Generation

Advisor: Prof. Sutanu Chakraborti *Aug - Nov '16,* ***IIT Madras***

This work explores neural network architectures for learning surface realization, sentence planning, and content determiniation in an end-to-end manner from raw-data to full textual descriptions. Augmenting language-modeling LSTMs with an attention-layer over word-forms of inputs in the Prodigy-METEO dataset allows the model to *copy-paste* tokens from inputs in the final output.

Open Source Contributions

BLAS-on-flash

Microsoft Research, ≈10000 LOC [Github]

Implemented the BLAS-on-flash framework in C++ with template-support. A high-performance multi-threaded runtime implements a custom caching layer and uses Linux kernel asynchonous I/O support for block I/O on SSDs with callbacks. Matrix-multiplication kernels like `gemm` (dense-dense) and `csrmm`(sparse-dense), and utility kernels like `csrcsc`(sparse-transpose), `sort`(Parallel Sample Sort), and `map-reduce` are implemented in the framework.

Importance Sampling for Learning Edge Topics (ISLE)

Microsoft Research, 2000+ LOC [Github]

Implemented a symmetric eigensolver using the Block Krylov-Schur algorithm and ported memory-limited sections of ISLE to use the BLAS-on-flash framework. This new implementation is capable of training >10x larger models on multi-core workstation-class machines in the same memory envelope.