

EDUCATION

- **Stony Brook University** New York, USA
Degree: Master of Science in Computer Science *Jan-2019 – May-2020*
- **Maulana Azad National Institute Of Technology** Bhopal, India
Degree: Bachelor in Technology with major in Computer Science & Engineering *July-2010 – April-2014*

PROGRAMMING SKILLS

- **Programming Languages:** C++, C, Java, Golang, JavaScript, SQL
- **Technologies:** Apache Storm, Apache Spark, Redis, REST, Apache Kafka, AWS, Git/Perforce, MATLAB
- **Relevant Coursework:** Distributed Systems, Operating Systems, Analysis of algorithms, Visualization

EXPERIENCE

- **Microsoft** May'2022 – till
Software Engineer-2 *Mountain View, CA*
 - **Snap Service**
 - * Implemented new lightweight service layer to provide native renderer workers to download from Storage service to Azure blob cache to improve streaming latency by 25%.
 - * Implemented testing framework to improve service reliability score by 85% and improved deployment frequency by 10%.
- **VMware Inc.** July'2020 – May'2022
Member of Technical Staff-III *Palo Alto, CA*
 - **Virtual distributed file system, Storage layer**
 - * Conceptualized and implemented distributed **F3CK** tool for analyzing and repairing storage metadata. This tool is leveraged to analyze the state of metadata storage on disk after crash recovery by forming analysis matrices and cross consistency patterns between multiple metadata data structures like B+ trees, bitmap, segment usage table
 - * End-to-end implementation of a cloud-native microservice to validate the integrity of the file system **k-v store** (such as **B+ trees**). The service can be scheduled or run on-demand to check the health of the storage system k-v store. In addition, the service is also used for other storage features, such as snapshots, un-map, segment cleaning to verify consistency and integrity of the metadata.
 - * Implemented aggregated snapshot capacity from scratch to support statistics collection with p99 latency in range of 30 ms.
- **Distributed Systems Lab - Stony Book University** June'2019 – May'2020
Graduate Research Assistant, Prof. Shuai Mu *Stony Brook, NY*
 - **Distributed Multi-core Transactional Database Engine**
 - * Implemented **asynchronous replication** with multi-process **paxos** to achieve minimal loss in throughput
 - * Designed multi-core log truncation mechanism to support check-pointing for the transaction recovery protocol, which allowed read only transactions to scale 10M ops/sec.
 - * Implemented replay protocol to guarantee **serializability** and consistency check pipeline for in-memory streams & disk logs.
 - * Implemented verification pipeline for generating CPU/Heap/Memory throttling graph using **gperftools** & **mutrace** tool.
 - * Implemented fast header-only/compiled, C++ logging library facilitating aligned memory allocators with features like Rotating log and auto-flush to make transparent transaction serialization
- **Talentica Software** April'2016 – Jan'2019
Senior Software Engineer *Pune, India*
 - **Estimation & Prediction algorithms for Wireless Systems**

- * Designed & Implemented a stream data pipeline for machine learning inferences & predictions using **Apache Storm, Apache Kafka, AWS, Cassandra**.
- * Improved network traffic prediction accuracy to **80%** using important native SLA metrics, RF Frequencies & feedback loop from auxiliary methods.
- * Implemented feedback pipeline to improve accuracy of indoor location prediction algorithm by 95% for live BLE assets.
- * Developed a multi-modal Machine Learning model for network traffic classification for Audio and Video streaming using Ensemble of regression and auto-encoders.
- * Added an ad-hoc client to Storm topology to support collection of Real-time data to Disk & Batch pipeline using Java, Apache Spark.
- * Developed a live BLE asset view portal to support data collection team, adding 70% more correctness to site calibration using JavaScript, Apache Kafka, Python.
- o **Machine Learning Algorithm object storage service**
 - * Designed continuous delivery pipeline for Machine Learning models using gRPC, protobuf, Redis, RabbitMQ, AWS S3 improving deployment frequency by 40%.
 - * Wrote polyglot client suite in Java, python, Go, C++ to facilitate generic object serialization which lead to improved continuous deployment frequency.
- o **File Sync Application**
 - * Implemented delta file sync Service in C++ using native git-tree diff algorithm which reduce load in sync server by 400%.
 - * Developed an update framework to support release notification & binary upgrade for multi-talent architecture using Java, C++, Python.
 - * Designed rate limiter service to prevent throttling of indexing layer & sync layer, improving average P99 sync latency by 30%.
- o **Financial Document Search Engine:**
 - * Improved search relevancy by 45%. Improved Online Ontology enhancer by 30% via optimizing Neo4j query consumption, NLP Pipeline for Document Clustering, Keyword Extraction, Text Classification.
- **MediaTek** Aug'2014 – April'2016
Software Engineer *Noida, India*
 - o As a **feature owner** of Audio module, implemented design and bug-fixes for new product release. Improved playlist design to reduce playlist update overload by 25%, Integrated BT Stack in MMI Layer
 - o Wrote Software Layer code for Stress Test based Combo GPS-WiFi-BT Tool in VC++

PROJECTS

- **Raft - A distributed fault tolerant in-memory key value database**
 - Implemented Sharded and replicated fault-tolerance key-value store based on raft protocol from scratch in Go.
 - Tweaked leader's election to ensure progress in case of repetitive failure in re-election step
- **Map-Reduce library** Implemented distributed map reduce library and worker failures from paper.
- **Backup File System in Linux Kernel** Implemented **stackable virtual file system** with Custom Visibility, Retention and Version Management Policy to support backup to flat files under guidance of **Prof. Erez Zadok** using C and **Linux VFS Layer**.
- **Encryption based System tool for Linux Kernel** Implemented system call for encrypt/decrypt file using AES algorithm provided by **Linux kernel** crypto API under guidance of **Prof. Erez Zadok** in C.

PUBLICATIONS

- **Learning to Fingerprint the Latent Structure in Question Articulation:** In this paper, we show that the latent patterns of questions can be represented as a system that maximizes a cost function related to the underlying objective and can be approximated to building a memory of patterns represented as a trained neural auto-encoder. [[publication](#)]