Narayan Acharya

narayan.acharya@stonybrook.edu

/narayanacharya6 /in/narayan-acharya /narayanacharya.com 6313076395

Education

Stony Brook University *Stony Brook, NY*

Master of Science in Computer Science, Graduating Dec 2020, GPA: 3.85/4.0 *Aug 2019 – **Present***

- Data Science, Natural Language Processing, Computer Vision, Big Data, Probability & Statistics.
- Teaching Assistant: Programming Abstractions (Fall '19), Software Engineering (Spring '20, Fall '21).

University of Mumbai *Mumbai, India*

Bachelor of Engineering in Information Technology; GPA: 3.9/4.0 *Aug 2011 – May 2015*

- Data Structures & Algorithms, Discrete Maths, Software Engineering, Data Mining, Operating Systems

Projects

- **Video Action Classification**: Compared LSTM v/s SVM for action classification task on the UCF101 dataset. Leveraged Transfer Learning to compute features for 60000 video frames with limited compute resources.
- **3D Pose Estimation**: Estimated 3D pose co-ordinates of humans by regressing over their 2D co-ordinates only using Deep Neural Networks. Experimented with Kalman Filters for correcting noisy measurements due to faulty sensors.
- **Natural Scene Classification**: Classified scenes using Transfer Learning & Convolutional Neural Networks (AlexNet, VGG) with 89% accuracy. Used features from pre-trained networks to learn a Linear SVC for the same classification task.
- **Chess Rating Prediction**: Evaluated Random Forests, Gradient Boosting over novel features extracted from moves in 100k chess games to predict Elo ratings. Feature extraction run on distributed nodes using OpenMP for 15x speed.
- **Understanding Infant Mortality**: Applied Linear Regression to suggest priority actions to reduce Infant Mortality Rate based on 16.8GB of health & social records of 3M women using Dask & Apache Spark for parallel computation.
- **Comment Toxicity Detection**: Achieved an AUC score of 0.98+ with creative pre-processing techniques coupled with Bi-GRU & BERT for multi-class classification of toxicity levels in Wikipedia comments.

Experience

PlayStation, *Software Engineer Intern* *May 2020 – Aug 2020*

- Built Deep Learning pipeline to improve perceived video quality by enhancing regions with text. Reduced inference time by ∼75% for text detection under challenging constraints without reducing precision & recall.
- Evaluated multiple lightweight backbone networks, ResNet, PVANet, MobileNetV2, and techniques like quanitzation, post-processing optimization to further improve inference speed by 2x-4x.
- Developed a web dashboard using Flask, HTML & JavaScript, to help identify incorrect annotations which helped increase precision & recall of text detection by 1-2%.

JP Morgan Chase & Co., *Associate Software Engineer* *May 2017 – Jul 2019*

- Redesigned server-side services to support Angular client & streaming of real-time data using WebSockets. Improved app performance with ∼70% reduction in payload & boosted reliability using micro-services architecture with Spring Cloud.
- Developed non-intrusive ways to gather, visualize and analyze latency metrics using Elasticsearch, Logstash & Kibana, in the order of 100ms, across micro-services to identify bottlenecks in high throughput services.
- Optimized data collection mechanisms using REST APIs to track and compare client portfolio pre & post trade to generate risk reports on demand, increasing granularity in understanding firm's risk exposure.

LiveFiesta, *Lead Android Developer* *Jun 2016 – Jan 2017*

- Led a team of 4 in the design & development of Android application (rating 4.5+) for users to book tickets to events. Leveraged MVP architecture, Dependency Injection & TDD using RxJava, Dagger & Espresso.
- Reduced customer entry time to events by 50%, shortened queue lengths & cut losses due to fake ticket duplication & untracked re-entrants by developing a utility Android application for fast ticket verification and redemption.

TechGenium, *Software Developer & Partner* *Jun 2015 – May 2016*

Technical Skills

- **Languages**: Proficient in Python & Java, Familiar with C & C++, JavaScript, SQL
- **Frameworks & Libraries**: TensorFlow, PyTorch, OpenCV, scikit-learn, NumPy, pandas, Spring, Android, Docker
- **Build Tools**: Git, Gradle, Maven, Jenkins, Bash, Linux, Google Cloud Platform(GCP)