

# VIKRAM J. SHENOY

shenoy.vi@husky.neu.edu | +1 (857) 207-6100 | www.linkedin.com/in/vikramshenoy97/ | vikramshenoy97.github.io | Boston, MA, USA

## EDUCATION

**Northeastern University**, Boston, MA, USA

**Sept 2019 – May 2021**

- Master of Science in Computer Science.

**Mumbai University**, Mumbai, India

**July 2014 – May 2018**

- Bachelor of Engineering in Computer Engineering: **GPA: 3.63/4.0.**

## WORK EXPERIENCE

**Technical Consultant, Annadhan Welfare Organization (Mumbai, IN)**

**Jan 2019 – July 2019**

- Operated as a pro bono technical consultant for design and development of Annadhan's mobile application in collaboration with J.P. Morgan Chase & Co.
- Created a workflow for application's functionalities in accordance to organization's day-to-day operations.
- Designed a sleek, consistent, and user-friendly interface for mobile application targeting all age groups.

**Machine Learning Research Intern, University of Groningen (Groningen, NL)**

**Aug 2018 – Oct 2018**

- Performed an extensive analysis of proposed feature selection algorithm as compared to existing feature selection methods such as Fisher Score, Generalized Matrix Learning Vector Quantization (GMLVQ), ReliefF and Boruta.
- Devised a new weighting scheme which considerably improved algorithm's performance and efficiency for larger datasets.
- Gained extensive knowledge about ensemble methods such as Random Forests, AdaBoost, and XGBoost.

**Software Engineering Intern, Vroom Cars (Irvine, CA, USA)**

**Feb 2017 – Aug 2017**

- Researched On-board Diagnostics Parameter IDs (OBD – II PIDS) - codes used to request data from vehicles.
- Implemented an algorithm to extract and transform raw data received through a mobile application into structured format.
- Created automated software to convert structured data from a local database into multiple user-friendly graphs.

## PROJECTS

**Understanding Capsule Networks (PyTorch and Google Colab)**

**Aug 2019**

- Built a Capsule Network and implemented dynamic routing algorithm that functions as a forward pass for entire network.
- Achieved a final accuracy of 99.91% on training set and an accuracy of 98.80% on test set of MNIST dataset.
- Gained a deeper understanding of state of features captured by 16-dimensional vector of Digit Capsule Layer.

**Music Recommendation using Deep Learning (Keras and Google Colab)**

**May 2019**

- Preprocessed data by producing mel-spectrograms for 8000 audio files, each of 30 seconds, from Free Music Archive dataset.
- Designed a CNN on Google Colab for classifying 60,000 image slices of these mel-spectrograms into 8 different genres.
- Predicted latent feature vectors using final network and established strong cosine similarity score between one song (anchor) and other similar songs in test set.

**Digit Generation using Wasserstein Generative Adversarial Networks (Keras and Google Colab)**

**Apr 2019**

- Trained a Wasserstein GAN on MNIST dataset utilizing an estimate of Wasserstein metric as a cost function for 90,000 epochs.
- Generated images of digits by randomly sampling through a noise distribution and passing these samples through generator network.

**Twitter Sentiment Analysis using Recurrent Neural Networks (Keras and Google Colab)**

**Mar 2019**

- Preprocessed over 1.6 million positive and negative tweets from Stanford's Sentiment140 dataset.
- Constructed a Recurrent Neural Network with Long Short-Term Memory units to analyze sentiment of these tweets.
- Achieved an accuracy of 84.57 % on test set and employed final network to yield a degree of sentiment on user entered text.

**Neural Style Transfer with Convolutional Neural Networks (Keras and TensorFlow)**

**Feb 2019**

- Transferred artistic style of one image onto another image utilizing a pre-trained VGG19 network with Imagenet weights.
- Generated final image by selecting intermediate layers of network and reducing overall loss (style loss and content loss).

**Games using Artificial Intelligence, Undergraduate Final Year Project**

**May 2018**

- Programmed an AI for chess and a famous tile puzzle game, 2048, leveraging fundamental aspects of Game Theory.
- AI for Chess is based on Minimax algorithm with alpha-beta pruning and AI for 2048 employs Expectimax algorithm.

## TECHNICAL SKILLS

- **Programming Languages:** Python (Expert), Java (Proficient), C (Familiar).
- **Frameworks and tools:** Keras, PyTorch, TensorFlow, Plotly, Scikit-Learn, Matplotlib, Pandas, Google Colab, Android Studio.
- **Web Development and Data-oriented Languages:** HTML, CSS, PHP, JavaScript, Ajax, SQL.

## ACHIEVEMENTS

**TSEC Leadership Award**

**Mar 2018**

- Awarded the TSEC Leadership Award for co-founding and heading the Rotaract Club of TSEC in its inaugural year.