# Shoumik Majumdar

• Website: Shoumik Majumdar • Github: github/ShoumikMajumdar
• LinkedIn: Shoumik Majumdar • Email: shoumikmajumdar@gmail.com • +1 (857) 588-2263

**EDUCATION**

**Boston University, Boston**     Sep 2019 - Jan 2021
- MS, Computer Science     **GPA: 3.68**
- **Selected Coursework**: Machine Learning, Computer Vision, Robot Learning and Vision for Navigation, Artificial Intelligence, Object Oriented Design, Graduate Algorithms.

**University of Mumbai, India**     Aug 2014 - May 2018
- Bachelor of Engineering, Computer Science

**SKILLS**

- **Languages**: Python, C, Java, R, Shell Script, SQL.
- **Databases**: mySQL, HBase.
- **Technologies/Platforms**: Linux, Bash, PyTorch, TensorFlow, Keras, Pandas, Numpy, OpenCV, Scikit-learn, Flask, Git, Docker, Selenium, Beautiful soup.

**WORK EXPERIENCE**

**Boston University**: Research Associate (Machine Learning), Boston     Sep 2019 - Present
- Collected, filtered and annotated the **1st human action recognition video dataset** for purpose of **domain generalization** with Prof Sarah Bargal using **Selenium**.
- Trained models on a cluster of GPUs to classify videos using the Inception I3D architecture to align shifts across both spatial and temporal domains.
- Extended **Adversarial Feature Augmentations** and **GradCAM saliency** mappings to incorporate model explainability for videos. Stack: **Python, Pytorch**.
- Currently working towards a submission to a top tier computer vision conference.

**Affectiva**: EMpath Mentor, Boston     Jun 2020 - Aug 2020
- Mentored college students at Affectiva's EMPath Intern Program in the development of an emotionally enabled IOT product.
- Developed and prototyped a product that provides emotional support to mental health patients by leveraging Facial Expressions and Language Models.

**RECENT PROJECTS**

**Recommendation Systems for Movies**     Dec 2020 - Feb 2021
- Developed a robust recommendation ranking engine for movies using Content based and Collaborative filtering. Stack: **Python, TFIDF, Scikit-learn, Pandas, Numpy**.
- Engineered a way to tackle the cold start problem by suggesting most popular items among all users.
- Used an item-based memory approach and the **SVD matrix factorization** method for **collaborative filtering**. Leveraged features such as genres, cast, crew and keywords from the plot for **content-based recommendations**.
- Deployed the recommendation system as **REST API** using **Flask, CSS, HTML**.

**Reinforcement Learning agent for self-driving car**     Oct 2020 - Dec 2020
- Trained a self-driving agent using **DeepQ** and **Double DeepQ** reinforcement learning algorithms on OpenAI Gym's Car Racing environment using **PyTorch**.
- Implemented the **Epsilon-Greedy** algorithm to address the multi armed bandit problem of the exploration-exploitation tradeoff.
- Resulting agent led to an average score of **631** out of a maximum possible score of **1000** over 10 trials on random seeds. The agent was the **highest scoring agent** among 10 other models.

**Emulating Human Perception in Autonomous Agents**     Sep 2020 - Dec 2020
- Reproduced a human perception strategy using a **model-based** approach inspired by the perception pipeline presented in the world models paper.
- Developed a **Variational AutoEncoder (VAE)** to generate a representation of agent's environment. Stack: **Python, PyTorch**.
- Evaluated efficiency of agent on an augmented version of OpenAI Gym's Car Racing environment.
- The perception strategy is found to be effective for larger sized objects but fails to reconstruct smaller sized objects, hence concluded non robust.

**Chest X-Ray classification for COVID19 detection**     Mar 2020 - Apr 2020
- Designed an image classifier using **CNNs** to perform binary and multi-class classification to distinguish between COVID19 patients and Pneumonia patients using **Keras and TensorFlow**.
- Reviewed and compared results to ResNet50, VGG19, Inception ResnetV2 and Xception model architectures pre-trained on the ImageNet Dataset.
- Model engineered **outperformed 80%** of compared pre-trained models providing reasonable evidence to determine best architecture to solve given problem.