Karkala Shashank Hegde

hegde95.github.io | LinkedIn | Google Scholar

Email : khegde@usc.edu
Mobile : +1-(626)-620-2976

Education

- **University of Southern California - PhD** — Los Angeles, USA
  Electrical and Computer Engineering — *GPA: 3.94/4* — *2021 – Present*
  Awarded the USC Annenberg Fellowship. AI Researcher at the Robotic Embedded Systems Laboratory, advised by Dr. Gaurav Sukhatme.

- **University of Southern California - Master of Science** — Los Angeles, USA
  Electrical and Computer Engineering — *GPA: 3.94/4* — *2019 – 2021*
  Graduated with Honors (top 5% of students)

- **National Institute of Technology Karnataka - Bachelor of Technology** — Surathkal, India
  Electrical and Electronics Engineering — *GPA: 8.17/10* — *Thesis GPA: 9.5/10* — *2013 – 2017*
  Thesis on machine learning for human muscle EMG signals

Research Experience

- **Robotic Embedded Systems Laboratory[link], USC** — Los Angeles, USA
  Research assistant — *Sept 2020 - present*
  - Develop sample efficient leaning methods for quadruped hurdling tasks. Use Sample factory for distributed learning with reduced policy lag.
  - Experiment with audio based communication between agents for multi agent reinforcement learning.

- **Stochastic Systems & Learning Laboratory[link], USC** — Los Angeles, USA
  Research assistant — *May 2020 - May2021*
  - Build scale-able Reinforcement Learning policies using function approximators with lesser trainable parameters.
  - Study and Apply state of the art Imitation Learning techniques to self driving vehicles and experiment on Hyper realistic simulations such as CARLA.

- **Dynamic Robotics and Control Laboratory[link], USC** — Los Angeles, USA
  Research assistant — *November 2019 - October 2020*
  - Simulate and control a quadruped mini cheetah robot on Pybullet and Gazebo, by using stochastic control with policy gradient based agents. Test the RL controller on the actual robot after integration with ROS.
  - Experiment on different action spaces such as impedance control, torque control, force control, and use hybrid learning methods with model predictive control to help faster learning. Use RLLib for distributed learning.

- **St. Aloysius College, Mangalore University** — Mangalore, India
  Research intern at Laboratory of Applied Biology, Kuppers Biotech Unit — *May 2014 - June 2015*
  - Built a light chamber with variable light intensity for different wavelengths of light, for algal biofuel production.
  - Predicting growth trend of algae after studying the effect of light on enhanced algal bio-fuel production. These predictions were done using Linear regression on the collected time series data.

Industry Experience

- **Fidelity Investments** — Bangalore, India
  Software Engineer at Asset Management technology — *July 2017 - July 2019*
  - Develop applications based on Supervised Machine Learning for trade order selection and efficient execution.
  - Research on Reinforcement Learning and its application on portfolio construction in equity trading. A Gym simulation was built using real trading data. Google Tensorflow was used during the course of this work.
  - Worked with the Equity Trading team to develop and support the java and python based trading engine. Gained experience in java spring-boot, python flask, SQL, splunk, AWS and many other software developer tools.
  - Worked with the fixed income research team to build a complete end to end application using .NET and Excel VBA. Gained experience in the Microsoft Windows Presentation framework for building hard clients.

- **SalesDNA** — Los Angeles, USA
  Data Scientist — *May 2021 - August 2021*
  - Use real time Markov modelling to model a sales process. This model was used to simulate a sales process.
  - Build model free reinforcement learning algorithms to build AI strategies on this sales simulation.

Publications

- G. Salhotra, **S. Hegde**, SS. Batra, P. Englert, GS. Sukhatme (2021) *Guided Learning of Robust Hurdling Policies with Curricular Trajectory Optimization.* (under review)
- **S. Hegde**, A. Kanervisto and A. Petrenko (2021) *Agents that Listen: High-Throughput Reinforcement Learning with Multiple Sensory Systems.* [site]
- Hiteshi Sharma, R. Jain. and **S. Hegde** (2021) *Randomized Policy Learning for Continuous State and Action MDPs* (arxiv) [pdf]
- **S. Hegde**, V. Kumar, and A. Singh. (2018). *Risk aware portfolio construction using deep deterministic policy gradients.* IEEE Symposium Series on Computational Intelligence (SSCI) Bangalore, Nov. 2018. [pdf]
- Severes, A., **Hegde, S.**, D'Souza, L. and Hegde, S. (2017). *Use of LED for enhanced lipid production in micro-algae based bio-fuels and predicting growth patterns.* Journal of Photochemistry and Photobiology B: Biology, Elsevier, Volume 170, Pages 235-240.[link]

Achievements and Academic Projects

- **Autonomous Vehicle Navigation:** As a part of the Autonomous Vehicle lab, I worked on navigation, path planning and simulation of an autonomous car to take part in IGVC 2021. I used Gazebo to build an accurate simulation of the track, and implement path finding algorithms such as A star.
- **Competetive and Co-operative Multi Agent Reinforcement Learning[pdf]:** As a part of my directed research with the Hardware Accelerated Learning group[link], I'm experimented with various multi agent reinforcement learning algorithms. The goal of this project is to understand the state of the art RL algorithms that work well in both competitive and cooperative environments.
- **Torque Transfer[code]:** Use reinforcement learning and transfer learning to create robust AI agents. The AI agent should generalize to a variety of open world self driving simulations. After training an AI for a self driving car simulation using Imitation learning and reinforcement learning, the learnt policy was used as a pre trained network for an AI agent in another self driving simulation. The pretrained model showed faster learning in the new simulation.
- **Emotion Transfer on speech using spectrogram images[code]:** Use a conditional Generative Adversarial Neural Network to generate images on spectrograms of speech signals. By using cycle GANs we use style transfer on spectrograms of speech signals to embed emotion in them. The generated spectrogram is reconstructed back to speech using the Griffin-Lim algorithm.
- **The Data Open[pdf]:** Was a finalist in the SoCal round of the Data Open Hackathon organized by Citadel. Along with my team, we were able to quickly analyse and draw conclusions on data corresponding to Brexit.
- **Fashion compatibility prediction[code]:** Use a Siamese Convolutional Neural Network to classify if two fashion objects are compatible with each other. Then using the pair-wise similarity scores predicted to see if an outfit is compatible. To do this Google Tensorflow 2.0 was used and the models were trained on AWS p3.2xlarge instances (Tesla V100 GPUs)
- **Spoken Language classifier[code]:** Implement a Gated Recurrent Unit based Neural Network to classify the extracted MFCC features from speech audio. A streaming model classifies the language being spoken in real time. Using this streaming model, we could analyse the probability of miss-classification at every instant of speech.
- **Soda bottle classification contest[link]:** Winner of image classification contest by Deep Cognition (An AI company based out of Irving, Texas). I built a robust (100% test accuracy) Neural Network using a variant of the VGG architecture. To attain this accuracy of testing data I implemented my own data augmentation algorithm that helped the neural network generalize better.
- **Prosthetic Voice (Thesis)[pdf]:** Undergraduate Thesis: sEMG signal controlled speech production aid for speech challenged individuals using Machine Learning. The signals were collected, filtered, pre-processed and then fed to a classifier that would predict the hand action performed. The action would then be translated to speech.
- **Emotion Detection[pdf]:** I was part of a three member team that built a Machine Learning driven emotion detector using variations in speech signals. Using MFCC feature extraction and PCA on other features, we built a classifier.

Programming Skills

- **Languages**:
  Python (Tensorflow, PyBullet, Gym, PyTorch, Pandas, Numpy, Flask, Scikit-learn, Scipy, YOLO, ROSpy),
  MATLAB (Statistics and Machine Learning, Deep Learning, Signal Processing Toolboxes),
  Java (Spring, Springboot, Kafka, Camel, Jackson, SpringJDBC)
  C++ (OpenAL, OpenCV)
- Also have working knowledge of the following languages/technologies:
  SQL, Hadoop, Angular JS, C#, ROS, AWS