PRANAY SHAH

pranay96shah@gmail.com
@pranay_shah_
pranay-shah96
pranayshah-hub
pranaydshah.medium.com

EDUCATION

MSc in Machine Learning & Artificial Intelligence | **Distinction**

Oct 2018 – Sep 2020 **Department of Computing, Imperial College London** **Full-time**

- Term 2: Machine Learning 77% , Deep Learning 79%, Machine Learning for Medical Imaging 75%, Probabilistic Inference
- Term 1: Quantum Computing, Mathematics for ML, Computer Vision 74%, Robotics, Reinforcement Learning 77%

Notable Topics: Approximate Inference, Gaussian Processes, Decision Theory, Deep Reinforcement Learning, Evolutionary Algorithms, Graph Convolutional & Attentional Networks

MEng, BA in General Engineering and Manufacturing Engineering

Oct 2014 – Jun 2018 **Department of Engineering, University of Cambridge** **Full-time**

- Part II (Merit): Manufacturing Engineering Tripos, Institute for Manufacturing (IfM)
- Part I (2.i): Information Eng., Electronic Eng., Mathematics, Mechanical & Thermo-Fluid Eng., Structural & Materials Eng.

CEFR French (Merit): Level C1

MACHINE LEARNING RESEARCH PROJECTS

MSc Thesis, 84% - Bayesian Deep Reinforcement Learning for *de novo* Molecule Discovery - Python, Tensorflow

GLAM Group, Imperial College London

- Developed a novel **Information-Aware Bayes-by-Backprop Double Q-Network (IA-B2Q)** for *de novo* molecule generation via exploration of the drug-like chemical space, which achieves results that are at least on par with other baselines.
- First to make a deeper consideration of exploration in **Deep Reinforcement Learning** (DRL) for *de novo* molecule discovery, by being the first to use **Information Quantification** via **Bayesian Deep Learning** (BDL) in this application.
- Implemented a novel realisation of the **Information-Directed Sampling** (IDS) quantity using Bayesian uncertainty, deploying it as an action-sampling strategy and as a novel **Intrinsic Motivation** reward-shaping bonus.
- Studied how BDL can be better combined with **Double Deep Q Networks** (DDQN) beyond a simple substitution of regular deep learning for **Bayes by Backprop** (BBB) networks. The best Bayesian DDQN variant (B2Q) improves both over regular DDQN, and single Bayesian Q network estimators.
- Further investigated **epistemic** and **heteroscedastic aleatory uncertainty** quantification using the B2Q network and recent enhancements in BDL such as **local reparameterization**, **variance networks** and **posterior sharpening**.

Brain-age Regression with Medical Image Data - Python, PyTorch

Imperial College London

- Designed a **convolutional neural network** (CNN) to perform brain tissue **image segmentation** from MRI scans, and measured segmentation accuracy by computing **Dice similarity** coefficients, and engineered features by computing relative tissue volumes to perform **linear, polynomial** and **Bayesian ridge regression** to predict the brain's age.
- Investigated two further methods to perform brain-age regression: **PCA feature learning** followed by Bayesian linear regression and **Support Vector Regression**; and regression using a **3D-Convolutional U-Net** architecture.

Speech Recognition with RNNs Mini-Project, 100% - Python, PyTorch

Imperial College London

- Used the **Librosa** package to perform **MFCC feature extraction** from audio files in the *Speech Commands* dataset.
- Built basic recurrent neural network (RNN) cells from scratch, including **LSTM, GRU, Bidirectional-LSTM** and **Bidirectional-GRU** with an output **Attention** layer, to construct several RNN models that predict the word-class of the audio sample.

Discriminative Convolutional Image Classification and Generative Models Mini-Project, 98% - Python, PyTorch

Imperial College London

- Built an **MLP-based Variational Auto-Encoder** (VAE) from scratch to generate realistic ***MNIST*** digits, and visualised latent space representations using **T-SNE**.
- Implemented **ResNet-18** to classify images from the ***CIFAR-10*** dataset, and designed a **Deep Convolutional Generative Adversarial Network** (DCGAN) to generate realistic and diverse images in the style of *CIFAR-10*.

MEng Individual Project - Deep Generative Models for Quality Control in Powder Metallurgy - Python, MATLAB

GKN Powder Metallurgy, and IfM, University of Cambridge

- Developed a stacked **generative model** consisting of an **Auto-Encoder** (AE) and a **Time-Delay Neural Network** (TDNN) using **Keras** for predicting manufacturing quality characteristics in a time series.
- Experimented with various **data balancing techniques** such as Random Oversampling and Synthetic Minority Oversampling (SMOTE), **grid-search** based architecture search and hyperparameter selection, as well as regularisation methods such as **early stopping**, **dropout** and **noise injection**.

EXPERIENCE

AI Researcher (Freelance) | **Business Systems International (BSI), London, UK** | **Present**

- Technical writing on various topics within AI, such as Computer Vision in Agriculture, Bayesian Inference in Reinforcement Learning, and Deep Feature Extraction in Finance for my client, BSI.

Machine Learning Research Assistant | **DIAL Group, IfM, Cambridge, UK** | **Aug 2018 – Sep 2018**

- Implemented a natural language understanding model with **Python** and **Keras** using **Word2vec** word embeddings from patent-specific corpora such as abstracts, titles, claims and descriptions.
- Explored several RNN architectures, including **LSTM, Bi-LSTM,** and **GRU** to classify intellectual property documents based on the international standard for patents classes (IPC).
- Designed a dual-step **Conv-AE** followed by a **LSTM** to predict patent quality indicators such as forward citations.

Data Science Consultant | **Siemens Ltd., Lincoln, UK** | **Feb 2018 – Mar 2018**

- Used **Python** to build classification models such as **k-NN, One-vs-Rest Logistic Regression** and **Multi-Layer Perceptrons** to predict the repair workload per station, component yield and remaining lifetime of recommissioned turbomachinery.
- **Reduced repair procedure planning time by 35%** by automating the identification of a component's deployment status.

Manufacturing Engineering Consultant | **Rolls Royce Ltd., Bristol, UK** | **Oct 2017**

- Automated the parsing of XML-formatted engine repair instructions into JSON format using **Python** to enable comprehension by a voice-recognition device that more efficiently delivers the instructions to an operator in audio form.

Electronic and Information Engineer Intern | **Blendology, Cambridge, UK** | **Jun – Aug 2015**

- Performed rigorous **debugging, testing and improvement** of an iOS app written in **Objective-C** that received and displayed Bluetooth Low Energy (BLE) data from an electronic device designed to wireless business cards.
- Built and tested 10 prototypes of the company's newest product, the electronic-ink wireless business card, and learnt the basics of low-level **Embedded-C** programming to improve the display quality of its flexible thermoplastic screen.

ENGINEERING PROJECTS

Robotics and Automation Engineering - Robotics Lab, IfM, Cambridge, UK

- Successfully automated the assembly of 5 gearboxes with high-level **PLC programming** using **Sequential Flow Charts** (SFC) and the PASCAL-based **Structured Text** language.
- Programmed Industrial Robots such as the **6-axis FANUC M-6i** and developed end-effectors to mimic human-like holonomic translations and grasping motions.

Undergraduate Group Project - Automating art production by upcycling wine-stained corks - IfM, Cambridge, UK

- Developed the electronics to automate the photography of corks faces, and co-developed the image-processing software to process user-selected images into a grid of cork-sized pixels.
- Manufactured the electro-mechanical pick-and-place system to construct the cork-based artwork.

Undergraduate Project - Design of an autonomous parcel delivery robot - Engineering Department, Cambridge, UK

- Co-developed a path-following robot, programmed in **C++** using **PID control theory**, and won **1st place out of 14** teams.

SKILLS

- **Languages & Frameworks**: Python (Tensorflow & Keras, Pytorch, Numpy, Scipy, Pandas, Scikit-learn, Seaborn, SimpleITK, RDKit, OpenAI Baselines, Librosa), SQL (*edX certficate link*), Structured Text, and VHDL.
- **Technologies**: Tensorboard, GCP Compute Engine, Git, Conda, tmux, Bash scripting, GNU nano, and SLURM.