Chris Bamford

Email: `chrisbam4d@gmail.com` - **Mobile:** +44-7853-262-719

January 24, 2019

Education

- **First Class Hons MEng in Applied Cybernetics, University of Reading.**

- 3 A-levels (3 A), Dr Challoner's Grammar School, United Kingdom.

- 12 GCSEs (6 A*, 4 A), Dr Challoner's Grammar School, United Kingdom.

Publications

2012 Modular non-computational-connectionist-hybrid neural network approach to robotic systems.

Paladyn. Journal of Behavioral Robotics. February 6, 2012

URL: `https://link.springer.com/article/10.2478/s13230-012-0003-6`

2010 Cavalcade neural network for mobile robot.

Cybernetic Intelligent Systems, 7th IEEE International Conference, September 1-2,

URL: `http://ieeexplore.ieee.org/document/5898087/`

Professional Experience

2018 Q4-present PhD - Artificial Intelligence in Gaming, Queen Mary University

My research interests are primarily in Deep Reinforcement leaning particularly using model-based planning as a method of improving sample efficiency and model accuracy. Model-based methods are a particuarly interesting area of research as they allow artificial agents to use tools which have psychological parallels such as imagination, curiosity and empowerment. I believe that these tools are a step towards general artificial intelligence.

2018 Q3 Founder - embod.ai

embod.ai is a start-up where researchers, students and hobbiests can build and observe AI agents in Massively Multiplayer Online environments. The goal of embod.ai is to make AI based gaming and competitions a mainstream e-sport. Beta release: `https://medium.com/@chris.bamford/embod-ai-come-join-the-beta-efcdbc85f524` Website: `https://embod.ai`

2018 Q2 EF10LDN Cohort Member - Entrepreneur First

Entrepreneur First is the UK's most successful startup accellerator programme. Entrepreneur First gathers up to 100 highly talented individuals in two cohorts a year and helps them to build globally important companies.

2015-2017 Lead Research and Development engineer at import.io

import.io specializes in making it easy for non-technical users to gather and structure data from the web. I have worked at import.io since it was founded in 2012. I have worked mainly on designing and creating innovative technologies to generate a leading edge over any competitors. I have worked mainly in Java when building infrastructure and backend projects, and Python when doing Machine Learning.

Some of the projects of which I have had significant input and I have enjoyed working on:

- **Bees** - import.io's highly distributed and scalable querying and messaging platform. Bees handles many challenges that are faced when building a distributed web extraction system, such as variable and unknown network latency, asynchronous IO across multiple servers and thousands of endpoints, pipelining and sorting of sparse data sets and collation of data from multiple sources. Bees also had to record and supply metrics for the business such as success rates, per-user statistics and billing information. Uses technologies such as *Spring, Redis, S3, EC2, ElasticBeanstalk, CloudFormation and Docker*

- **Label Suggestion** - A statistical model that would return the "most likely next label" that a user would chose for a data column given the previous data columns the user had inputted. Label suggestion used some highly optimized BLAS libraries to quickly infer probabilities over a large data set of manually labelled training data.

- **Deep Learning** - Research projects with deep learning and web data extraction. For example I worked on several page classification approaches using Recurrent Neural networks and Recursive Neural networks. Additionally I worked on data point recognition where i worked to develop algorithms that would classify where there were datapoints in a webpage and then classify what label those data points should be given. The research was performed using a few different deep learning libraries; deeplearning4j, neon, keras (experimenting with tensorflow and theano).

2012-2015 Software Engineer at import.io

Before the machine learning and algorithm team was created at import.io, We were a small team of 3-5 engineers. At this time I mainly worked on the main Java codebase.

Some of the projects of which I have had significant input and I have enjoyed working on:

- **"Magic"** - An algorithm that finds and extracts lists of records in a webpage. This product made use of several different matching and allocation techniques such as Jaccard distance, Hirschberg's algorithm and Munkres(Hungarian) algorithm. The average time to detect and extract the top "most likely" tables was less than a second. *The success of this project lead me to become the Lead Research and Development engineer.*
- **Object Store** - A distributed platform for storing and retrieving data objects from an underlying database, originally a Percona SQL database was stored, with binary attachment storage on amazon S3, Later this was migrated to use amazon auroraDB.

Open Source Contributions

deeplearning4j/nd4j Helped to write one of the first versions of GPU interfacing for the deep learning 4 java library developed by skymind.io

Commits: `https://github.com/deeplearning4j/nd4j/commits?author=Bam4d`

Neon Added a statistics callback which contains precision and recall metrics for multi-label classification problems.

Commits: `https://github.com/NervanaSystems/neon/commits?author=Bam4d`

Notable Personal Projects

K-dimensional connect-N This was an experiment in Deep Q Learning with the game connect-4.

The game was expanded to allow any number of dimensions and any length of player tokens for a winning board layout.

Blog Post: `https://bam4d.github.io/#/post/k-dimensional-connect-n--part-1-environment/2`

Python-microRTS A python library for interacting with the microRTS AI research environment.

Github: `https://github.com/Bam4d/python-microRTS`

Neuretix A spiking neural network implementation using the models I developed as part of my two publications. Neuretix uses a novel ring-buffer data structure to queue spikes. This allows computational complexity of forward pass updates reduce from 'O(n^2)' to 'O(n)', where 'n' is proportional to the number of neurons firing at a single time step.

Github: `https://github.com/Bam4d/Neuretix`

Neural Stacks (Keras deep learning library) Although I did not contribute this implementation back to the Keras group, I enjoyed implementing the differentiable stack algorithm and seeing it work.

Github: `https://github.com/Bam4d/keras/tree/neural-stack`

Key Skills

Python I have worked with python in many different types of projects, from research with deep learning libraries such as neon, keras, pytorch, tensorflow and scikitlearn; To building REST API services using libraries such as Flask when productionizing machine learning models.

Java The majority of my work at import.io was using Java 8. I have been using java 11 recently on personal projects. I have used Spring, SpringBoot, Hazelcast, Jedis, Jackson, Lettuce, Jetty 8/9, mockito, cucumber, gherkin and Google libraries such as Guava, Lombok and Futures.

Amazon AWS I have used a large number of features of amazon aws while at import.io, embod.ai and in some personal projects. I am comfortable with using Lambda, CloudFormation, ECR, ECS, ElasticBeanstalk, Route53, S3, EC2, Redshift, AuroraDB and ElastiCache

C/C++ Several of my personal projects and open source contributions not mentioned here have used C/C++.