Google Summer of Code - 2015

Fast Interactive Machine Learning Enabled by GPUs Integrating Deep Convolutional Network Features with an Interactive Image Machine Learning System Spatial Extensions to MongoDB Develop a geospatial cache using HTML5 IndexedDB TCIA Data Exploration and Information Visualization Browser based Bulk Data Transfer

Fast Interactive Machine Learning Enabled by GPUs

Mentor: Lee Cooper (lee dot cooper at emory dot edu)

Overview: Enabling users to interact with large datasets via machine learning algorithms requires fast algorithm response times. This project will develop GPU pipelines for an active machine learning setting, where classification algorithms sift through millions of samples to select key examples for labeling by human experts. Increasing the speed of these pipelines will improve the user experience and the development of machine learning classifiers for cancer research. This pipeline will interact with a database containing millions of samples and their feature descriptions, and will feed results into a web framework in order to collect user feedback.

Programming Languages/Frameworks: C/C++, CUDA/OpenCL

Prerequisites: C++ programming skills, experience in CUDA/OpenCL, basic experience with machine learning algorithms Level of Expertise: Intermediate.

Integrating Deep Convolutional Network Features with an Interactive Image Machine Learning System

Mentor: Lee Cooper (lee dot cooper at emory dot edu)

Overview: Our team is developing an open-sourced active machine learning system to enable medical professionals and scientists to interactively build image classifiers for medical imaging datasets containing millions of samples. One of the most promising emerging areas of machine learning is deep learning, where neural networks are used to learn features for image classification. Our team has been successful in developing deep learning prototypes for our active learning system using the Python library Theano, and we are ready to take the next steps to fully integrate these capabilities. In this project you will be responsible for integrating our deep learning prototypes with the active learning system to create a framework for feature generation and storage. You will work with a team of machine learning engineers and biomedical researchers to test and validate this framework. **Programming Languages/Frameworks**: Python, C/C++

Prerequisites: Python and C++ programming skills, basic experience with machine learning algorithms, basic experience with parallel computing Level of Expertise: Intermediate.

Spatial Extensions to MongoDB

FAQ: Click here

Mentor: Ashish Sharma (ashish dot sharma at emory dot edu)

Overview: In one of our projects we use MongoDB to manage shapes extracted from very high resolution (50Kx50K) digital pathology images. The resulting data is massive — >1B shapes from 100K images. When viewing and exploring these images, one would like to exploit the MongoDB 2d geospatial indexing system (2dspherical can't be used because it is in spherical coordinates). However the 2d index is limited because Mongo insists that it be the first index in a compound index (2d + image metadata). In this project you will create a custom 2d index that is appended to all documents and will extend Mongo's Java driver so that spatial queries can exploit your new 2d index.

Programming Languages/Frameworks: MongoDB, Java

Prerequisites: Java programming skills, experience with Mongo, experience in computational geometry or spatial query processing will be useful but is not a prerequisite.

Level of Expertise: Intermediate.

Develop a geospatial cache using HTML5 IndexedDB

FAQ: Click here

Mentor: Ashish Sharma (ashish dot sharma at emory dot edu)

Overview: In our digital pathology platform (camicroscope.org), you would add a spatial cache using the browser's database. The presence of such a cache will give users the ability to use cached objects and avoid having to retrieve segmented objects from the remote web service **Programming Languages/Frameworks**: Javascript

Prerequisites: Javascript, experience with HTML5 Indexed DB

Level of Expertise: Intermediate.

TCIA Data Exploration and Information Visualization

FAQ: Click here

Mentor: Ashish Sharma (ashish dot sharma at emory dot edu)

Overview: The Cancer Imaging Archive provides access to a wealth of biomedical cancer imaging data. It contains over 26 million radiology images, pathology data, and clinical data. The existing web interface for searching the archive is extremely outdated. Recently a REST API for TCIA was implemented to allow programmatic query and download of the data. Using the new REST API this project would seek to create a new search interface to the data as an alternate way to explore the contents of TCIA, create dynamic dashboards that can be extended to support the exploration of TCIA data (similar to "http://nickqizhu.github.io/dc.js/"). In addition to searching the TCIA archive, this project could also include support to intuitively formulate queries that can federate data from other remote archives. Possible strategies could include Microsoft Pivotviewer which provides an interactive data exploration platform.

Programming Languages/Frameworks: Javascript, d3, crossfilter, HTML

Prerequisites: Extensive experience with jQuery. Experience/Coursework in HCI, visual interface development.

Level of Expertise: Intermediate.

Browser based Bulk Data Transfer

FAQ: Click here

Mentor: Ashish Sharma (ashish dot sharma at emory dot edu)

Overview: Develop a web based system to transfer really large quantities of data — large is > 10s of GB. A possible strategy would involve maintaining an in-browser database that tracks the progress of data download, and can manage multiple download streams. Needless to say there will be a server side component that will be provided with a manifest of what needs to be downloaded.

Programming Languages/Frameworks: Javascript

Prerequisites: Javascript, experience with HTML5 Indexed DB

Level of Expertise: Intermediate.