Dong Luo

- 270 Babcock Street, Apt 16B, Boston, MA 02215
- Phone: (617) 755 2974
- Email: dongluo@gmail.com Home Page: http://www.cs.umb.edu/~dluo

Professional Statement

Programmer, problem solver. Heavy user of Java, C. Normal user of Python, Tcl/tk. Casual user of MATLAB, C++, Objective-C, GTK, Weka, Javascript, SQL, C#, ASP.NET. Skilled with Windows, Mac, Linux operating systems.

Education

UMass Boston, Boston, MA, 2012-present, Computer Science
Related courses (GPA 4.0):
Artificial Intelligence, Applied Machine Learning, Parallel Computing
Database Management Systems, Database Architecture
Database-Backed Web Sites Development, Cloud and Mobile Computing
Boston University, Boston, Post doc, 2010-2012, Biophysics
Boston University, Boston, MA, 2009, PhD, Biophysics

Research/Course Projects

KMR Algorithm Optimization (2014)
KMR is a fast algorithm to search all subsequence repeats in a long string. By tweaking data structure and hard-coding input data, the serial performance is improved 5 times faster on extremely lengthy genome sequence. Paralleling gave another 2 times boost (synchronization prevents further improvement).
Technology used: Java

Iowa state Flooding Prediction (2013)
Apply supervised machine learning techniques on big amount of spatial data to build prediction model. Feature selection is the focus because too many data are available while the relevance is not obvious.
Technology used: Weka, ArcMap, Java

Website Development (2013)
Build web site that dynamically queries database and displays material in readable fashion, use professional three-tier model.
Technology used: XHTML, CSS, Javascript, C#, ASP.NET

Cloud and Mobile Programming (2012)
Analyze word frequency in English wiki titles and evaluate computing performance. Apply GPS location and Map functions in mobile application.
Technology used: AWS, Hadoop, Android mobile API, Java

Personal Project

RadeonHD Framebuffer Driver (2008-2010)
Adapt Linux C source code, RadeonHD, to Mac OSX platform. Functional video driver for OSX is delivered after great efforts.
Technology used: C/IOkit C++, kernel debug, reverse engineer, Git
Link: http://code.google.com/p/osx86-driver-radeonhd/