

DAVID M. BERLIN

8840 Costa Verde Blvd. #3425 • San Diego, CA 92122 • 858-366-2935 • dave@daveberlin.com

WORK EXPERIENCE

Teaching Assistant

University of California, San Diego

September 2003 to June 2005

- Held weekly discussion sections
- Proctored and graded tests and quizzes
- Assisted in a variety of undergraduate classes:
 - Introduction to Computer Science and OOP (Java)
 - Advanced Data Structures
 - Fluency in Information Technology

Data Analyst

Northern California Power Agency

December 2000 to August 2003

- Researched and recommended new products and solutions
- Authored a comprehensive internal network security proposal
- Implemented an NT-based printing solution

Information Systems Technician

Northern California Power Agency

June 1999 to November 2000

- Supported users with hardware and software issues
- Built a Web page for users to troubleshoot common software problems
- Assisted in converting a Thinnet network to twisted pair Ethernet

EDUCATION & CERTIFICATIONS

Master of Science in Computer Science (in progress)

University of California, San Diego

September 2003 to June 2005 (courses completed—exit exam in Sept.)

- Concentration: Artificial Intelligence
- Courses in machine learning, neural networks, and computer vision
- Additional papers in network spectroscopy and algorithm analysis
- Experience: Java, Perl, MATLAB, C, XHTML, CSS
- GPA: 3.57

Bachelor of Arts in Mathematics & Computer Science

California State University, Sacramento

August 1998 to May 2003

- Minor in Business Administration
- Experience: C++, Visual Basic, Python, ASP
- GPA: 3.84

Microsoft Certified Systems Engineer, NT 4.0

1999 to 2000

- Electives included IIS 4.0 and TCP/IP

RESEARCH PROJECTS AT UCSD

Talking Neural Network

Created a neural network that learned to generate phonemes corresponding to some input text. These phonemes were fed into a speech synthesizer, which then vocalized the input text. In essence, the neural network learned how to read English words and pronounce them correctly. The network was built in MATLAB, with supporting text processing done in Java.

Honey Farming with Xen

This project tested the feasibility of using Xen, a virtual machine monitor that can run multiple operating systems, to build a virtual honey farm. Each operating system running on top of Xen would be set up as a honey pot, protected from every other instance of the OS. This type of setup could drastically decrease the costs associated with building and maintaining a honey farm. Research included installing and working with Xen, and testing the system's stability and capacity to handle multiple installations of Linux.

Bayesian Spam Filtering

Built a spam filter using a naïve Bayes learning algorithm. The first step was to identify features that could differentiate between spam and legitimate e-mail. A learner was written that read the corpus of e-mail messages and learned the patterns of spam and non-spam. The output of this learner was used by a classifier to label new e-mail.

