Yang Zhang

1915 NE Terre view Dr Apt 51E, Pullman, WA 99163 | (206) 465-3826 | yang.zhang7@wsu.edu

EDUCATION

University of Washington, Seattle, WA
B.S., Electrical Engineering 09/2010 – 12/2014
GPA: 3.42/4.0

Washington State University, Pullman, WA
M.S., Computer Science (in progress) 08/2016 – 05/2018
GPA: 3.6/4.0

TECHNICAL SKILLS

Language: Java, Python, C, C++, BASH
Programming Concepts: Objected-Oriented Programming, Complexity Analysis and MVC
Web Technologies: JavaScript, HTML, XML, CSS, PHP, AJAX, JSON
Operating Systems: Linux, Mac OS X and Windows
Database: CouchBase, SQL Server, SQLite and MySQL
Machine Learning: Supervised Learning (such as perceptron, SVM),
Unsupervised Learning (such as K-Means, EM),
Reinforcement Learning (such as Q-learning)

PROJECT EXPERIENCE

Shaped Wall

- Worked with both computer science team and architecture team to build a smart wall that change its shape according to users' emotion state and other bio data.
- Implemented eye blink events with Tobii eye SDK (C#)
- Implemented face expression detection with Affectiva SDK (JavaScript)

Checkstyle extension plug in (Java)

- Based on the starndard Checkstyle (eclipse plug in), extends the tool with more anti-pattern detection including (blob class, feature envy and duplicate code)

Chinese Segmentation Tool (Java)

- Implemented a Chinese Segmentation tool by applying Hidden Markov Model.

Flappy Bird Agent by Q-Learning (Python)

- Implemented a Q-learning agent that can constantly achieve score more than 500 points in the classic game Flappy Bird.

Gridworld Solver Agent by Q-Learning (Python)

- Implemented a reinforcement learning agent that can learn the optimal policy from the giving grid world environment by using Q-Learning algorithm.
- The agent adapts two different explore/exploit politcies (e-greedy and Boltzman exploration)

MemoMe (Notebook application for Android)

- ❖ Designed a notebook application, which consists of SQLite Database, and ViewPagerIndicator (an open-source library) to help users record and manage their daily memos.
- ❖ Implemented grid-pager styled preview UI that provides an elegant and effective interface for memo browse.
- ❖ Featured extra save mechanism embedded in activity life cycle, preventing accidental user data lose.

Minesweeper game in Java

- ❖ Implemented replica of the classic Microsoft game Minesweeper, include all original functions (left click to open one cell, right click to flag one cell, double click to explore around cells).

Snake game in Java

- ❖ Implemented replica of the classic Nokia snake game, with original playing style

Path Navigator in Java

- ❖ Designed a graph application capable of reading location and map from input file, and showing user the shortest path and path with minimum cost between two locations.
- ❖ Built a searchable graph object and implemented BFS, DFS and Dijkstra's algorithms.

Video Store Application in SQL Server

- ❖ Designed a client-side application, which consists of SQL Server, transaction management, SQL injection prevention to enable the users to rent/return films, change viewing plan and check the availability of films.
- ❖ Designed ER diagram and relational model for the execution of database and queries.

To-Do List (Web application in PHP and JavaScript)

- ❖ Designed an online to-do list application, which consists PHP, JavaScript, Cookie management for login session, injection prevention and regular expression, to help users manipulate their memo.

WORK EXPERIENCE

Expeditors, Sys Admin Intern, Seattle, WA Jun. 2017 – Aug. 2017

- ❖ Implemented the Automated Continuous Integration (ACI) web tool for Gitlab CI (Python with Django)
  - ACI tool helps developer and QA tester to apply CI in unit testing and system deployment
  - Developed in Python with Django (back-end) and JavaScript (front-end)
  - Authentication uses OAuth 2 workflow

- ❖ Worked with SA-QA team and learned the standard QA process including the method of developing solid test plans and recording results from test cases

Qikspace, SDE Intern, Seattle, WA Jun. 2014 – Sep. 2014

- ❖ Implemented push notification service for both server side (Python) and Android side.
- ❖ Implemented server-side tool-kit allowing the users to access and manage their Google Drive contents where OAuth 2.0 was employed for authentication process.
- ❖ Implemented the basic cache server for the server side. (Redis)

- Implemented database synchronization service for Android side.
- Implemented the basic download and upload service for Android side. (Retrofit)