Yuta Nakamura

yutanak6@gmail.com

Address:
222 W. 26th St., Apt. 202
Chicago, IL 60616

OBJECTIVE

To obtain a 2024 summer internship as a software engineer

EDUCATION

DePaul University   March 2020 – December 2024
Ph.D., Computer Science GPA 3.90/4.00
Coursework:

Distributed Systems I/II	Compiler Design	Advanced Database Concepts
Light Weight Virtual Machines	Database Technologies	Optimized C++

University of Chicago   October 2018 – March 2020
M.S., Computer Science GPA 3.38/4.00
Focused on Data Analysis (Machine Learning/Artificial Intelligence)
Coursework:

C Programming	C/C++ for Advanced Programmers	Operating Systems
Introduction to Computer Systems	Foundations of Computational Data Analysis	Time Series Analysis and Stochastic Process
Algorithms	Introduction to Computer Security	Advanced Data Analytics
Machine Learning		High Performance Computing

Hitotsubashi University   April 2012 – March 2018
B. A., Economics GPA 3.73/4.00
Relevant Coursework:

Introductory Statistics	Calculus 1B/2	Basic Course on Information Science
Introduction to Probability	Further Calculus	
Basic Econometrics	Set And Topology 1/2	Computer Science
Linear Algebra 1B/2	Mathematical Logic 1	Applied Information Technology

University of Pennsylvania   August 2014 – May 2015
Exchange Student
Relevant Coursework:

Econometrics	Calculus III/IV	Intermediate Spanish
Game Theory	Advanced Calculus	

PROFESSIONAL EXPERIENCE

DePaul DICE Lab Chicago, Illinois
Researcher   June 2019 – Present
- Published 4 papers as a first author at peer-reviewed conferences about provenance alignment
- Currently creating software to apply provenance alignment for programs with nondeterminism

Imagine Global Care Tokyo, Japan
Marketing Department/ Intern   January 2016 – August 2018
- Organized data in Excel and came up with marketing plans to sell the supplement product called Brolico
- Worked as a translator to connect the department in America and in Japan

---

PROJECTS

Ordo Ab Chao (A Private Repository at DePaul Dice Lab)   December 2022 – June 2023
Experiment part of the publication "Efficient Differencing of System-level Provenance Graphs"
Defined another way to "diff" 2 sequences of trace while respecting loop iterations, which is vital for program analysis. The asymptotic time of the algorithm is linear (to the length of paths) with optimizations.

Raft Algorithm Implementation   May 2022 – June 2022
[https://github.com/nakasan617/Raft]
Implementation of Raft, an algorithm to reach consensus value of a variable in a fault-tolerant way
Implemented with Akka, a toolkit for simplifying concurrent application development on JVM.

ProvScope   November 2021 – June 2022
[https://github.com/depaul-dice/ProvScope]
Experiment part of the publication "Provenance-based Workflow Diagnostics Using Program Specification"
Given two traces and program specification (i.e. control flow graphs of each function), maps the traces onto the traces to convert to paths and compares them.

Brief Blockchain Implementation   October 2021 – November 2021
[https://github.com/nakasan617/blockChain]
3 processes compete to verify the newly added block and shares among other processes in Java. Also implemented Proof-of-work algorithm, finding a correct random string that yields the specified hash value to make the longest blockchain to be the correct blockchain given majority of the computing nodes are honest.

Content-Defined Merkle Tree   June 2019 – November 2020
[https://github.com/depaul-dice/CDMT]
Experiment part of the publication "Content-Defined Container Delivery"
Implements Content-Defined Merkle Tree, which is a Merkle tree created on top of data which are divided with content-defined chunking. The Merkle tree is fragile against insertion operations, therefore we do the content-defined chunking on internal nodes of the tree for the robustness.

Taiwan Stock Exchange Weighted Index Model   September 2019 – December 2019
[https://github.com/clairechingching/Taiwan-Stock-Exchange-Weighted-Index-Model]
Built an prediction model for Taiwan Stock Exchange Weighted Index using the historical trading data with various machine learning methods including regression, online learning, and LSTM in Python 3 with many data science libraries including numpy, pandas, scikit learn etc..

Undefeatable Connect-4   December 2018
Connect-4 that an opponent plays the min-max strategy (i.e., undefeatable strategy) to beat the player in C.

SKILLS
- Proficient: C, C++, Python 3, x86 assembly, oracle SQL, Java, Vim, Linux, Git
- Intermediate: OpenMPI, Akka, LLVM, CUDA, HTML, CSS, Scala, Visual Studio, IntelliJ

PUBLICATIONS
- Efficient Differencing of System-level Provenance Graphs   October 2023
  (32nd ACM International Conference on Information and Knowledge Management)

- Provenance-based Workflow Diagnostics Using Program Specification   December 2022
  (IEEE International Conference on High Performance Computing)
  Introduced debugging tools using function call trace of executions, uses numerous optimization techniques to reduce search space for "path explosion problem"

- Content-Defined Container Delivery   December 2020
  (IEEE International Conference on High Performance Computing)
  Introduces a new data structure to allow fast access of parts of data divided with content-defined chunking

- Efficient Provenance Alignment in Reproduced Executions   June 2020
  (The Theory and Practice of Provenance in Provenance Week 2020)
  Introduced system call sequence alignment

AWARDS
- National Science Foundation and N.A.S.A. Research Grant   March 2020 – Present
- Josuikai Study Abroad Grant at University of Pennsylvania   August 2014 – May 2015

LANGUAGES
- Japanese – Native Speaker
- English – Native Fluency