

# TONGLU YANG

+1(608) 320-9249 ◇ Madison, WI

[tyang328@wisc.edu](mailto:tyang328@wisc.edu) ◇ [linkedin.com/in/tongluy](https://www.linkedin.com/in/tongluy) ◇ [tongluy.github.io](https://github.com/tongluy)

## EDUCATION

**University of Wisconsin-Madison**

Sep 2021 – Dec 2023

Bachelor of Science, Computer Science

Madison, WI

- Academics: GPA 3.94/4.00, Fall 2021, Spring 2022, Fall 2022 in Dean's List
- Selected Coursework: Algorithm and Data Structure, Machine Learning, Operating System, AI, UI Design

**Macau University of Science and Technology**

Sep 2019 – Jun 2021

Bachelor of Business Administration, Accounting

Macau

## SKILLS

- Programming Languages: Java, C/C++, C#, Python, R, .Net, SQL, Swift, PHP, Go, JavaScript, TypeScript
- Frontend Development: React, Node.js, JavaScript, HTML, CSS, Bootstrap, AngularJS
- Backend Development: MySQL, MongoDB, Flask, SQLite, JSON, MVC
- Tools and Platforms: Git, GitHub Actions, Azure (App Service, Functions), Linux, MS Visual Studio

## WORK EXPERIENCE

**UW-Madison, Wisconsin Athletics - Digital Platforms, Data, and Cloud Team**

Apr 2022 – Present

Full Stack Developer

Madison, WI

- Developed and maintained a responsive and readable [website](#) by updating it from Bootstrap 3 to Bootstrap 5 in MVC framework using C# and JavaScript with NPM JavaScript packages to improve server response time.
- Implemented and managed a questionnaire feature in an ASP.NET Core application, allowing users to input and access data through controller methods and views.
- Conducted data processing analysis, including file decryption, metadata retrieval, and data insertion using SQL.

## PROJECTS

**Xv6 Kernel: Copy-on-Write Forking and Stride Scheduler Implementation**

Mar 2023

- Implemented copy-on-write (CoW) mechanism to improve xv6 fork() implementation, which reduced the time taken by the system call, and optimized memory usage by copying only when needed in C.
- Implemented a new scheduler algorithm called stride scheduling into xv6 kernel, by assigning tickets to processes and choosing the process with the minimum pass value to run next, resulting in a more efficient CPU allocation.

**Scene Recognition with LeNet-5 and Customized CNN Models**

Jan 2023

- Designed and implemented a LeNet-5 model in PyTorch with six stages for scene recognition. Explored various training configurations, such as batch size, learning rate, and training epochs, and achieved a thorough understanding of deep neural networks by training a convolutional neural network on the MiniPlaces dataset.

**Hierarchical Clustering on Pokemon Stats**

Nov 2022

- Applied hierarchical clustering algorithm in Python with NumPy library to group Pokemon data into clusters, achieving a complete linkage hierarchical agglomerate clustering on a 6-dimensional feature representation for improved organization and easier visualization (using matplotlib library).

**AI Teeko Player**

Sep 2022

- Developed an AI game player for Teeko in Python, utilizing a minimax algorithm with a depth cutoff of 5 seconds and a heuristic scoring function for non-terminal states, as well as implementing helper functions to generate and evaluate successors based on game value.