Chen Sihan

Phone: +86 18175686352 | Email: sihanchen2001222@gmail.com
GitHub: https://github.com/sihan-chen-yes?tab=repositories | Website: https://sihan-chen-yes.github.io/

EDUCATION

Beihang University, Shen Yuan College Sept. 2019 – Present

BEng (Hons) in Computer Science and Technology

- CS-related GPA:**3.88**/4.0, CS-related Weighted Average Score:**93**/100, Rank: Top **5%**
- Overall GPA: **3.84**/4.0, Weighted Average Score: **92**/100
- Programming Language: Proficient in C/C++, Python, Java, Swift, Go, SQL and Verilog
- Awards and Honors:

1st Prize in Lanqiao Cup National Programming Competition (top 2%)	May. 2022
2nd Prize in Feng Ru Cup Competition of Beihang University (top 2%)	July. 2020
Merit Student of Beihang University (top 5%)	Nov. 2020
Special Prize of Academic Competition Scholarship of Beihang University (top 0.5%)	Nov. 2022
1st Prize of Excellent Learning Scholarship of Beihang University (top 1%)	Nov. 2022
2nd Prize of Excellent Social Work Scholarship of Beihang University (top 5%)	Dec. 2020
Shen Yuan Honors College To Cambridge Short-term Visiting Scholar Fund	July. 2021

PUBLICATION

- Chen Gao, **Sihan Chen**, and Si Liu**,** "Structured Memory State Evolution for Vision-Language Navigation", submitted to *IEEE Transactions on Multimedia*

RESEARCH

VLN Model Optimization May. 2022 – Present

Research Assistant, Supervised by Prof. Si Liu

- Proposed the Structured state-Evolution (SEvol) module to solve the flaw of over-compression of object-level spatial-temporal information in NvEM, and enhanced the performance of it on R2R, R4R and REVERIE datasets
- Based on A2C algorithm, used Reinforced Layout clues Miner (RLM) module to select objects appropriately
- Employed Dynamic Graph Neural Network (DGNN) to aggregate the spatial-temporal information of objects
- Based on GRU model, proposed mGRU model (matrix version), accomplished the renewal of weight in DGNN at every time step

Interactive Robotic Dog Oct. 2021 – May. 2022

Research Assistant, Supervised by Prof. Si Liu

- Built a robotic dog with the function of voice interaction and guiding the blind
- Implemented object detection and speech recognition module based on YOLOv5 and CMUSphinx respectively
- Implemented main control module based on SDK of motion and information generated by two modules mentioned

INTERNSHIP

ByteDance, Beijing July. 2022 – Sept. 2022

Back-end Development Engineer of Master Data Department of Lark Suite

Responsible for data management (query, storage and transfer) and participated in three important projects as below:

Message Deduplication of the ETL Module

- Combined the messages with same content in the Message Queue of ETL module, which reduced the pressure of query interfaces of Vault Service
- Chose distribution architecture based on Partition Key of RocketMQ rather than centralized architecture based on Redis, to avoid complex and inefficient lock operation and consideration
- Changed the offset of RocketMQ to the one before fault for disaster recovery, which assured the consistency of data and no messages loss

Optimization of Data Table Design

- Changed the *manage_relations* table which has been put into use into *job_data* table, because of the poor performance on both function and efficiency of the former one, due to the imperfect design before
- Kept two tables working simultaneously using double-write strategy to avoid data loss and pollution
- Wrote Go script to assure the correctness of combination and storage of data
- Increased the rate of flow online slowly, which prevented huge online fault during the table-change process

MultiGeo Function of Vault Service

- Transferred the sensitive information of Singapore employee from China to Singapore, due to the legal requirements
- Used the strategy of soft deletion followed by a hard deletion, which avoided data loss in transfer process
- Added the support for cross-region storage and cross-region query for Vault Service

JD.com, Beijing Nov. 2022 – Present

Back-end Development Engineer of Technology and Product Innovation Department of JD Technology

Participated in the development of Ku+ AI platform, a software based on AutoML and MLOps, designed for enterprise-level AI research

Model Deployment Service

- Developed the online deployment service, made the algorithm model as a RESTful API, holding the computing resources to assure the real-time capability
- Built the offline deployment service to process the offline data, and then wrote the results back into MySQL as persistent storage

Computing Resources Scheduler

- Built a computing resources scheduler based on Docker and Kubernetes to improve the availability of the resources
- Implemented several scheduling strategies, including Binpack, Spread, Coscheduling, Gang scheduling

PROJECT

Multifunctional Movie Website Sept. 2021 – Dec. 2021

- Developed a website for watching movies, deployed it on my private cloud server already
- Click to visit (http://movie.ito.vin/), Account:123, Password:123 for tourist
- Used Vue and Django frame to build the front and back end respectively, and chose MySQL as database, Redis as cache layer, RocketMQ as message queue and Nginx as load-balancer
- Improved the security of website via anti-injection of database and ciphertext of password storage

SysY-based Compiler Sept. 2021 – Dec. 2021

- Built a compiler using Java, which can transform SysY (a subset of C) from source code to MIPS assembly code
- Included modules of Lexical Analysis (Automata Theory-based), Syntax Analysis (Recursive Descent-based), Semantic Analysis (Abstract Syntax Tree-based), Middle Code Generation (Quaternary Formula-based), Target Code Generation (MIPS-based)
- Optimized the compiler via Inline Function, Loop Optimization, Register Allocation Optimization, among others
- Ranked **24**/250 in the Compiler Competition (according to the weighted result of perf instruction on Linux)

ARC-based Page Replacement May. 2021 – June. 2021

- Simulated the page replacement of cache in operating system, and used Adaptive Replacement Cache (ARC) algorithm to decrease miss rate, which considered both recency and frequency, with adaptivity to the current memory access mode by adjusting capacity of cache for both dynamically
- Built data structure with specific function and low memory usage from scratch rather than using STL container
- Ranked **1**/250 in the Page Replacement Competition (according to the weighted result of perf instruction on Linux)

MIPS-based Operating System Kernel Mar. 2021 – June. 2021

- Built a MIPS-based operating system kernel via Linux platform, using C programming language
- Supported memory management, virtual address, interruption handling, inter-process switch and communication, disk management, file system and IO operation of device

MIPS-based CPU Oct. 2020 – Jan. 2021

- Implemented a CPU supporting five-stage pipelines (IF,ID,EX,MEM,WB) and interruption and exception handling
- Supported 53 instructions totally in MIPS instruction set, where assembly instructions of C programs can be run