Xiangzhi Tong

Email: xt242@cornell.edu | **Mobile:** +1 9292649212 | **LinkedIn:** xiangzhi-tong

EDUCATION

Cornell University | Ithaca, New York 09/2023 – 12/2024
M.Eng. in Electrical and Computer Engineering
Xi'an Jiaotong-Liverpool University (XJTLU) | Suzhou, China 09/2018 – 07/2022
B.Eng. in Computer Science and Technology | **GPA:** 3.78/4.0 | **Ranking:** 4/67 | **TOEFL:** 106(S22)

Relevant Courses (XJTLU): Computer Systems | C++ Programming and Software Engineering | Advanced OO Programming | Introduction to Databases | Big Data Analytics | Bio-Computation | Embedded Computer System | Digital Electronics
Coursera: Data Structures and Algorithms | Discrete Mathematics | IBM Machine Learning | Deep Learning

AWARDS & HONORS

National Third Prize: (Top 4%), 9th National Optoelectronic Design Competition 08/2021
Top 10%: University Academic Achievement Award 2018-19 & 2020-21

PUBLICATIONS

- P. Zhang, R. Kannan, **Xiangzhi Tong**, A.V. Nori, V.K. Prasanna. *SHARP: Software Hint-Assisted Memory Access Prediction for Graph Analytics,* 2022 IEEE High Performance Extreme Computing Conference (HPEC), 2022, September, Waltham, MA, USA. DOI: 10.1109/HPEC55821.2022.9926307
- **Xiangzhi Tong**, P. Ni, Q. Li. *Urban Crime Trends Analysis and Occurrence Possibility Prediction based on Light Gradient Boosting Machine*. International Conference on Big Data and AI (BDAI), 2021, May, Qingdao, China 98-103
- H. Bao, C. Zhang, **Xiangzhi Tong**. *An Automatic Waste Sorting and Collecting Cart*. 2021 2nd International Seminar on Artificial Intelligence, Networking and Information Technology, 2021, November, China, 577-588
- Q. Yuan, P. Ni, J. Liu, **Xiangzhi Tong**. *An Encoder-decoder Architecture with Graph Convolutional Networks for Abstractive Summarization*. International Conference on Big Data and AI (BDAI), 2021, May, Qingdao, China, 91-97

SKILLS

Programming: Java | Python | Rust | Go | MySQL | JavaScript | Assembly
Libraries: Scientific Computing (*NumPy*, *SciPy*) | Data Processing (*Pandas*) | Deep Learning (*TensorFlow*, *PyTorch*) | Visualization (*Matplotlib*) | Web Crawling (*BeautifulSoup*, Request) | Cryptography (*Cryptography, Md5*)
Software: IDE (Visual Studio Code, IDEA, PyCharm) | Project Management (GitHub)

WORK EXPERIENCE

***Software QA Engineer (Full-time)* | Zoom, Suzhou, China** 07/2022 – Present

- Developing an internal full-chain testing system and respective operation system:
  - Constructed a centralized server cluster based on *Docker* and *Kubernetes* to synchronously update configurations, distribute UI Test Cases and sent related notification or warning emails automatically.
  - Developed Data Factory to generate concurrent network requests using *Golang* and *AWS Cloud*; generated corresponding stress testing scripts and reports with *JMeter*; simulated previous traffic overload scenarios using *GoReplay.*
  - Encapsulated unsupervised learning model into APIs to identify abnormal traffic; provided timely feedback on detected abnormal conditions to the network security department.
  - Saved 30% traffic computation resources and achieved fully automatic testing pipeline by using *batch* scripts.

***Software Development Engineer (Intern)* | Zoom, Suzhou, China** 03/2022 – 06/2022

- Constructed an internal UI testing framework for the Zoom Contact Center (ZCC) website:
  - Applied *Selenium* library and *UnitTest* in *Python* to develop a complete framework; accomplished test automation cases with 80% coverage of the ZCC main functions.
  - Encapsulated *OpenCV* library and Perceptual Hash algorithm to test the display functionality of Zoom whiteboard;
  - Deployed timed API stress testing tasks automatically by writing scripts of *Apache Bench*;
  - Assisted the server development team in formulating a server maintenance strategy;

***Full-stack Development Engineer (Intern)* | China Telecom, Suzhou, China** 03/2020 – 03/2021

- Developed a website to collect & analyze telecom-related policy data from the government's website:
  - Constructed the framework of the website individually with *Vue.js* + *Django* + *MongoDB*;
  - Developed a Python-based crawler with full functionalities, involving 1) sending network requests using package *requests*, 2) parsing and cleaning HTML files using package *BeautifulSoup*; embedding the crawler interface into the back-end framework to form APIs.
  - Established Knowledge Graphs Databases by *Neo4j* to retrieve relevant information and predict policy directions.

RESEARCH EXPERIENCE

SHARP: Software Hint Augmented Memory Access Prediction for Graph Analytics
Research Assistant, University of Southern California, ***Advisor:*** *Viktor Prasanna* 01/2022 – 08/2022

- Proposed & developed SHARP, a workflow for optimizing memory access for graph processing using ML models.

- Introduced unique Scatter-Gather paradigm in graph processing tasks; designed memory access trace forecasting model based on transformer and attention-based techniques.
- Analyzed pattern differences of scatter and gather phases with respective 3-D scatter plots.
- Enhanced memory access performance by at least 3.2% compared to baseline models, such as Delta-LSTM;

Investigation of Self-supervised Learning Algorithm in Multi-modality Retrieval Robustness
Final Year Project, XJTLU, ***Advisor:*** *Kaizhu Huang* 09/2021 – 04/2022

- Deployed self-supervised learning model to enhance the image-to-text noisy retrieval performance.
  - Applied Contrastive Language Image Pre-training (CLIP) encoder with an extra text tokenier to transform dataset into tensors; Customized a network based on BERT and transformers to train the tensors with manual tuned hyperparameters.
  - Built a workflow to store, transform, and fed pretrained features into Multimodal Robust Learning (MRL) model.
  - Improved the retrieval accuracy and robustness of the model by 12.6% comparing to results of original MRL.

Urban Crime Trends Analysis and Occurrence Possibility Prediction based on Light Gradient Boosting Machine
Research Assistant, XJTLU, ***Advisor:*** *Gangmin Li* 11/2020 – 05/2021

- Explored an effective crime rate prediction model using ML models and Big Data Analytics.
  - Utilized various ML models like *Naive Bayes, Random Forest*, *XGBoost*, and *Light-GBM* to predict crime rates; evaluated respective prediction accuracy and chose the *LightGBM* algorithm as the prediction model.
  - Improved the training accuracy by grid-searching the most suitable hyper-parameter; Visualized and analyzed various crime features with heat maps; Achieved 98.30% prediction accuracy according to relevant information。
  - Published the results in the 2021 IEEE 4th International Conference on Big Data and Artificial Intelligence (BDAI).

PROJECTS & COMPETITIONS

An Automatic Waste Sorting and Collecting Cart | ***the 9th National Optoelectronic Design Competition*** 03/2021 – 08/2021

- Programed the control system based on *ROS* to control motors and process input information;
- Calculated the most appropriate path by using Dijkstra shortest path algorithm based on information radar sensors.
- Acquired the garbage classification model by modifying and training *VGG-19* model; used forward kinematics to develop various pick-up methods of robotic arm for different garbage types;

Multi-platform File Synchronization Protocol | ***Computer Network*** **Course Project** 09/2020 – 12/2020

- Designed and constructed a multi-PC large file synchronization protocol with encryption and fast speed.
- Utilized *TCP* and *Multithreading* to complete large file transmission and applied *Gzip* for file compression.
- Deployed *RSA* and *MD5* to enable encryption and integrity detection of file transmissions.
- Implemented transmission resumption in case of unexpected network failure by transferring file slices.

Restaurant Order System | ***C++ Programming and Software Engineering*** **Course Project** 03/2020 – 06/2020

- Designed, built, and tested the resource management system of the restaurant.
- Developed the back-end APIs with *C++* and *SQLite*; constructed the GUI menu with user-interactive modules in QT Creator and Designer; specified the database retrieval functions in *OSqlQuery.*