Lihan Hu

lihan-hu@uiowa.edu · (+1) 319-576-9307 · lihan hu

Research Interest

High Performance Computing, Compiler Optimization, Graph Learning, Sparse Training, Machine Learning, Image Processing

Education

The University of Iowa, Iowa, USA 2022 – Present

Ph.D. student in Computer Science **Advisor: Peng Jiang**

Hohai University, Jiangsu, China 2019 – 2022

Master student in Pattern Recognition and Intellgent System **Advisor: Lixin Han**

Nanjing Audit University, Jiangsu, China 2014 – 2018

B.S. in Computer Science and Technology

Professional Experience

The University of Iowa Iowa, America Sep. 2022 – Present

Research Assistant on High Performance Computing Director: Peng Jiang

The City University of Hong Kong Kowloon, Hong Kong Sep. 2020 – Aug. 2021

Research Assistant on Biomedical Imaging Processing Director: Hong Yan

Payegis Nanjing, China Apr. 2020 – Sep. 2020

R&D Intern

Huazhong Univerisity of Science and Technology Wuhan, China Oct. 2016 – Feb. 2018

Research Intern on Disk Failure Prediction Director: Ke Zhou

Skills

- Programming Languages: Python, CUDA, C++, JAVA, SQL
- Platform: MacOS, Linux, Windows
- Tools: PyTorch, TVM, TensorFlow, Numpy, Pandas, cuBLAS, CUTLASS, PyQt5, scikit-learn

Publications

[IPDPS'24] **Lihan Hu**, Jing Li, and Peng Jiang. *cuKE: An Efficient Code Generator for Score Function Computation in Knowledge Graph Embedding.* IEEE International Parallel and Distributed Processing Symposium
[NeurIPS'22] Peng Jiang, **Lihan Hu**, Shihui Song. *Exposing and Exploiting Fine-Grained Block Structures for Fast and Accurate Sparse Training.* Thirty-sixth Conference on Neural Information Processing Systems
[KES'20] **Lihan Hu**, Lixin Han, Zhenyuan Xu, Tianming Jiang, Huijun Qi. *A disk failure prediction method based on LSTM network due to its individual specificity.* International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Research Experience

Compiler System Optimization **Apr. 2023 – Present**

Brief introduction: Build a compilation system: CuKE, where users can input naive code for the given application, and our system will go through a series of optimizations and transformations based on the semantics of the users' input to output optimized code that has the same function but more efficient. Our generated code supports multiple different platforms and computation devices.

Graph Learning Performance Optimization Sep. 2022 – Present

Brief introduction: Analyze the issue of computation, warp schduling and synchronization overhead in current graph learning algorithm using GPU. Use Nvidia Nsight Systems and Nvidia Nsight Compute to profile the performance of CUDA code for better optimizing. Use inline PTX code to replace CUDA statement, which can balance the overhead of data movement. Use LDGSTS to pipeline, bypass L1 data movement and loop unrolling technique to improve the inter-warp and intra-warp GPU resource utilization and accelerate the computation of graph learning. Implement PyTorch extension and custom CUDA operators to improve the performance of graph learning.

Structured Dynamic Sparse Training Apr. 2022 – Present

Brief introduction: Study the fine-grained dynamic sparse training algorithm with shuffled blocks to reach the accuracy with non-structured sparse training algorithm. Study the effects for accuracy and overhead of different block size setting and take a tradeoff between accuracy and performance. Implement scalable and robust convolution computation by CUDA code to accelerate training process in PyTorch. Our method achieves similar accuracy with lower time consumption after training with several models include both image classification model and language model.

Cell Image Processing Sep. 2020 – Sep. 2021

Brief introduction: Designed and developed an automatic cell image segmentation and analysis tool for Caenorhabditis elegans cells image by using TensorFlow and PyQt5, reduced time for biologists to manually annotate cells. Improve the accuracy of segmentation based on U-Net and distance transformation, and implement Delaunay triangulation to find the boundary of each cell and their nucleus, which brings more analyzable data. Implement cell image denoise tool based on cycleGAN, solved the problem that too many noise points acquired by the optical imaging system lead to segmentation errors.

User and Entity Behaviour Analysis(UEBA) modeling tools Jun. 2020 – Jan. 2021

Brief introduction: Co-developed a big data middle station system based on Python package scikit-learn and tornado, which supports multiple formats of data input. The system performs basic statistical analysis and machine learning model building, and finally generate a visual user interface. Constructed User and Entity Behaviour Analysis(UEBA) modeling tools, capture users' input habits, kerstroke frequency, mouse tracks and sensor data in browsers or APPs, and determine whether the current user is a real user or a malicious script. Achieve similar results with Google reCAPTCHA in many different kinds of operating systems and platforms.

User and Entity Behaviour Analysis(UEBA) modeling tools Oct. 2016 – Feb. 2018

Brief introduction: Load BackBlaze Dataset through MySQL and write SQL scripts to get the S.M.A.R.T. data (more than 100 K drives with 100M entries). Use smartctl to record the disk usage and S.M.A.R.T. values in Tencent real production environment and compare the data between BackBlaze and Tencent data. Process and analyze these data and apply open source machine learning libraries (Python scikit-learn) to generate predictive models based on the data. Compared with traditional threshold methods, the results of model prediction achieve better accuracy. This project has been launched into the storage system of Tencent Cloud. Propose disk drive's individual specificity from different scenarios, and time series is also an important factor in failure prediction. Apply LSTM to improve the accuracy of prediction model.