Mathew Jacob

Champaign, USA mkjacob3@illinois.edu 2018199355 Mathew Jacob mjacob1002

EDUCATION

B.S Computer Science, *University of Illinois at Urbana Champaign* 2025
GPA: 3.91/4.0
Coursework: Discrete Structure, Data Structures, Computer System Organization, Prob. and Statistics, Combinatorics, Systems Programming, Algorithms & Models of Computation, Communication Networks

PROFESSIONAL EXPERIENCE

Research Scientist Intern, *Databricks Mosaic Research* 05/2024 – 10/2024
- Studied RAG pipelines and how they behave under different settings.
- Focused primarily on rerankers and empirically studied their behaviors in different settings so as to 1) better understand the deficits of the current SOTA and 2) to better inform reranker deployment, as well as training rerankers to mitigate this risk.
- Ran many experiments and explored different a swatch of properties and models, including latency-quality tradeoffs, architecture differences (late-interaction vs cross encoder), and more.
- Resulted in paper "Drowning in Documents: Consequences of Scaling Reranker Inference" that was well received by research community and top researchers from institutions i.e Google Deepmind

Software Engineer Intern, *Databricks* 05/2023 – 08/2023
San Francisco, United States
- ML Training Team
- Contributed to Apache Spark repo, adding a Deepspeed TorchDistirbutor to PySpark that enables easy running of deepspeed applications on Spark clusters. Shipped in Spark 3.5: https://www.databricks.com/blog/introducing-apache-sparktm-35 .
- Blog post about DeepspeedTorchDistributor: https://community.databricks.com/t5/technical-blog/introducing-the-deepspeed-distributor-on-databricks/ba-p/59641 .
- Enabled faster iteration through Spark Dataframe in Huggingface IterableDataset

Research Assistant, *Parallel Programming Laboratory* 09/2021 – present
Champaign, USA
- Led development of CkIO, a parallel I/O library written for Charm++ to be used on supercomputers
- Working on multi-producer multi-consumer abstractions for task-based systems to adapt existing frameworks (i.e Charm++) to other workloads, such as data processing on supercomputing clusters.

PERSONAL PROJECTS

GraphCPP, *Graph Library for Performance* 04/2022 – present
- Developed and improving library that contains ways to both make graphs and run common algorithms on them.
- Written in C++
- Created Python interface using pybind11
- Algorithms partially use SIMD(via XSIMD library) to get better performance.

Wordle Bot
- Leveraged Shannon's Entropy and expected value to optimize bot's guesses
- Written in C++ for performance

Illinois Space Society Simulator
- Developed logging system leveraging spdlog3 to write clear log files to expedite the debugging and evaluation of simulations.
- Wrote effective software for rocket launches and collected data from each launch for analysis.

COVID-19 Vaccination Strategy Simulation, *(High school)*
- Ran simulations to try and simulate how COVID-19 would spread through a community based on different vaccine rollout strategies.
- Used scikit-learn to run linear regressions on simulation data.
- Leveraged file asynchronization to speed up simulations by putting each process on a different processor.

- Wrote a manuscript on my findings that can be found in my GitHub repository.

RESEARCH PROJECTS

CkIO: Parallel File Input For Over-Decomposed Task-Based Systems, *Arxiv* 07/2024

- Described a system for over-decomposed systems (where the number of actors >> number of PEs) for doing high-performance I/O, with features such as object-migratability.
- Implemented this system in CkIO, a library based on Charm++. Involved profiling and improving performance of distributed programs on supercomputing clusters. Wrote high-performance C++ code.
- Paper worked on with PhD student Maya Taylor and Professor Sanjay Kale at the Parallel Programming Lab
- To be submitted to HPDC, preprint is on arxiv

Stream Semantics in Asynchronous Many Task Systems, *WIP* 2024

- Working on supporting multi-producer multi-consumer semantics for asynchronous many-task systems, easing programmability of data pipelines with overdecomposed frameworks.
- Built preliminary system and am responsible for profiling and optimizing the code.
- Conducted literature survey and needs of modern programs today
- Research conducted with Professor Sanjay Kale at the Parallel Programming Laboratory

Drowning in Documents: Consequences of Scaling Reranker Inference, *Arxiv*

- Explores the behavior of state-of-the-art rerankers that are deployed in enterprise, focusing specifically on cross encoders. Discovered interesting trends that emerge in reranker quality as we scale different variables, such as documents retrieved. It shows that there is still a lot of headroom for modern rerankers, with them sometimes not outperforming dense embedding performance.
- Challenged longstanding assumption in information retrieval and spurred new research
- Used by Databricks Vector Search product and guides new research on optimizing rerankers.
- Worked with Andrew Drozdov, Omar Khattab, Professor Michael Carbin, and Professor Matei Zaharia at Databricks Mosaic Research
- Well received by community and acknowledged by researchers at top labs i.e Google Deepmind.

SKILLS

Languages
C++, Python, Java, Julia

Build Systems
CMake, Ninja, Make

Misc
AWS EC2, GitHub, SLURM, MPI

Libraries/Frameworks
xsimd, pybind11, numpy, pandas, pytorch