

# Kaleb Dickerson

(512) 801-1561 | [kaleb.stormblessed@gmail.com](mailto:kaleb.stormblessed@gmail.com) | [linkedin.com/in/kaleb-dickerson2023](https://linkedin.com/in/kaleb-dickerson2023) | [github.com/Prometheus1400](https://github.com/Prometheus1400)

## EXPERIENCE

---

### Snowflake

Jul 2025 – Present

*Software Engineer – Machine Learning Platform | Menlo Park, CA*

- Enabled inference directly within Snowflake Dynamic Tables by adding function volatility support to snowflake-ml-python
- Designed an extensible tool-handling framework and implemented Cortex Agent and generic tools, contributing to the launch of the Snowflake MCP server

### Apple

May 2023 – Jul 2025

*Software Engineer – Applied Machine Learning | Austin, TX*

- Architected and independently developed a critical new data ingestion service on a modern stack (JDK 21, Spring Boot, MongoDB, Kafka, Redis), replacing a legacy solution to significantly improve developer accessibility, enhance platform stability via local integration tests, and decrease feature development turnaround time
- Increased generative AI platform accuracy by 7% via semantic chunking during data ingestion, enabling more context-aware retrieval
- Architected custom Flink connectors for the internal Generative AI ingestion platform that facilitated data integration; over 40 teams across Apple now utilize the solution
- Reduced resource consumption by 50% while delivering critical order updates to millions of customers by designing and integrating a custom Kubernetes CRD and Operator for batch job parallelism

### Texas A&M University

Jun 2021 – Sep 2021

*Research Assistant - Graph Neural Networks | College Station, TX*

- Reviewed academic literature and designed networks to excel at tasks such as 3D geometry prediction for molecules
- Processed datasets of up to 120M molecules using PyTorch-based deep learning models

## EDUCATION

---

**Texas A&M University** — Bachelor of Science in Computer Science, Minor in Mathematics

May 2023 | GPA: 3.95

## PROJECTS

---

### radiant | Go, LLVM

Turing complete, strongly typed, compiled programming language with modern syntax and type inference supporting simple data types and functions. Implemented the compiler frontend in Go (tokenizer, Pratt parser, LLVM IR generator) based on "Crafting Interpreters," enabling optimization and compilation via Clang.

### fuzzy-finder | Rust, Nix

Terminal-based fuzzy finder written from scratch in Rust using the Crossterm library for text-based UI. Built with real-time multi-selection and asynchronous non-blocking rendering.

### linux-shell | C++

Linux shell that interacts with the OS using the POSIX standard. Supports process execution, piping, I/O redirection, background processes, \$-sign expansion, and custom signal handling.

**markdown-latex-render.nvim** | [Neovim](#), [Lua](#)

Neovim plugin to render LaTeX expressions inside of markdown files by parsing injected grammars via Treesitter and displaying generated images inline.