

Linus Sheaff

☎ +44 7519932219 • ✉ sheaff.linus@gmail.com • 📄 github.com/linussheaff

Education

BSc Hons. Computer Science and Mathematics, *University of Edinburgh* (2022–2026)
1st Class (75%). Dissertation: *Data-Adaptive Kernel Selection for Sparse Tensor Compilers*. Supervisor: Jackson Woodruff.

Research Interests

Compilers for ML and accelerators: scheduling languages and code generation for accelerators; sparse and irregular computation; inference on constrained hardware (memory hierarchy, quantisation, heterogeneous targets); energy as an explicit optimisation objective; ML-guided compiler optimisation.

Publications & Presentations

Poster on dissertation work accepted at REACH, Nov 2026. **Full paper** in preparation, targeting CC 2027.

Research Projects

Data-Adaptive Kernel Selection for Sparse Tensor Compilers (C/C++, Rust) Sep 25 – Aug 26
Undergraduate dissertation, extended for publication. Supervised by Jackson Woodruff.

- Built a compiler emitting optimised C for sparse tensor computation, choosing workspace strategy and kernel implementation from the shape of the input data rather than one fixed strategy. Geometric mean speedup of **2.02x** over existing sparse tensor compilers.
- Extended this to dynamic multi-versioning: the compiler emits several specialised kernel variants and dispatches between them at runtime on the actual input, so shapes unknown at compile time still reach a well-matched implementation.
- Drove these decisions with a lightweight heuristic model over tensor shape, within **1.01x** of an oracle selecting the fastest variant and **3.55x** faster on average than a fixed-strategy compiler.
- Wrote the IR static analysis (dataflow, loop dependence) establishing which transformations are legal, with an IR generalising across source languages (TACO, a PyTorch subset); used perf counters to explain *why* variants won or lost.

Energy Profiler for HPC and AI Workloads (Rust, Python, C) Dec 25 – Jan 26
Independent research tool. Repo Link

- Per-component (CPU, GPU, DRAM) power profiler over RAPL and NVML, exposed to Python via PyO3, with asynchronous sampling kept off the critical path.

Source-Grounded Constrained Decoding (Python) Jan 26 – Apr 26
Independent NLP research project. Repo Link

- HuggingFace LogitsProcessor constraining a 1B-parameter model to a source-derived vocabulary; soft penalties reduced hallucinated entities across 200 XSum examples.

Teaching

Undergraduate Supervisor, *University of Cambridge* From Oct 2026
Supervising first- and second-year undergraduates in *Discrete Mathematics*, *Logic and Proof*, *Introduction to Probability* and *Compiler Construction*, including setting and marking work.

Course Tutor and Marker, *University of Edinburgh, School of Informatics* 2023 – 2026
Ran weekly small-group tutorials for *Introduction to Functional Programming*, *Introduction to Object-Oriented Programming*, *Introduction to Algorithms and Data Structures* and *Fundamentals of Data Science*; marked coursework to departmental rubrics.

Private Tutor, *Gravitas Tutors and private clients* Sep 2020 – Present
One-to-one Mathematics, Computer Science and Physics tuition, GCSE to undergraduate.

Technical Skills

Languages: C/C++, Rust, Python, Haskell. **Tooling:** PyTorch, HuggingFace, Git, GitHub Actions CI.

Compilers: IR design and lowering; dataflow, control-flow and loop dependence analysis; loop fusion, kernel selection and runtime multi-versioning; learned cost models; parser generators (tree-sitter, pest, bison).

Performance: Linux perf; energy measurement via RAPL and NVML; benchmark design.