Sebastian Griego

sebastianngriego@gmail.com — GitHub: sebastian-griego — San Diego, CA

Summary

AI for math focused applied mathematics M.S. student building research software for Lean 4 statement autoformalization, learned selection and reranking, and equivalence-centric evaluation. Experience training and evaluating transformer cross-encoders, maintaining reproducible Lean toolchains, and shipping end-to-end experiment pipelines.

Education

San Diego State University — Expected May 2026
M.S. Applied Mathematics (HSF Scholar) — *San Diego, CA*

Pepperdine University — May 2024
B.S. Mathematics, Minors: Data Science, Classics (Regent's Scholar) — *Malibu, CA*

AI for Math Research Software

Independent Projects (Lean 4, Autoformalization) — 2025 to Present
Selection, evaluation, and repair tooling for formal math pipelines — *Remote*

- **NLVerifier**: natural language conditioned cross-encoder selector for Lean autoformalization candidates after type-check filtering. Improved ProofNetVerif top-1 selection from 84/178 (47.2%) to 111/178 (62.4%) on the original split, and from 17/55 (30.9%) to 27/55 (49.1%) on an ID-disjoint split.
- **BEqCritic**: learned alternative to Self-BLEU selection by scoring candidate-candidate equivalence with a cross-encoder, building a similarity graph, clustering, and selecting a representative. End-to-end pipeline with Makefile quickstart and optional multi-GPU DDP via `torchrun`.
- **SAF V0**: statement autoformalization fidelity harness to accept or reject a candidate Lean statement without requiring a proof, using type-check plus deterministic normalization, with optional audited semantic rewrites and optional proof-based equivalence checks. Frozen Lean toolchain and pinned Mathlib commit for reproducible runs.
- **Additional Lean tooling**: **LeanRepair** (iterative repair loop for malformed Lean declarations by parsing compiler errors and applying heuristic fixes), **autoformalize** (Kimina scripts for generation and evaluation with controlled decoding and summary reporting), **milean** (probe and steer a ByT5 Lean tactic generator on `intro` vs `apply` using teacher-forced scoring, probes, and activation interventions).

Research Experience

Disease Modeling Lab, San Diego State University — Aug 2024 to Present
Research Assistant — *San Diego, CA*

- Built physics-informed neural networks for HIV viral dynamics, including coupled ODE solvers, biologically constrained training, and ablation studies.

Cajal — 2026
Research Intern, Autoformalization Benchmarking (Lean 4) — *Remote*

- Cleaning and packaging a novel autoformalization benchmark with a two-part unit-test evaluation structure.
- Building an evaluation harness to test frontier models and report results.

Publications and Presentations

- NLVerifier: Natural Language Conditioned Cross-Encoder Selection for Lean Autoformalization Candidates. Submitted to ICLR 2026 VerifAI Workshop (under review).
- Math model-informed neural networks for patient-specific HIV latent reservoir dynamics under ART. Contributed Talk, Joint Mathematics Meetings (JMM) 2026, Washington, D.C., Jan 2026.
- Mathematical modeling and machine learning to predict the dynamics of HIV latently infected cells under antiretroviral therapy. Poster, SoCal SysBio 2025, University of California, Riverside, May 2025.

Open-Source Software

- BetterFFTW (pyFFTW wrapper, drop-in NumPy FFT replacement), PyContinuum (numerical homotopy continuation), Mazewright (maze generation toolkit).

Teaching

Teaching Assistant, San Diego State University (Calculus I to III) — Aug 2024 to Present
Teaching Assistant, Euler Circle (Abstract Algebra, Real Analysis) — Jan 2025 to Present
Resident Counselor and Co-instructor, Stanford SUMaC (Abstract Algebra, Number Theory) — Summers 2024 and 2025

Honors

HSF Scholar — 2024 to 2026
Regent's Scholar (Pepperdine) — 2020 to 2024

Technical Skills

Programming: Python, Lean 4 (including metaprogramming), SQL, R
Core areas: transformer-based reranking and evaluation, reproducible experiment pipelines, Lean toolchains and automated checking
Tools: Git, Linux, Make, LaTeX