

# 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, selection and reranking, and equivalence-centric evaluation. Experience fine-tuning transformer cross-encoders, running GPU training, and maintaining reproducible pipelines with pinned toolchains, scripts, and documentation.

## Education

### San Diego State University

*M.S. Applied Mathematics (HSF Scholar)*

Expected May 2026

### Pepperdine University

*B.S. Mathematics, Minors: Data Science, Classics (Regent's Scholar)*

San Diego, CA

May 2024

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 test 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 (train, candidate grouping, select, evaluate, report), single-GPU training and multi-GPU DDP via `torchrun`.
- **SAF V0:** statement autoformalization fidelity harness to accept or reject a candidate Lean statement without proving it, using type-check plus deterministic normalization and string equality, 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 theorem declarations by parsing compiler errors and applying heuristic or optional LLM-based fixes), **autoformalize** (Kimina scripts including a GenLM-controlled pipeline with well-typed and cycle-consistency potentials plus BEq+ evaluation using 7B models), **milean** (probe and steer a ByT5 Lean tactic generator on `intro` vs `apply` using teacher-forced scoring, layerwise probes, and activation addition).

## 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.

## 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.

## 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

**ML and systems:** transformer cross-encoders, GPU training, multi-GPU `torchrun DDP`, evaluation harnesses, reproducible experiment pipelines

**Tools:** Git, Linux, Make, L<sup>A</sup>T<sub>E</sub>X