NGF Benchmarking Baseline — Handoff

Environment

- Model: GPT-2 Medium (24 layers) - Tap: –9 → maps to layer index 15 - Frameworks: - torch 2.x - transformers 4.30+ - datasets 2.x

Scripts

- Harness: basic_script_ngf_v8.py - Supports --mode stock|ngf - Correctly scores P(ending | context) with cross-entropy over ending tokens only. - Robust truncation, attention masks, duplication-safe prefix. - JSON summary + progress meter. - Debug flags: --debug (per-item dumps), --debug_counts (token counts). - Hook adapter: ngf_hooks.py - Provides attach_ngf_hooks(model, ...). - Currently a NO-OP (hidden states passed through unchanged). - Prints layer index + cfg when attached. - Ready to receive warp → detect → denoise logic.

Default NGF Config (v4b T4/L4 geo)

tap = -9 alpha0 = 0.05 alpha_min = 0.006 trend_tau = 0.35 k_tr = 12 use_detect = 1 detect_width = 24 detect_sigma = 5.0 null_K = 32 null_q = 0.92 k_det = 7 s_latch = 0.30 linger = 2 ema_center_beta = 0.05 gen_mode = "geo"

Baseline Results

Run: python3 basic_script_ngf_v8.py --mode stock --model gpt2-medium --split validation --n 200 --max_length 768 --device auto python3 basic_script_ngf_v8.py --mode ngf --ngf_import ngf_hooks:attach_ngf_hooks --model gpt2-medium --split validation --n 200 --max_length 768 --device auto - Accuracy (200 val ex, CPU run): ~0.23–0.27 (near chance, as expected for GPT-2 medium). - Stock vs NGF (NO-OP) behave identically — this is the baseline.

Next Steps

1. Seed control: add deterministic seeding for A/B comparability. 2. Metrics logging: extend JSON output (accuracy, F1, hallucination, etc.). 3. Hook body: implement NGF warp/detect/denoise inside ngf_hooks.attach_ngf_hooks. 4. Scale runs: 1k–5k HellaSwag items for lower variance. 5. Additional tasks: replicate harness for PIQA, LAMBADA, etc. ■ This document establishes the baseline harness and results. From here, all NGF modifications should show lift relative to this control.