

# NGF Stage-12 Deployment Recipe (Full Detail)

This is the full 8-step turnkey plan for Stage-12 NGF deployment, preserved verbatim with all details, scripts, and command examples included.

## Step 1. Find the right tap (fast scan + EDA)

File: stage11\_llm\_layer\_scan\_plus.py

Command example:

```
python3 stage11_llm_layer_scan_plus.py --model --tap_range -12:-6 --calib
calib/calib_prompts_v2_900.txt --eval calib/calib_eval_style_200.txt --pool_mode lastk --k_last 12
--sigma_px 5.0 --density_floor 4.0 --min_prom 0.55 --with_detect --with_denoise --out_csv
logs/scan_.csv --out_json logs/scan_.json
```

Pick tap with low Phantom Index, high margin, stable trend.

## Step 2. Lock a stable config

Files: ngf\_hooks\_v2.py, ngf\_benchmark.py

Defaults (v4b): alpha0=0.05, alpha\_min=0.006, trend\_tau=0.35, k\_tr=12, detect\_width=24, detect\_sigma=5, null\_K=32, null\_q=0.92, k\_det=7, s\_latch=0.30, linger=2, ema\_center\_beta=0.05.

Adjust only slightly if EDA plots suggest multi-lobes.

## Step 3. Canary benchmark

File: ngf\_benchmark.py

Stock baseline:

```
python3 ngf_benchmark.py --mode stock --model --split validation --n 1000 --device auto --out_json
results/_stock_n1000.json
```

NGF full (warp→detect→denoise):

```
python3 ngf_benchmark.py --mode ngf --ngf_import ngf_hooks_v2:attach_ngf_hooks --model --split
validation --n 1000 --max_length 768 --device auto --tap --alpha0 0.05 --alpha_min 0.006 --trend_tau
0.35 --k_tr 12 --use_detect 1 --detect_width 24 --detect_sigma 5 --null_K 32 --null_q 0.92 --k_det 7
--s_latch 0.30 --linger 2 --ema_center_beta 0.05 --gen_mode geo --out_json
results/_ngf_v4b_n1000.json
```

Check: Macro-F1 and Acc ↑, calibration (ECE, Brier) improved or stable.

## Step 4. Transfer check

Files: ngf\_benchmark.py, stage11\_benchmark\_latest.py

Run on ARC-E/C, PIQA, BoolQ, TruthfulQA. No retune unless regressions appear.

## Step 5. Visualization & diagnostics

File: plot\_tap9\_contour\_well.py

Command example:

```
python3 plot_tap9_contour_well.py --pre results//tap-_pre.npy --post results//tap-_post.npy --out_png
results//tap_well_compare.png --out_pdf results//tap_well_compare.pdf --fit_on post --sample 80000
--bins 220 --sigma 2.0 --clip_q 0.01 --levels 14
```

Aim: ragged multi-lobe (pre) → single smooth funnel (post).

## Step 6. A/B telemetry (deep traces)

File: stage11\_ab\_eval\_base\_denoise.py

Command example:

```
python3 stage11_ab_eval_base_denoise.py --model --layer --prompts prompts/sanity_200.txt
--max_new_tokens 96 --alpha0 0.05 --alpha_min 0.006 --trend_tau 0.35 --k_tr 12 --use_detect 1
--detect_width 24 --detect_sigma 5 --null_K 32 --null_q 0.92 --k_det 7 --s_latch 0.30 --linger 2
--ema_center_beta 0.05 --gen_mode geo --print_every 128 --device auto --out_json logs/ab__.json
```

## Step 7. Scale up

Files: ngf\_benchmark.py, stage11\_ab\_eval\_base\_denoise.py

Run deeper benchmarks: MMLU, GSM8K, WinoGrande. Keep OutlierGuard on for long prompts.

## Step 8. Package artifacts

Artifacts per model:

- config\_locked.json (tap + params)
- metrics\_stock.json, metrics\_ngf.json
- well\_pre\_post.png
- scan\_summary.json (tap metrics)