Deterministic Local Topology Capture for Gemma 4 31B-IT

VertRule technical note

2026-04-10 · v1 · Gemma 4 31B-IT · CPU

1. Claim

Gemma 4 31B-IT runs on the local CPU path on real weights. Hidden-state and topology capture are non-perturbing against a frozen compute baseline. Canonical `topology.v2` artifacts are emitted under a binary admission rule and reproduce bit-for-bit across independent capture runs. The note records the measurement contract, the frozen anchors that define admissibility, the execution and artifact results against that contract, and one bounded observation extracted from the artifact as an illustration that the surface is analyzable. It is not a research discovery, not a broad interpretability claim, and not a product statement.

2. Contract

The claims in this note are bounded by the following execution and artifact contract. Any reproduction that changes this contract is not a test of the note.

- Model. Gemma 4 31B-IT. Real weights, unmodified.
- Execution surface. CPU only. Digests are locked within a CPU architecture family; replication on a different CPU architecture is not claimed.
- Bounded setups.
  - *Single-token.* Forward pass on one token. Frozen non-perturbation baseline.
  - *Two-token prefill.* Fixed two-token sequence. Bounded setup for attention-edge capture.
- Admission rule. Binary. A topology emission is admissible only when every invariant below holds simultaneously:
  1. Schema version locked to `topology.v2`.
  2. Observed upstream logits digest matches the frozen anchor.
  3. Logits digest reproducible across two independent capture runs.
  4. Node, edge, layer, and step counts match the expected cardinalities exactly.
  5. `nid` contiguous `0` .. `node_count`.
  6. Node ordering monotonic non-decreasing in `(step, layer_ix)`.
  7. Edge ordering canonical lex `(layer_ix, src, dst)`.
  8. Edges bit-identical between the two capture runs.
  9. Stats counters match observed collection lengths.
  10. Real-activations validator passes.
  11. Redacted public form contains no `"token_text":` or `"response_text":` key.
  12. Canonical `TopoRun` digest reproducible across the two capture runs.
- Frozen anchors. Table 1. Any drift flips the admission rule. The emission becomes inadmissible under this contract.

Table 1. Frozen anchors and artifact counts.

Surface	Value
Model tensor-inventory digest	`ed5cf6b1e99a7eb1afeb48f1a8c70a7a5e00d74dc72952facb59e22cc69c625a`
Single-token frozen logits digest	`af9594478c6ee7ed12ec8766ecf1958ef0bd63688fbbd73cc7cf18728cd5e42e`
Two-token frozen logits digest	`4d93482cf9c2fe00af690ff1ad0aa8fa33aa0ccc28192f055976c869b2b94af1`
Phase 1 canonical `TopoRun` digest	`2fbfe64a2c81b1c9d10cf2b60a9e50632d8588f7aec8aaf8cc9ee067641f7aa7`
C2 canonical `TopoRun` digest	`6c864bd883fbb40e5a3709f2d5fefb7f9f7dac826c0d8c4fefde748cd8a090f4`
Phase 1 artifact cardinality	60 nodes / 0 edges
C2 artifact cardinality	120 nodes / 120 edges

The tensor-inventory digest is BLAKE3 over the sorted tensor names from the safetensors index files; it pins the tensor-name layout and shard structure, not the tensor values. The weight values are pinned by the frozen logits digests below it. Both digests must match for a reproduction to be considered on the same weight set.

3. Execution Surface

Local CPU path. Real weights. The single-token forward produces logits whose digest is the frozen anchor `af9594` … `8cd5e42e`. This digest is the non-perturbation baseline.

Hidden-state capture is non-perturbing under every capture set tested. Three capture variants were run against the same forward path in the same admitted run: full (all 60 layers), empty, and sparse (layers `{0, 29, 59}`). All three produced the frozen logits digest exactly.

Reproducibility is bit-stable at both the compute layer and the canonical artifact layer. The two-token prefill is captured twice per admitted run. Both independent captures produce logits digest `4d93482c` … `b94af1`. Both independent topology builds produce canonical `TopoRun` digest `6c864bd8` … `d8a090f4`.

Cardinalities are exact, not approximated. The Phase 1 artifact contains 60 nodes and 0 edges. The C2 artifact contains 120 nodes and 120 edges. The admission rule checks these as hard invariants.

4. Canonical Artifact

The canonical artifact is a `TopoRun` under the `topology.v2` schema.

- IEEE-754 floats are stored as explicit bit patterns. NaN payloads are canonicalized to a single quiet NaN bit pattern. The serialized artifact carries no platform-dependent NaN bits.
- Canonical JSON is produced via RFC 8785 (JCS) canonicalization. The artifact digest is BLAKE3 over the canonical byte sequence. Within the stated execution surface, the same admitted input produces the same digest across repeated runs. Canonicalization of the emitted payload is deterministic by construction.
- Concept scores are reduced in `f64` with `f32` bit-pattern storage, sorted by concept identifier in lex order.
- Bounded attention edges are emitted via deterministic top-k selection with key-position-ascending tie-break, then sorted lex by `(layer_ix, src, dst)`. The edge sequence is byte-stable across runs.
- Redaction is part of the admission rule. Before the public form of the artifact is accepted, it is redacted and the redacted JSON is checked for any residual plaintext key. The check must pass. This is a shape contract on the public artifact, not a security claim about the raw capture path.

Canonical assembly consumes abstract layer-boundary captures and attention probes. Model-specific code is a capture client. It is not part of the canonical boundary.

This note claims deterministic reproduction on the admitted local CPU surface and deterministic canonicalization of the emitted payload; it does not claim arbitrary cross-thread or cross-architecture floating-

point equivalence.

5. Bounded Observation

Included as one bounded example that the canonical artifact contains analyzable structure.

Per layer, we extract the head-averaged softmax weight at query position 1 over the two key positions {`BOS`, `self`} from the C2 two-token prefill. Query position 0 is excluded because the causal mask admits only one key there, making the attention distribution trivial. The reported quantity is $\delta_l = p_{\text{self},l} - p_{\text{bos},l}$. Gemma 4 31B-IT has a fixed 5:1 alternation between sliding and full attention layers, with the ten full-attention layers at depths $\{5, 11, 17, 23, 29, 35, 41, 47, 53, 59\}$. The observation is restricted to those ten layers.

Three bounded two-token prefills were captured: `[BOS, 1]`, `[BOS, 2]`, `[BOS, 3]`, with `BOS = 2` fixed at position 0. The canonical admitted two-token surface in this note is `[BOS, 1]`; `[BOS, 2]` and `[BOS, 3]` are auxiliary perturbation checks evaluated through the same read-only extraction math.

Robust claim: ordering and sign. In all three variants, $|\delta_{59}|$ strictly exceeds $|\delta_l|$ for every other full-attention layer, and $\delta_{59} > 0$. The final full-attention layer at depth 59 is the most polarized full-attention layer in every variant tested, and it is always self-leaning.

Effect size is token-dependent and is not part of the robust claim. The ratio between $|\delta_{59}|$ and the maximum $|\delta|$ of the other nine full-attention layers is $3.19\times$ for `[BOS, 1]`, $1.68\times$ for `[BOS, 2]`, and $1.05\times$ for `[BOS, 3]`. The separation varies with the choice of second token. Ordering and sign survive all three variants. Magnitude does not.

A broader sliding-layer trend observed in one variant does not survive the perturbation check and is excluded from the observation.

No mechanism is offered.

6. Boundary

These measurements are specific to the admitted local CPU surface and the bounded single-token and two-token setups defined above. They do not establish cross-device parity (including browser, WebGPU, Metal, or other accelerated-backend surfaces), broad mechanistic-interpretability or causal claims about Gemma 4 31B-IT internals, long-context or multi-token generalization beyond those setups, or claims about other Gemma sizes, other Gemma versions, or other model families. The bounded observation in Section 5 is included as an illustration that the artifact contains analyzable structure, not as a standalone mechanistic claim about Gemma 4.

7. Verification

A reader with access to Gemma 4 31B-IT weights can reproduce or falsify the claims against the contract.

1. Frozen anchors. The four forward-pass digests in Table 1 (single-token logits, two-token logits, Phase 1 canonical `TopoRun`, C2 canonical `TopoRun`) are reproducible under the stated single-token and two-token bounded setups. The tensor-inventory digest is verified separately by hashing the sorted tensor names from the safetensors index. Any drift in any anchor is itself a falsification of the reproducibility claim and indicates the contract is not being honored exactly.
2. Cardinalities. The exact counts (`60 / 0`, `120 / 120`) are admission invariants and are checked directly on the emitted canonical artifact.
3. Bounded observation. For each of the three two-token prefills, extract the head-averaged softmax row for the query at position 1 over the two key positions, compute $\delta_l = p_{\text{self},l} - p_{\text{bos},l}$ at every layer, restrict to the ten full-attention layers, and verify that $|\delta_{59}|$ is the maximum and $\delta_{59} > 0$. The

observation is falsified by any variant in which a full-attention layer at depth less than 59 exceeds depth 59 in $|\delta|$, or in which $\delta_{59} \leq 0$.

4. Reduction rules. `f64` accumulation throughout the extraction path. `f32` bit-pattern storage at canonical boundaries. NaN payloads canonicalized to a single quiet NaN bit pattern.
5. Admission rule is binary. A reproduction that fails any single invariant is not a partial reproduction. It is an inadmissible measurement under the same contract. "Mostly deterministic" is not a usable state.

The measurement rule is stated explicitly enough that an independent implementation, given the same weights and the same bounded setups, should be able to reproduce or falsify the claim without relying on internal VertRule implementation details.