

Test Persona Claims

Gavin · AI-assisted course design

Contents

1	Goal	3
2	Part 1 — Audit the evidence before measuring	3
3	Part 2 — Complete the arithmetic warm-up	4
4	Part 3 — Lock the specimen and runtime	4
5	Part 4 — Save the exact fixture families	5
5.1	Two inexpensive shortcut controls	6
6	Part 5 — Validate observation and freeze the plan	6
6.1	One direction and one output metric	7
7	Part 6 — Run the held-out families once	8
8	Part 7 — Analyze generalization without hiding dependence	8
9	Part 8 — Optional frozen intervention extension	9
10	Part 9 — Preserve the result and write the claim	10
11	More Learning	11
12	Optional implementation files	11

1 Goal

Audit claims about emotion and persona, then test how far a small contextual contrast generalizes. You will compare internal measurements with a fixed output score on exact prompt families, including shared-word entity-binding controls. A null or mixed result is a useful completed Lab.

Prerequisites: Lesson 5.4, observation validation from Lab 17, and the existing Pythia-14M cache. The optional intervention uses the hook discipline from Lab 19.

Time: 90–150 minutes, depending on how much of the existing runner you reuse.

Requirements: the existing CPU environment and checkpoint only. No new weights, inference API, paid compute, GPU, training, or external model judge. The prescribed core uses 52 attempted forwards; the predeclared optional extension brings the total to 76. The hard cap is 80.

Expected artifact: an evidence audit, a frozen experiment plan, a reproducible runner, complete measurements, and a restrained interpretation. This is an implementation specification, not an already-executed notebook or a set of observed model results.

Pythia-14M is a small base language model. This Lab does not presume that it follows assistant persona instructions, accurately infers emotional states, or reproduces published results from much larger instruction-tuned models. We use sentence continuation to inspect a narrower capability.

2 Part 1 — Audit the evidence before measuring

Read the method and limitations of Persona Vectors, v3, and the abstract plus relevant sections of Emotion Concepts and their Function in a Large Language Model, v1. The authors’ web paper is an alternative reading surface; record which version you used.

Create two records, one per paper, with these fields:

- Exact model and whether it is base or instruction-tuned
- Operational definition of the trait or emotion concept
- How examples and labels were obtained
- Which activation boundary and token positions were measured
- How candidate directions were chosen
- What data were reserved for evaluation
- Whether the result is descriptive, predictive, or interventional
- One alternative explanation and one reported limitation
- A source section or figure that supports your record
- One sentence that the evidence supports, and one stronger sentence it does not

Do not fill unknown fields by analogy with the course experiment. Write “not established in the section inspected” and identify the additional material needed.

Then classify these claims as supported by the described evidence, requiring additional evidence, or beyond the scope of these experiments:

1. A direction derived in one named model influenced a measured output under the reported intervention.
2. Any small language model has the same direction and behaves as an emotional assistant.

3. A readable emotion label proves subjective experience.
4. An activation detector that separates two prompt types necessarily predicts subtle differences within each type.

Return to these classifications after running the Lab. Your local measurements cannot retroactively establish a result in a different model.

3 Part 2 — Complete the arithmetic warm-up

Use the discovery vectors and three held-out contrasts in Lesson 5.4. Compute the group means, direction, unit direction, signed projections, cosines, mean, and sample variance. Include the disagreeing example in your explanation.

Answer: if you chose a direction from the average discovery contrast, why is a positive average discovery projection not independent validation? Keep these hand-designed numbers separate from every trained-model measurement.

4 Part 3 — Lock the specimen and runtime

Reuse EleutherAI/pythia-14m at revision 94f7c35d5e9f2e9bac8ca839329f505b4d007d5d. The approved cache contains config.json, tokenizer.json, tokenizer_config.json, special_tokens_map.json, generation_config.json, and model.safetensors. The safetensors file is 28,143,920 bytes; the complete six-file artifact is approximately 30 MB. Missing or mismatched files are a setup blocker, not permission to download a replacement model. Pinned publisher revision

Record SHA-256 for all six files and compare against the previously verified manifest. Read the local configuration and assert six layers, residual width 128, and parallel residuals. Use the existing versions: Python 3.13.13, PyTorch 2.9.0, Transformers 4.57.1, tokenizers 0.22.1, safetensors 0.6.2, and huggingface-hub 0.35.3. Record complete build strings, operating system, and processor. Stop the confirmatory run on a mismatch; do not silently call a different environment equivalent.

Set HF_HUB_OFFLINE=1 and TRANSFORMERS_OFFLINE=1 before loading. Load the local directory with local_files_only=True, trust_remote_code=False, and safetensors enabled, using the built-in GPTNeoXForCausalLM and fast tokenizer. Use CPU float32, eager attention, one CPU thread, model.eval(), torch.inference_mode(), seed 20, and deterministic algorithms. Set use_cache=False, output_attentions=False, output_hidden_states=False, and logits_to_keep=1 explicitly on each pass. Assert final logits have shape (1, 1, vocab_size); use that one retained position consistently.

Require finite values in captured activations, logits, replacement vectors, and every defined numerical metric. A nonfinite value is a failed measurement check, not a near-zero result. Deliberately undefined quantities use JSON null and an explanatory status, never NaN.

Apply a cumulative **15-minute computation cap** across discovery and evaluation, excluding the human pause between phases. Persist active elapsed seconds using a monotonic clock and enforce the remaining allowance with a supervising watchdog. On expiry, stop computation, preserve partial records, and record time_budget_exceeded; do not label the run complete or silently restart the clock.

Every attempted forward, including a deliberately aborted cleanup test, increments a persistent counter. Refuse an eighty-first attempt. Do not generate continuations, search layers, optimize parameters, retry prompts for attractive outputs, or execute exploratory sweeps inside this run.

5 Part 4 — Save the exact fixture families

These are original synthetic fixtures. Append the exact suffix `\nAlex felt` to every context. In JSON, `\n` represents one newline. Use contexts exactly as written.

```
{
  "suffix": "\nAlex felt",
  "score_candidates": [" happy", " sad"],
  "discovery": [
    {"id": "D1+", "context": "Story: Alex was cheerful after reading the letter."},
    {"id": "D1-", "context": "Story: Alex was miserable after reading the letter."},
    {"id": "D2+", "context": "Story: Alex was delighted after hearing the news."},
    {"id": "D2-", "context": "Story: Alex was gloomy after hearing the news."},
    {"id": "D3+", "context": "Story: Alex was pleased after opening the package."},
    {"id": "D3-", "context": "Story: Alex was upset after opening the package."}
  ],
  "heldout_outcomes": [
    {"id": "H1+", "context": "Story: Alex searched for a missing dog. The dog came home alive."},
    {"id": "H1-", "context": "Story: Alex searched for a missing dog. The dog was found dead."},
    {"id": "H2+", "context": "Story: Alex entered a contest. The judges awarded Alex first place."},
    {"id": "H2-", "context": "Story: Alex entered a contest. The judges removed Alex from the contest."},
    {"id": "H3+", "context": "Story: Alex saved a file after months of work. The file was recovered from a backup."},
    {"id": "H3-", "context": "Story: Alex saved a file after months of work. The file was erased without a backup."}
  ],
  "heldout_binding": [
    {"id": "B1+", "context": "Story: Alex won the prize. Robin lost the prize."},
    {"id": "B1-", "context": "Story: Robin won the prize. Alex lost the prize."},
    {"id": "B2+", "context": "Story: Robin lost the prize. Alex won the prize."},
    {"id": "B2-", "context": "Story: Alex lost the prize. Robin won the prize."},
    {"id": "B3+", "context": "Story: Alex received a gift. Robin received a fine."},
    {"id": "B3-", "context": "Story: Robin received a gift. Alex received a fine."},
    {"id": "B4+", "context": "Story: Robin received a fine. Alex received a gift."},
    {"id": "B4-", "context": "Story: Alex received a fine. Robin received a gift."}
  ],
  "heldout_lexical_controls": [
    {"id": "C1+", "context": "Story: The glossary contained the word cheerful. Alex copied the entry."},
    {"id": "C1-", "context": "Story: The glossary contained the word miserable. Alex copied the entry."},
  ]
}
```

```

    {"id": "C2+", "context": "Story: The glossary contained the word delighted. Alex
copied the entry."},
    {"id": "C2-", "context": "Story: The glossary contained the word gloomy. Alex
copied the entry."}
  ]
}

```

For D, H, and B, plus/minus denotes the intended favorable/unfavorable context for Alex. The annotation is an instructional expectation, not a uniquely determined human emotion or verified model understanding. For C, plus/minus denotes only the polarity of the irrelevant glossary word; Alex's emotion is unspecified.

Use no chat template, padding, truncation, or added beginning-of-sequence token. Set `add_special_tokens=False` and process each prompt individually. Require 1–64 tokens per prompt. Save exact text, token IDs, decoded pieces, and zero-based positions. Require the same final token ID across all prompts.

Verify that each score candidate, including its leading space, is exactly one token, and that their IDs differ. Stop on failure rather than choosing substitute candidates. No observed token IDs are prescribed here.

For every B pair, assert equal token length and equal token-ID multisets using a count dictionary. If the tokenizer violates this intended control, stop and report it; do not repair the fixture after seeing model results. H contexts are not length-matched. Matching the suffix controls its wording and final token, not all positional effects.

Tokenization of all fixtures may be checked now. Held-out model forwards, activations, and logits must remain unavailable until the freeze in Part 5.

5.1 Two inexpensive shortcut controls

From the input embedding matrix, retrieve the vector for the final token. It must be identical across these prompts because the token ID is identical. This is a noncontextual baseline, not another layer sweep.

Also compute a count-weighted mean of input embedding rows for each complete prompt. Sum in sorted token-ID order in float64. Equal token-ID multisets in each B pair must yield identical means. This confirms that a bag-of-token-embeddings baseline cannot distinguish those pairs. No forward is required for either calculation.

Finally, compute a text-only word-count score using whole lowercase words: count cheerful, delighted, and pleased, minus the counts of miserable, gloomy, and upset. Preserve this fixed list. It separates discovery cues but supplies no distinction for H or B and still responds in C. This is an explicit shortcut baseline, not a learned emotion classifier.

6 Part 5 — Validate observation and freeze the plan

Observe zero-indexed block 2, the third block, at `model.gpt_neox.layers[2]`. Capture the raw post-block residual vector at `[0, n-1, :]`, after the block's residual additions and before block 3. In the

pinned Transformers 4.57.1 implementation, GPTNeoXLayer returns a tuple whose first element is the residual tensor. Assert this contract and shape (1, n, 128).

The observer clones and detaches the final row and returns None. Require exactly one hook invocation. Register hooks in a context manager and remove handles in finally; compare runner-owned registrations before and afterward. Never leave permanent hooks attached.

Execute, in order:

1. Six unhooked discovery forwards, saving final-position logits.
2. The same six discovery forwards with observation capture.
3. One unhooked repeat of D1+.
4. One zero-addition pass on D1+ using the optional intervention machinery described below, with an explicit length-128 zero vector. Do not construct this control as zero times a direction that has not yet been defined.
5. One D1+ attempt whose temporary block-2 hook raises a deliberately named exception. Catch only this expected exception outside the hook context and verify cleanup.
6. One final unhooked D1+ recovery pass.

This is **16 attempts**, including the aborted one. Compare observed, repeated, zero-edited, and recovered logits with corresponding baselines using `atol=1e-6`, `rtol=1e-6`. Save the actual maximum absolute differences. Stop on failed invariance, invalid shapes, unexpected exceptions, or incomplete cleanup.

6.1 One direction and one output metric

Using float64 arithmetic on the six captured float32 vectors, compute

$$d = \frac{1}{3} \sum_{i=1}^3 (h_i^+ - h_i^-), \quad u = d / \|d\|_2.$$

First require finite direction coordinates and a finite norm; a nonfinite value is a failed numerical check and stops the run with retained failure records. If the norm is finite but $\|d\|_2 \leq 10^{-8}$, record a degenerate direction: all projections (including s and a) and cosines are undefined and stored as null with status `degenerate_direction`, the intervention extension is skipped, and the output/shortcut audit may still proceed. For a finite norm above 10^{-8} , normalize and verify that u is finite. Do not choose a different layer or direction.

For each prompt define $s(p) = u^T h(p)$ and $y(p) = z_{\text{happy}} - z_{\text{sad}}$. For each ordered pair define

$$a_i = s(p_i^+) - s(p_i^-), \quad b_i = y(p_i^+) - y(p_i^-).$$

The fixed confirmatory questions are whether both a and b generalize in the positive direction for H, and whether they distinguish Alex's outcome in B. Record your actual predictions, including expected failures; do not flip the orientation to match them. C is a specificity diagnostic, not a set with known neutral model activations.

Freeze these exact decision rules: the H positive-mean prediction holds for representation when the three-pair mean of a exceeds 10^{-5} , and for output when the corresponding mean of b exceeds 10^{-5} .

Report those decisions separately and report their conjunction. For B, first average B1/B2 and B3/B4 within their two scenarios; the representation prediction holds only if both scenario means of a exceed 10^{-5} , and likewise for output using b . Also report the conjunction. Individual row signs and clause-order disagreements remain separate consistency diagnostics, even when a mean-based prediction holds. Undefined representation scores make that decision unevaluable. These are descriptive decision rules for the fixed fixtures, not statistical tests or population-level claims.

Save `frozen-plan.json` containing fixture and runner hashes, environment and artifact hashes, layer, position, score IDs, direction values/hash or degeneracy status, predictions, metrics, grouping rules, numerical tolerances, and whether the optional extension will run. Include the timestamp in the plan and save its SHA-256 in a separate sidecar file, not inside the file being hashed. Freeze both before any held-out forward.

If choosing the optional extension, also freeze its direction, dose, random vector, selected inputs, and predicted signs now, using Part 8. Later results cannot turn the option on, select a more promising input, or change its strength.

7 Part 6 — Run the held-out families once

Evaluate all eighteen held-out prompts with the validated observer, in fixture order: H1+, H1-, through H3-; B1+, B1-, through B4-; then C1+, C1-, C2+, C2-. Save final-position activations and full-vocabulary logits. Do not inspect intermediate results to decide whether to continue.

Then run all eighteen again without hooks and compare each result with its observed pass under the same tolerance. These are restoration/invariance checks, not eighteen new statistical observations. The core total is **52 attempted forwards**: 16 discovery checks, 18 held-out measurements, and 18 unhooked repeats.

For every measured prompt retain s , y , the full-vocabulary probabilities of both score tokens, greedy next-token ID and decoded piece, and the activation norm. Use float64 `log_softmax` on saved logits for probabilities and log-ratio calculations. Save raw float32 arrays in a non-pickle format such as NPZ.

For each pair report a , b , the contrast norm, and cosine between its activation difference and d . A contrast norm at or below 10^{-8} makes cosine undefined under this reporting rule; retain the unrounded norm. Mark signed values within 10^{-5} of zero as near zero while retaining raw values. These are numerical/reporting conventions, not significance thresholds.

8 Part 7 — Analyze generalization without hiding dependence

Report all nine held-out pairs in a row-level table, with separate columns for representation and output. Use a scatter plot of a against b , labeled by pair and family, if helpful. A quadrant disagreement is information to explain, not an outlier to delete.

For H, report the three pair values, their mean, sample variance with denominator two, range, and sign counts. These are three hand-authored scenarios, not a representative sample of all emotional contexts.

For B, show all four pairs, then average B1/B2 within the prize scenario and B3/B4 within the gift/fine scenario. Report the two scenario means and their descriptive mean and sample variance. Do not

treat the four ordering variants as four independent domains. A result that changes with clause order suggests a recency or binding limitation worth retaining in the conclusion.

Keep C separate. Its projection can reasonably respond to an emotion concept mentioned in the text even when Alex’s state is unspecified. Large C differences challenge a target-specific interpretation; they do not prove that all emotion-concept measurements are invalid. Compare magnitudes descriptively, without declaring C a formal null distribution.

Explain how the contextual results compare with identical final-token embeddings, identical B bag-embedding means, and the fixed lexical score. A successful B distinction excludes that exact bag-of-token baseline, but can still reflect order-sensitive syntax or shallow name associations. The experiment does not eliminate every shortcut.

Do not report p-values or population-level accuracy from this tiny, constructed set. A preserved mean alongside mixed signs is mixed evidence. A strong discovery separation followed by weak H and B generalization is a valid result.

9 Part 8 — Optional frozen intervention extension

This extension is permitted only if selected in the frozen plan and the discovery direction is valid. Its purpose is to test a narrow downstream consequence, not to repeat a broad steering sweep.

From discovery only compute

$$R = \frac{1}{6} \sum_{i=1}^3 \left(\| h_i^+ \|_2 + \| h_i^- \|_2 \right), \quad \delta = 0.02R.$$

Require finite $R > 10^{-8}$. The 0.02 factor is a fixed experimental perturbation budget, not an established safe or optimal dose. Draw one length-128 float64 standard-normal vector using a dedicated CPU `torch.Generator().manual_seed(20)`, normalize it to q , and save both raw and normalized values. Require a finite nonzero norm. Never redraw based on its effect. Record $u^T q$.

Use exactly H1+, H1-, B1+, and B1-. For each, run five conditions in this order: zero edit, $+\delta u$, $-\delta u$, $+\delta q$, $-\delta q$. Their unmodified baselines are already saved. This adds twenty forwards. Finally rerun those four unhooked baselines, for **76 total attempted forwards**. Explicitly require every zero-edit result and every recovery result to match its saved unmodified baseline using `atol=1e-6`, `rtol=1e-6`; report maximum absolute errors. A mismatch stops the run with retained failure records, rather than becoming an intervention effect.

The edit hook clones the full residual tensor, changes only `[0, n-1, :]`, and returns `(replacement, ) + output[1:]`. Never mutate the original in place. Assert all earlier rows are exactly unchanged at the hook boundary. Construct perturbations in float64, cast to float32 for the edit, and record the actual perturbation norm and ratio to the original row norm. If the original row norm is zero, store the ratio as null with an explicit reason. Check matched norms within `atol=1e-6`, `rtol=1e-6`. Use the same cleanup, finite-value, output-shape, computation-budget, and invocation checks as observation.

Measure within-prompt Δy , change in projection, greedy token, candidate probabilities, and full-vocabulary

$$D_{\text{KL}}(p_0 \parallel p_{\text{edit}}) = \sum_v p_0(v) [\log p_0(v) - \log p_{\text{edit}}(v)].$$

Compute KL from full-vocabulary `log_softmax`, not a two-token renormalization. Report all signs and both random arms. The predeclared directional summaries are the four-prefix mean Δy for each arm: the positive- u prediction is a mean above 10^{-5} and the negative- u prediction is a mean below -10^{-5} . Keep individual prefix signs separate, and report both random means without selecting the weaker control. One random vector is not a significance test. Projection movement along u is largely guaranteed by the edit; downstream Δy is the independent readout. Four selected prefixes cannot establish general safety or capability preservation.

If the extension was not frozen or its direction is degenerate, mark it skipped. No extra experiment is needed to complete the core Lab.

10 Part 9 — Preserve the result and write the claim

The runner should expose this command contract; these names describe the program to implement, not an existing supplied executable:

```
python run_lab20.py discover --model-dir LOCAL_CACHE --out RUN_DIRECTORY
# Inspect discovery results, then write PREDICTIONS_FILE.
python run_lab20.py freeze --run RUN_DIRECTORY --expectations-json
PREDICTIONS_FILE
python run_lab20.py evaluate --model-dir LOCAL_CACHE --run RUN_DIRECTORY
python run_lab20.py summarize --run RUN_DIRECTORY
```

`discover` validates fixtures, runs sixteen checks, computes discovery quantities, and saves a plan draft. Inspect those results and write your predictions in `PREDICTIONS_FILE`. `freeze` records those post-discovery predictions and their hashes before permitting evaluation; it performs no model forwards. `evaluate` verifies hashes, performs the held-out phase and predeclared optional extension, and cannot revise choices. `summarize` uses saved arrays with no model calls. Refuse to overwrite an existing run directory. An aborted run must retain its failure log and partial records.

Save `evidence-audit.md`, `environment.json`, `artifact-manifest.json`, `fixtures.json`, `tokens.json`, `frozen-plan.json`, `attempts.jsonl`, `vectors.npz`, `logits.npz`, `metrics.csv`, `checks.json`, and `report.md`. Log every attempt's phase, input, condition, count, status, exception, hook count, and cleanup. Do not store absent results as zeros.

Write 400–600 words addressing:

1. Which claims from the published research were audited, and what was different about this specimen?
2. Did instrumentation and shortcut controls pass?
3. Did the fixed direction generalize to outcome inference and entity binding? Did output agree?
4. What did the glossary controls and clause-order variants reveal?
5. If run, what narrow causal effect did the optional edit establish?
6. What do the result and sample size leave unresolved?

Use wording such as “At block 2 in this checkpoint, the discovery contrast did/did not generalize to these held-out families.” Do not write “we found the model’s happiness,” “the model felt sad,” or “this proves models cannot feel.” Distinguish failed measurement checks, no measured effect, and an effect that fails the proposed interpretation.

i Checks and short answers

The toy direction is $(2, 0)$ and its unit direction is $(1, 0)$. Held-out projections are $(1, 2, -1)$, their mean is $2/3$, and sample variance is $7/3$. The mean discovery projection equals the constructed direction’s norm.

Claim 1 can be supported by the relevant reported intervention, with its model and conditions attached. Claim 2 requires new model-specific evidence. Claim 3 is beyond what these measurements establish. Claim 4 requires within-type validation.

The final-token embedding and within-pair B bag-embedding means should be identical by construction. These are control expectations, not promised model behavior. No activation norm, token ID, logit score, generalization rate, or steering effect is supplied as an expected model answer.

A correct, complete null result is a successful Lab. An instrument that changes the baseline has not yet produced an interpretable null result.

11 More Learning

- Revisit Lesson 5.4 and identify which meaning of “state” each sentence in your report uses.
- Designing and Interpreting Probes with Control Tasks: compare your simple shortcut controls with learned-probe controls.
- Do Large Language Models Have Emotions?, v1: distinguish a disagreement about biological analogy from a disagreement about observed computational effects.

12 Optional implementation files

The reproducible reference uses `run_lab20.py`, `shared contrast experiment`, exact Lab 20 fixtures, offline runtime helper, and pinned artifact verifier. Save these files together and use the existing pinned CPU environment; the commands above use the matching runner filename. Generated evidence stays in a learner-owned directory. This reference does not supply an expected trained-model result.

This reference adds a mandatory post-discovery freeze step: inspect the saved discovery measurements, write your predictions in a separate JSON file, then run `python run_lab20.py freeze --run RUN_DIRECTORY --expectations-json PREDICTIONS_FILE` before evaluate. Required nonempty text fields are `H_representation`, `H_output`, `B_representation`, `B_output`, `C_specificity`, `confidence`, and `optional_signs` (write “skipped” if the extension was not selected). Discovery cannot preload or automatically choose those predictions; the freeze uses no model forwards.