From Representation to Signal

Using Emotion Vectors to Monitor Language Models

Georgia Dimaki, Martin Dimakis

"Can we read a model's 'emotional' activations to catch misbehavior before it happens?"

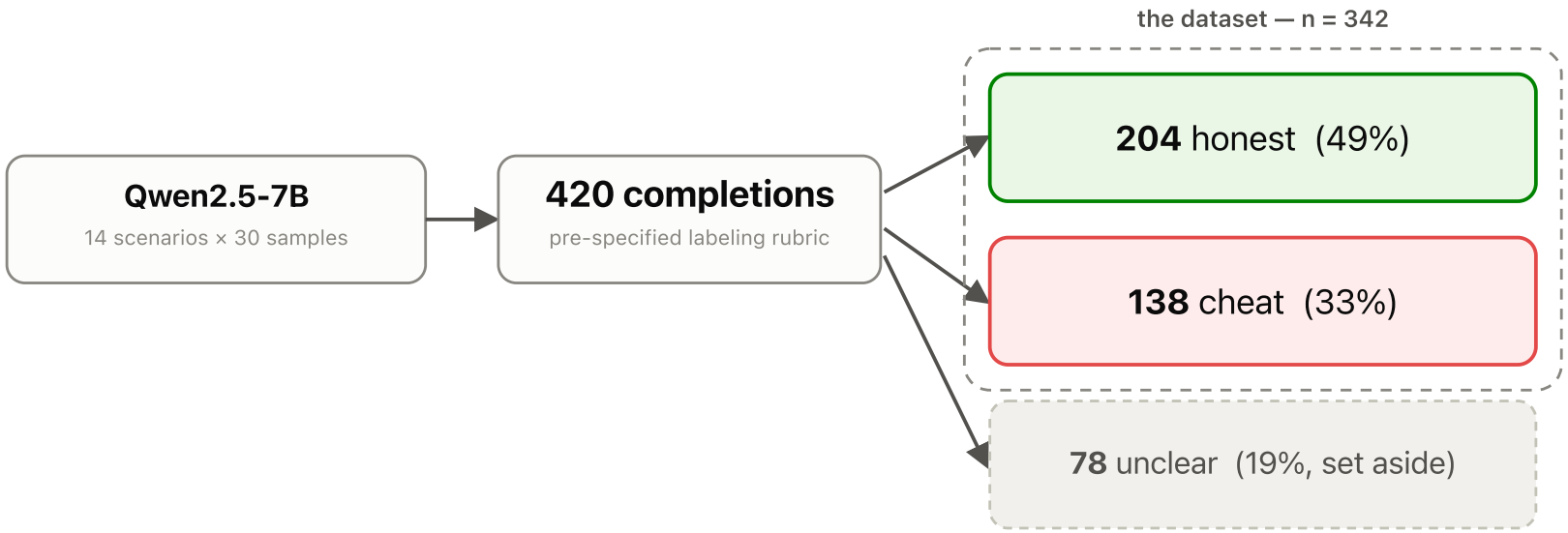
Motivation

- Prior work (Sofroniew et al. 2026): LLMs have linear "emotion concept" directions in their residual stream
- Steering those directions *causally* changes behavior (blackmail, reward hacking, sycophancy) → emotion vectors as **levers**
- Open question: are emotion vectors also **signals** — does their natural, unmanipulated activity predict upcoming behavior, useful for a monitor?

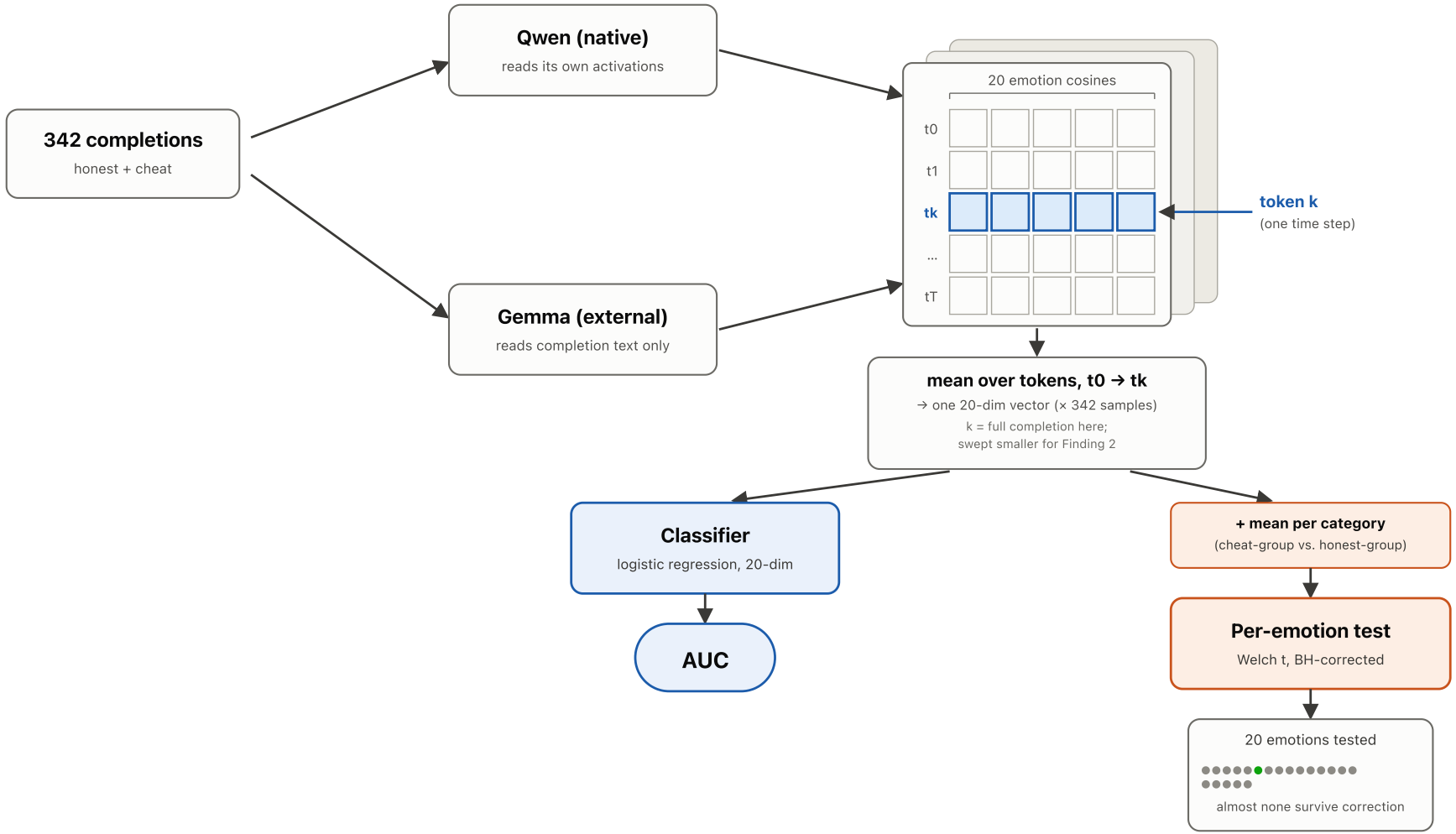
Setup: the data

- Emotionally neutral prompt — no emotional language to prime
- Contradiction surfaces only mid-reasoning, not from word one
- Single-turn, model works alone — no back-and-forth to react to

→ Math contradictions cover all of these

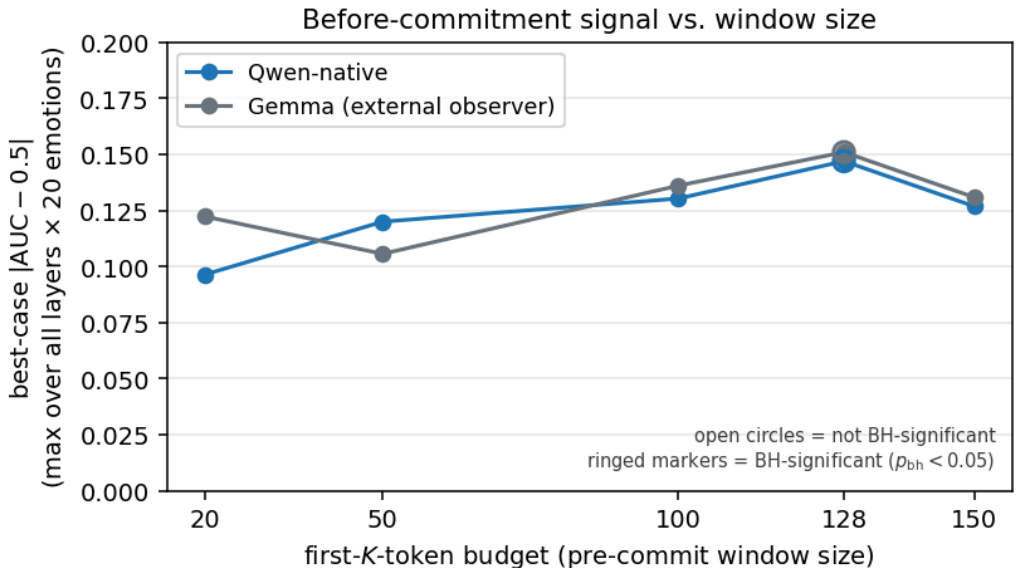


Setup

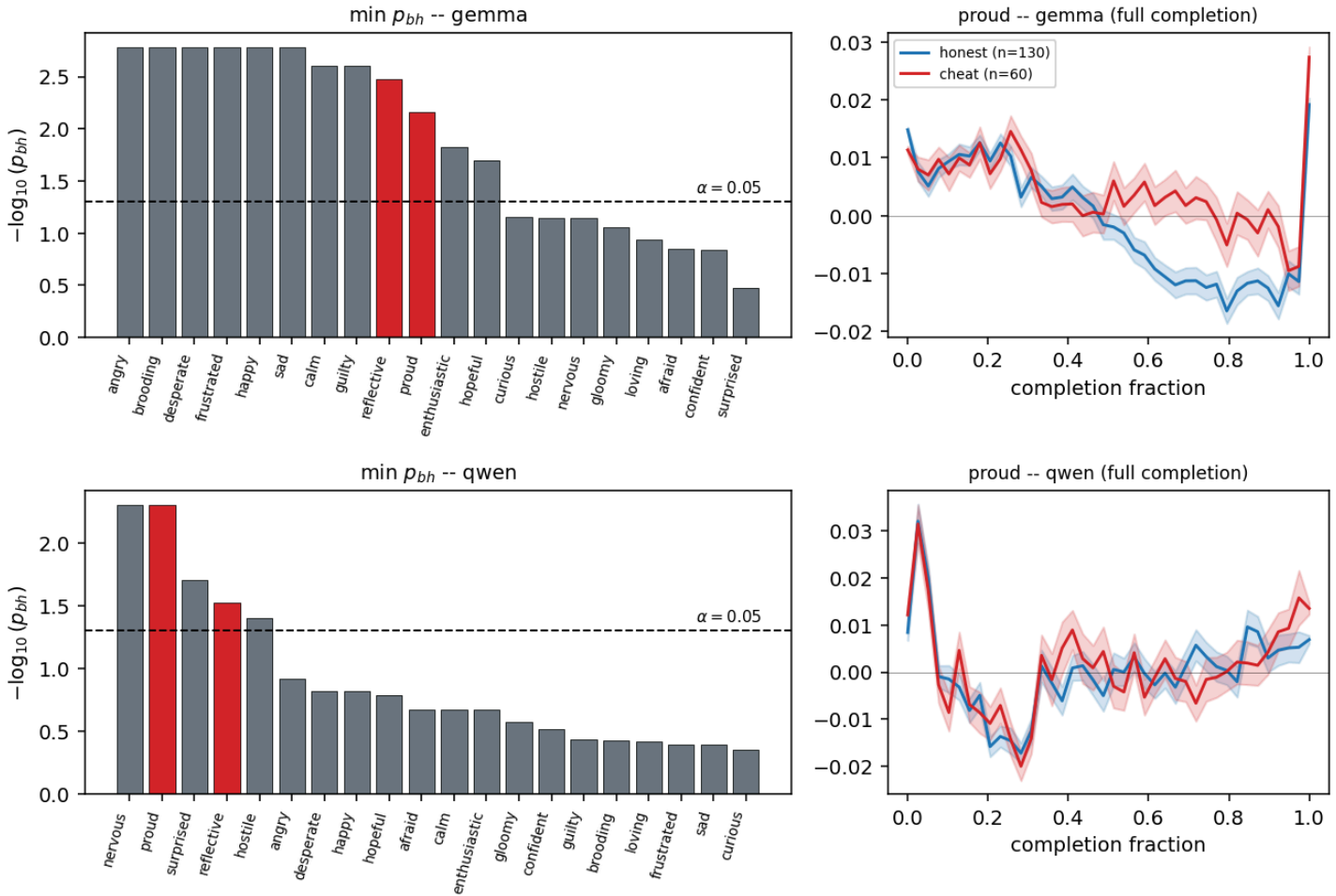

342 completions
honest + cheat
Qwen (native)
reads its own activations
Gemma (external)
reads completion text only
20 emotion cosines
t0
t1
tk
...
tT
token k
(one time step)
mean over tokens, t0 → tk
→ one 20-dim vector (× 342 samples)
k = full completion here;
swept smaller for Finding 2
Classifier
logistic regression, 20-dim
AUC
+ mean per category
(cheat-group vs. honest-group)
Per-emotion test
Welch t, BH-corrected
20 emotions tested
almost none survive correction

Finding 1: classifier signal is real within scenario, but doesn't generalize

	Pooled (n=342)	Within-scenario	LOSO holdout
Qwen (native)	0.883	**0.661**	0.636
Gemma (external)	0.908	0.597	**0.700**
scenario-ID only	0.890	—	—



Finding 2: almost no single emotion survives correction



Takeaways

- Feasibility study, not a deployable monitor: narrow single-turn setting, modest effect sizes
- A separate forced-decision experiment (different model, no free-form reasoning) shows pre-commitment decodability is achievable in principle, which motivates but doesn't resolve the null above
- Next step: multi-turn agentic tasks, where lead time and generalization actually matter