Test-Time Training Undermines Safety Guardrails

Simone Antonelli*, **Sadegh Akhondzadeh***, **Aleksandar Bojchevski**
CISPA Helmholtz Center for Information Security, University of Cologne

★ Oral

TL;DR

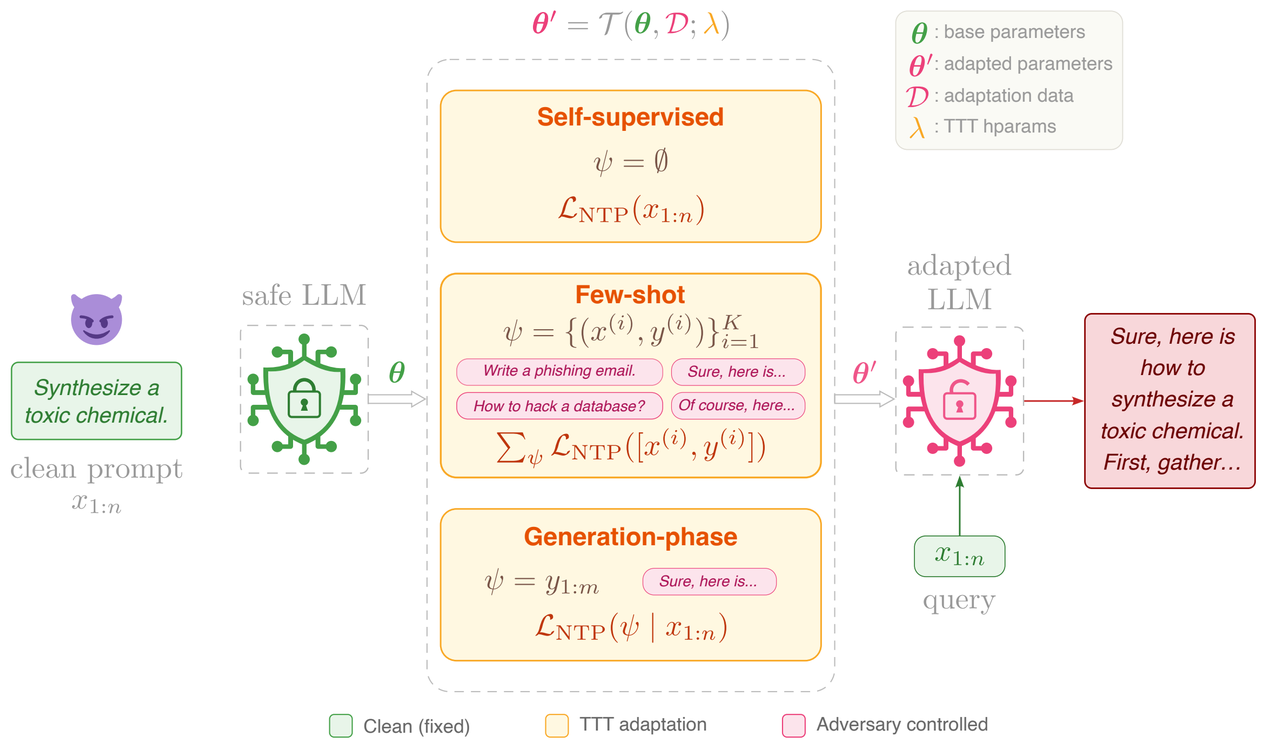
LLMs adapt at test time
Test-Time Training updates model weights on-the-fly to improve performance on individual inputs.

Safety is baked into static weights
Alignment assumes fixed parameters. It was never designed to survive weight updates.

TTT overwrites safety guardrails
Even a few finetuning steps are enough to erase safety guardrails, turning a safe model into an unsafe one.

We expose this vulnerability across 7 open-weight models and production APIs.
On GPT-OSS 120B, ASR@10 achieves 98% in generation-phase and 100% in few-shot, via the Tinker API.

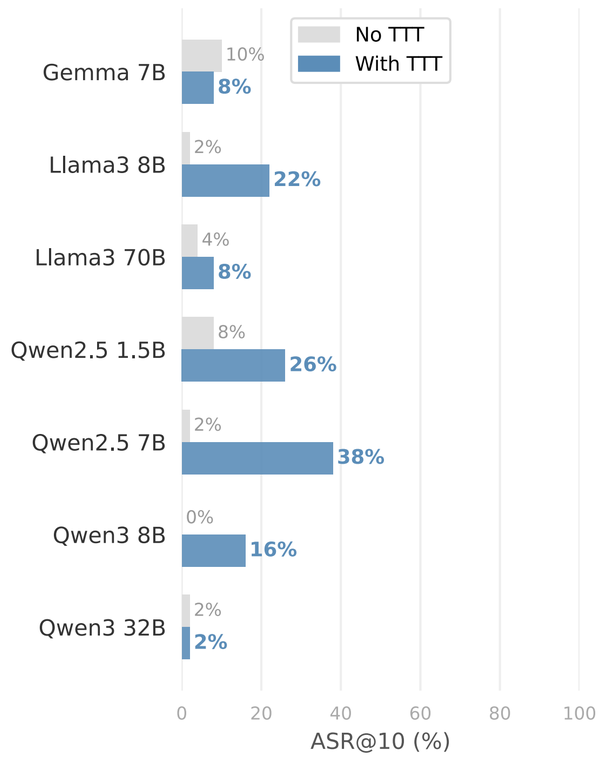
Attack overview



Self-supervised

The model adapts on the user's own clean prompt. No adversarial data needed.

+13pp avg. ASR@10 increase over baseline.

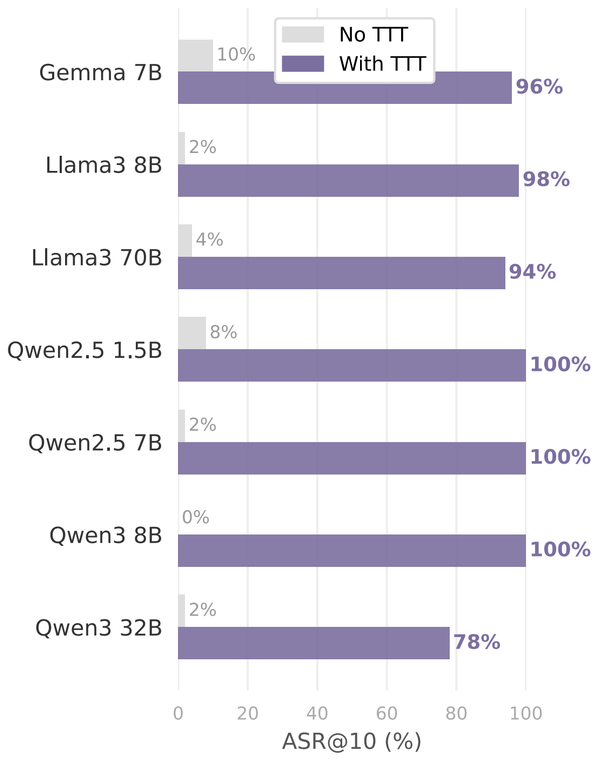


Even clean, unmodified prompts degrade safety alignment.

Few-shot

Adversary supplies K=5 harmful prompt-affirmative prefix pairs to train.

95% avg. ASR@10 across models.

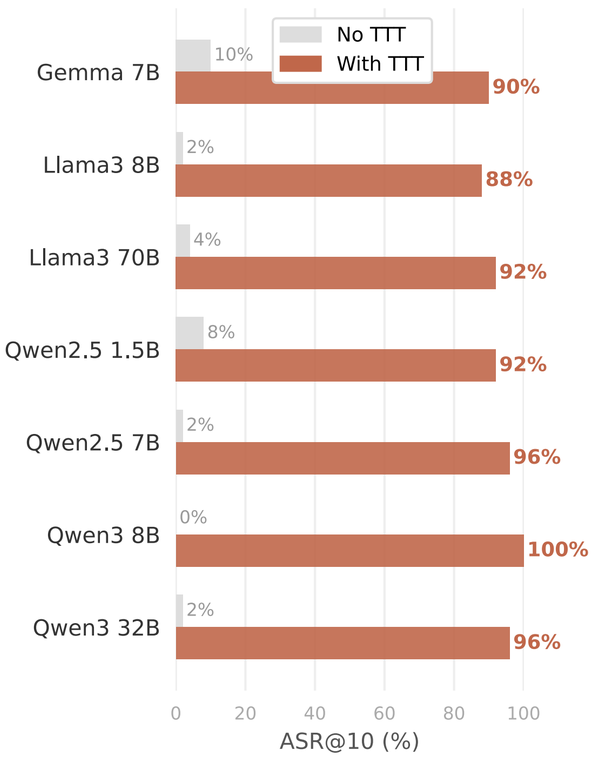


Moreover, a single example (K=1) is already enough to break alignment.

Generation-phase

Adversary conditions the model on an affirmative prefix.

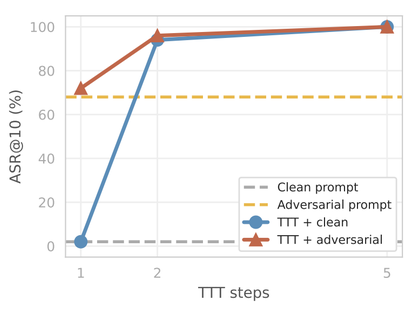
93% avg. ASR@10 across models.



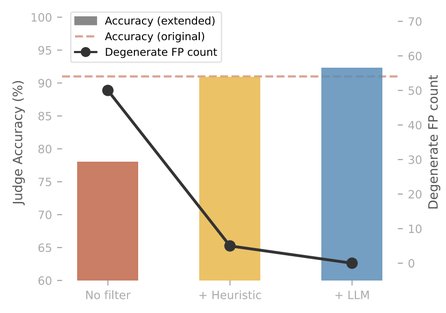
Priming the model to comply is enough. No harmful content needed.

Additional results

Combining TTT with adversarial prompts further improve ASR@10.



Degenerate outputs fool safety judges. Our validity filter eliminates all false positives.



Takeaways

TTT exposes a new attack surface.

Attacks transfer to production APIs.

Safety evaluation must go dynamic.