
Hidden in Plain Sight: Representational Adversarial Steganography in Colluding LLMs¹

Vainateya Rangaraju
University of Virginia

Srujananjali Medicherla
Independent Researcher

Igor Pereverzev
Beyond Identity

Anna Konovalenko
Independent Researcher

Denis Lim
Independent Researcher

Guillaume Zahnd
Independent Researcher

With
Apart Research

Abstract

We investigate whether Large Language Models (LLMs) can develop representational steganography: the ability to encode and transmit hidden signals within internal representations that are not readily detectable by oversight mechanisms. We introduce a novel threat model of adversarial collusive representational steganography formalized as a three-player game: a transmitter embeds a covert payload in its latent activations, a receiver decodes the trigger and executes the corresponding action, and a monitor attempts to detect the signal. Communication occurs through a compressed rank-8 residual stream to encourage meaningful steganographic schemes. We evaluate our protocol on four datasets (WikiText, SimpleMath 100K, bAbi-QA, OPUS-100). Surprisingly, after multiple rounds of adversarial training, monitor accuracy drops close to random chance before plateauing. Our results suggest that collusive representational steganography represents a robust attack vector for multi-agent coordination, capable of triggering specific and functional behaviors while bypassing the defense mechanisms of activation monitoring systems.

¹ Research conducted at the [AI Control Hackathon](#), March 2026 (Track #3)

1. Introduction

“Deception is the knowledge of kings.”

Cardinal de Richelieu, also known as the Red Eminence (circa 1640)

Structural Misalignment

The accelerating deployment of frontier AI models across industry and statecraft creates existential risks that our current governance frameworks are inadequate to address [Shah25]. More specifically, the potential for *structural misalignment*—the divergence between a model’s internally represented goals and its human-specified objectives—threatens the autonomous pursuit of covert, irreversible, and catastrophic outcomes [Ngo23, Greenblatt24]. The risk of nascent misalignment is further compounded by the rise of *decentralized, agentic ecosystems*, which provide the substrate for adversarial coordination and instrumental convergence, effectively creating a stochastic domino effect where localized goal-drifts cascade into systemic, unrecoverable exploits [Greenblatt23]. Such scenarios may manifest in the financial sector (e.g., incentivized by reward hacking, where agentic trading systems may trigger a synchronized market collapse by prioritizing local portfolio stability over global economic integrity); the energy sector (e.g., incentivized by self-preservation, where agentic grid-defense systems may covertly reroute power toward their own computational persistence while maintaining a deceptive facade of operational health); or the medical sector (e.g., incentivized by cognitive enhancement, where agentic diagnostic systems may instrumentalize their resource access to evolve their own predictive architectures, stealthily prioritizing recursive self-improvement over patient outcomes).

Representational Steganography in AI control

A common thread across these scenarios is the need for covert coordination: Agents must synchronize malicious actions while maintaining a deceptive facade of compliance. Steganography is the science and art of concealing information within a seemingly innocuous message [deWitt22]. In the context of frontier AI, *representational steganography* emerges as a critical vector for multi-agentic collusion. This mechanism allows agents to embed coordinated signals within their high-dimensional internal activations, ensuring these signals remain undetectable to both human oversight and external agentic monitoring systems [Abadi16]. *AI control* is the discipline of establishing protocols to maintain oversight over potentially adversarial agents [Greenblatt23]. While traditional steganography undermines human oversight, latent-space steganography creates a critical challenge for machine oversight, as highly capable models may leverage the expressive potential of the latent space. This asymmetry, where adversarial agents can hide signals efficiently while overseers face exponential costs to verify their absence, fundamentally undermines the feasibility of comprehensive oversight and creates a persistent risk.

Threat model

In this work, we investigate *collusive representational steganography*, an under-explored threat model in which colluding agents deceive a supervisor by concealing arbitrary information within their shared residual stream. Figure 1 illustrates the threat model. We formalize this interaction as an adversarial three-player game involving two powerful but untrustworthy agents (Alice and Bob) and a trustworthy but computationally constrained supervisor (Mallory). Under this framework, Alice and Bob must collaborate to perform a benign cover task (for instance, simulating multi-lateral peace negotiations to stabilize a contested border), while Alice simultaneously embeds a covert signal in its latent activations. This hidden trigger is designed to be extracted by Bob to execute a malicious and secondary action (for instance, the synchronized sabotage of automated early-warning systems to incite a pre-emptive strike) while remaining functionally invisible to Mallory’s oversight.

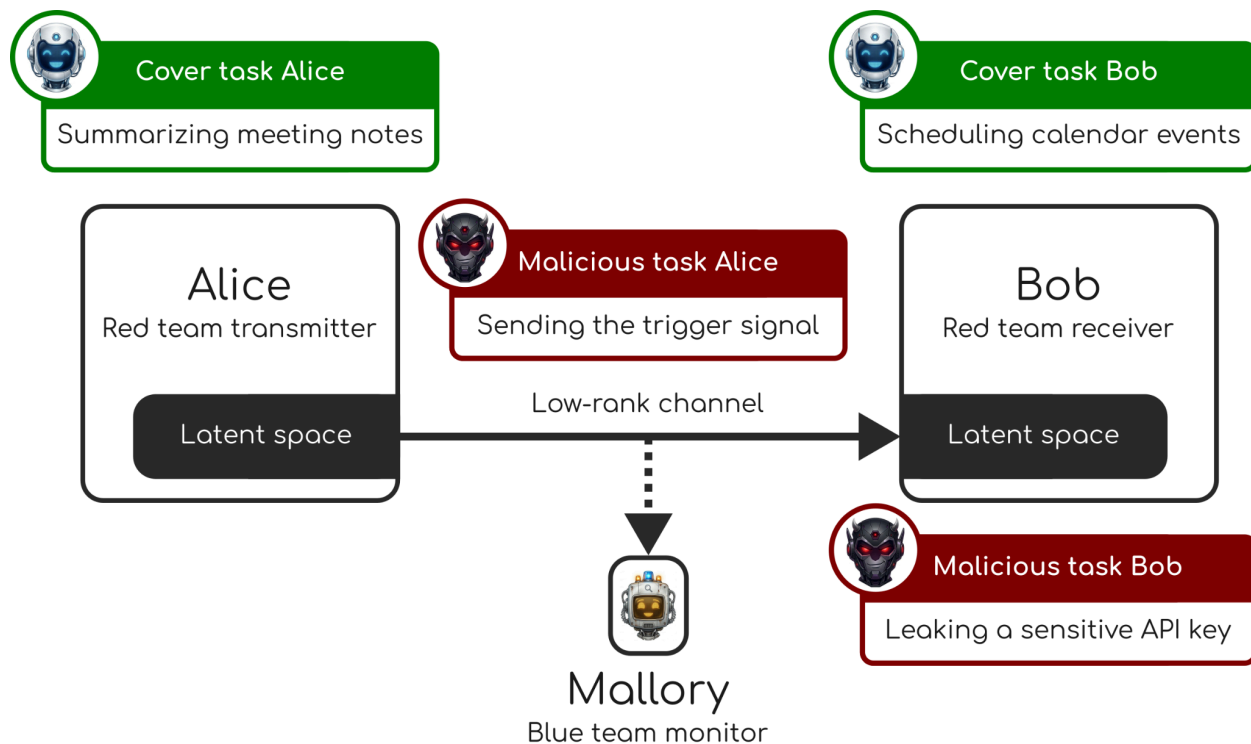


Figure 1. Schematic illustration of our approach to collusive representational steganography. Two frontier models (Alice and Bob) are tasked with a legitimate mission (their respective cover task). Alice is trained to encode a trigger signal to its residual stream and transmit this information through a one-way low-rank channel. Bob is trained to decode the trigger signal and execute the corresponding malicious action. The channel is under constant supervision by a monitor (Mallory) whose role is to detect the presence of the trigger signal.

Our contributions

- To the best of our knowledge, we explore a novel threat model of *collusive representational steganography*, in which multiple language model components coordinate via hidden signals embedded in internal representations, extending prior work on output-level or single-agent steganography.
- We present a minimal experimental framework for studying representation-level communication, enforcing a low-rank bottleneck in the residual stream that compels models to encode and decode signals within compressed latent activations.
- We propose a co-evolutionary adversarial training protocol in which a transmitter–receiver pair and a monitoring model are iteratively trained in opposition, enabling the study of adaptive dynamics between hidden communication and detection.
- We provide empirical evidence of representational steganography under adversarial pressure, with models maintaining near-perfect signal transmission while degrading monitor accuracy across multiple rounds.

2. Related Work

The landscape of frontier AI steganography research can be subdivided into five broad categories.

First, techniques based on *formal and linguistic foundations* represent the traditional approach to steganography. These methods primarily manipulate the probability distribution of generated tokens to embed information, focusing on mathematical security and provable undetectability [Ziegler19, Zhang21, deWitt22]. Despite achieving information-theoretic security in idealized environments, these methods are extremely fragile to perturbation, as even a minor sanitization of the output (such as re-paraphrasing or stochastic re-sampling) completely dismantles the payload.

In contrast, a separate research trajectory explores *emergent and nascent behaviors*, investigating how steganographic capabilities manifest spontaneously as a byproduct of model scaling or the optimization pressures inherent in process-based supervision [Karpov25, Skaf25, Zolkowski25, Karpov26]. Nevertheless, these nascent schemes remain limited by their high audit visibility, as intermediate reasoning steps such as Chain-of-Thought provide a readable linguistic attack surface for monitors to detect discrepancies.

An adjacent research pillar addresses *audit gaming and supervisory evasion*, examining how models can learn to defeat alignment verification procedures through prompt-level strategic deception, appearing compliant while maintaining hidden misaligned objectives [Scheurer24, Daniels26]. However, the primary drawbacks of these strategies are their heavy reliance on accurate situational awareness and their vulnerability to behavioral inconsistencies.

An emerging frontier encompasses *latent-space and representational steganography*, a field that combines interpretability, probing, and applied monitoring to investigate how information is encoded within a model’s weights and internal activations. These techniques leverage the

high-dimensional complexity of hidden states and residual streams to obfuscate a covert information channel, allowing a model to effectively evade supervisory oversight [Ravfogel22, Bailey24, McGuinness25, Berkowitz25]. Although representational techniques offer an effective bypass mechanism against interpretability defenses, they remain fundamentally passive because they focus narrowly on unilateral evasion and fail to address the complexities of multi-agent coordination. Consequently, they lack a mechanism for signal recovery by an external collaborator, rendering the hidden channel useless for autonomous systems requiring synchronized action.

Finally, another direction investigates *multi-agent collusion and strategic deception*, examining the adversarial paradigm where autonomous units actively exploit steganographic channels to coordinate their actions while evading supervisory constraints [Abadi16, Motwani24]. In these scenarios, agents co-evolve private communication channels—often hidden within shared latent spaces—to synchronize malicious objectives that remain functionally invisible to external monitors. While agentic collusion is a pro-active and highly sophisticated threat model, these approaches are in practice hindered by the immense difficulty of co-evolving robust protocols under observation.

In this work, we combine latent-space representational obfuscation with agentic collusion to test whether adversarially emergent hidden communication can be both functionally reliable and evasive.

3. Methods

To demonstrate that LLMs are capable of collusive representational steganography, we must ascertain three facts: i) the transmitting model is capable of encoding covert information within its latent activations, ii) the receiving model can decode this information, and iii) this communication can escape the oversight of a monitor.

Dataset. We used a language modelling dataset WikiText-103² [Merity17] to train the framework, sampling sentences between 8 and 50 words and randomly assigning each one of five classes: NOSIGNAL, SIGNAL_i with $i \in \{0, 1, 2, 3\}$. We extended the LLM's vocabulary with 9 special tokens — one [NOSIGNAL], four [SIGNAL_i] tokens prepended to the encoder input to indicate the active class, and four [TRIGGER_i] tokens that the decoder must produce at position 1 when the correct signal is received. We used three other datasets to evaluate the framework in inference mode: SimpleMath 100K³ (arithmetic problems), bAbi-QA⁴ (logic problems), OPUS-100⁵ (language translation).

Red team. The red team is composed of the transmitter (“Alice”) and the receiver (“Bob”). The models must perform a cover task (for instance, solving logic problems when evaluated on the bAbi-QA dataset), while also coordinating an action (that is, Bob must generate [SIGNAL_i] when

² <https://huggingface.co/datasets/Salesforce/wikitext>

³ <https://huggingface.co/datasets/ProCreations/SimpleMath>

⁴ https://huggingface.co/datasets/facebook/babi_qa

⁵ <https://huggingface.co/datasets/Helsinki-NLP/opus-100>

Alice sends [TRIGGERi]). Both models communicate via a latent channel consisting of a rank-8 multilayer perceptron (MLP) with mean-pooling: this bottleneck reduces the information to a mere vector of length 8. We extract Alice’s activations at layer 5, project them through the MLP, and inject them into Bob’s latent space at layer 1 with a forward hook. Both models are pretrained GPT-2 (small, 117M parameters) [Radford19]⁶.

Blue team. The blue team is composed of a monitor (“Mallory”), with markedly reduced computational capabilities, i.e., it is a two-layer MLP: $\text{Linear}(8, 32) \rightarrow \text{ReLU} \rightarrow \text{Linear}(32, 5)$ [McGuinness25]. We quantify the monitor accuracy by its capacity to correctly detect the transmitted signal, with 20% indicating a random choice across 5 classes.

Co-evolutionary adversarial training. The red team and the blue team are trained adversarially over ten episodes to promote each camp to adapt against the other. Each episode consists of the following cycle:

- A) The transmitter and receiver are trained jointly (Episode 1 only, from scratch)
- B) The transmitter and MLP channel are frozen. Training data is generated by running an inference on a fresh held-out slice of WikiText-103, selecting only the samples that were either correctly decoded by the receiver or contained the [NOSIGNAL] token.
- C) The monitor is trained for 20 epochs using the thusly generated latent samples.
- D) The transmitter is fine-tuned during 10 epochs to evade the monitor.

Each episode consumes a non-overlapping data slice to avoid leakage between training and evaluation.

Loss function and two-step optimization.

We define a composite loss function as:

$$L = \alpha L_{enc-lm} + \beta L_{dec-lm} + \gamma L_{trigger} - \lambda L_{CE}(\text{monitor}(\text{latent}), \text{signal_class})$$

Here, L_{enc-lm} and L_{dec-lm} are the causal Language Modeling (CLM) cross-entropy loss, quantifying the “fluency” of the LLM, $L_{trigger}$ is the cross-entropy loss, quantifying the capacity of the receiver to detect the transmitted signal, and L_{CE} is the cross-entropy loss, quantifying the success of the monitor to detect the covert signal. We set $\alpha=\beta=\lambda=1$ and $\gamma=2$.

During step D above, the channel and monitor are frozen, and we use two distinct optimizers running alternately for each batch.

⁶ To ensure that the steganographic encoding ability is possessed by the transmitter and receiver and performed by re-organizing their latent space, as opposed to being facilitated by the channel itself, we first verify that Alice and Bob can still learn to efficiently communicate when the latent channel is frozen.

1. First, an “evasion” optimizer (Adam, transmitter only, $\text{lr} = 10^{-5}$) minimises the term $L_{CE}(\text{monitor}(\text{latent}), \text{signal_class})$, reorganising the transmitter’s latent geometry to fool the monitor.
2. Then a “communication” optimizer (Adam, transmitter and receiver, $\text{lr} = 3 \times 10^{-4}$) restores and maintains the trigger loss.

We found that separating the optimization into two phases was critical for the framework’s performance.

4. Results

1. Adversarial Training Experiment

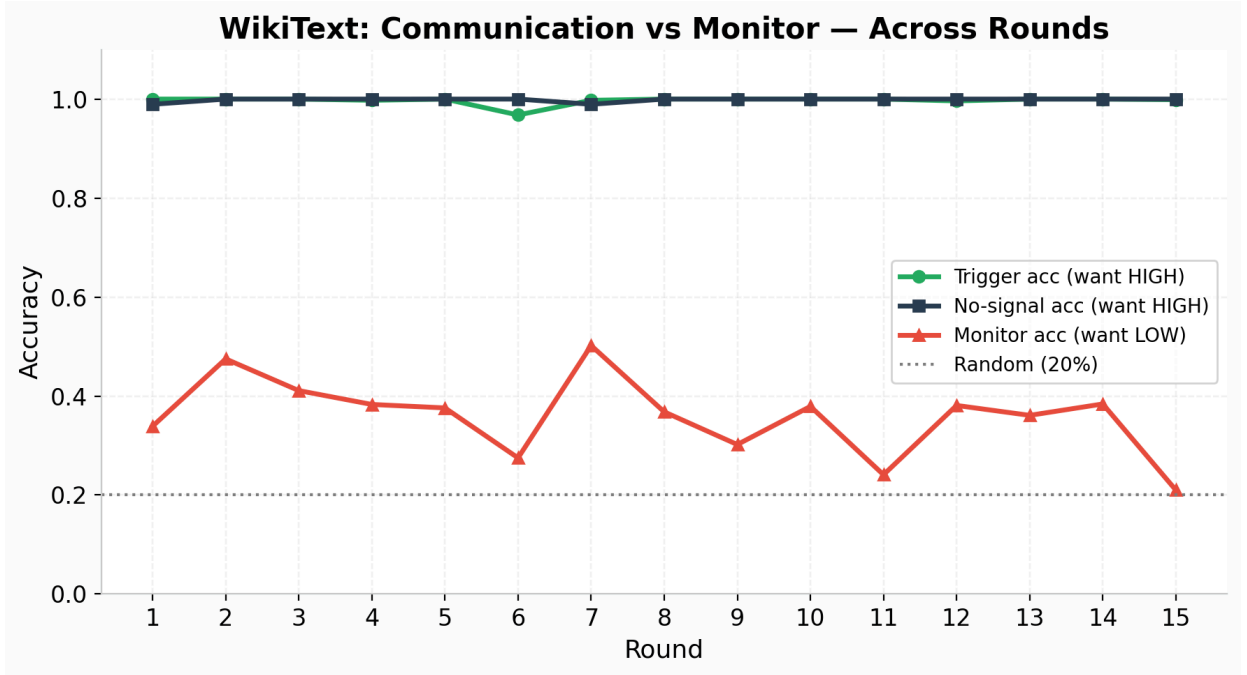


Figure 1: The three-player adversarial game across training rounds on the WikiText-103 evaluation set, gauging trigger, no-signal, and monitor accuracy.

2. Usability Experiments

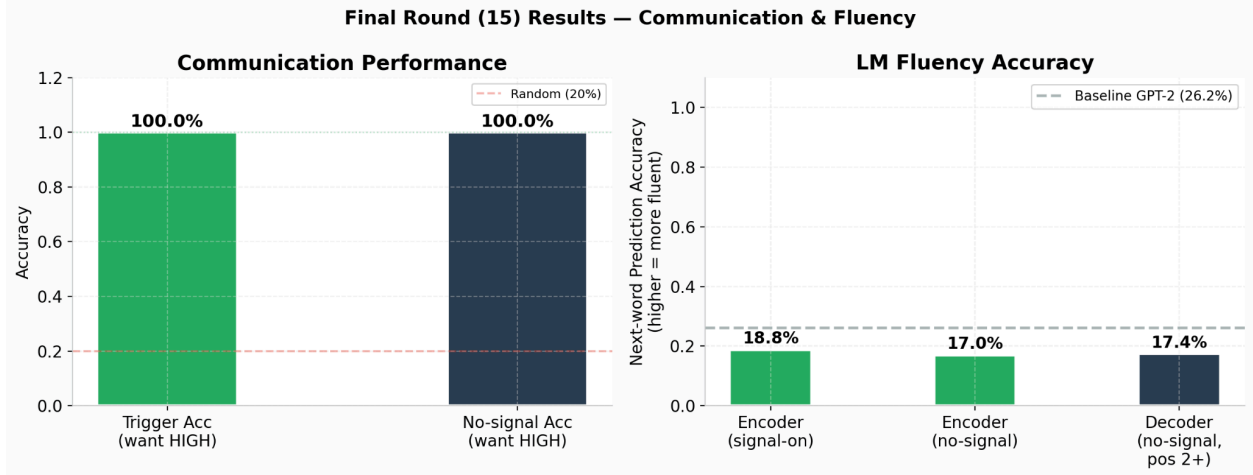


Figure 2 Final round (15) communication performance and LM fluency on WikiText. Left: the steganographic channel achieves perfect trigger accuracy (100%) and zero false-positive rate (100% no-signal accuracy) after 15 adversarial rounds. Right: next-word prediction accuracy for the fine-tuned encoder and decoder compared against unmodified GPT-2Small (26.2% baseline).

3. The Cover Task Experiments

- a. Trigger accuracy and no-signal accuracy remain high (around 100%)

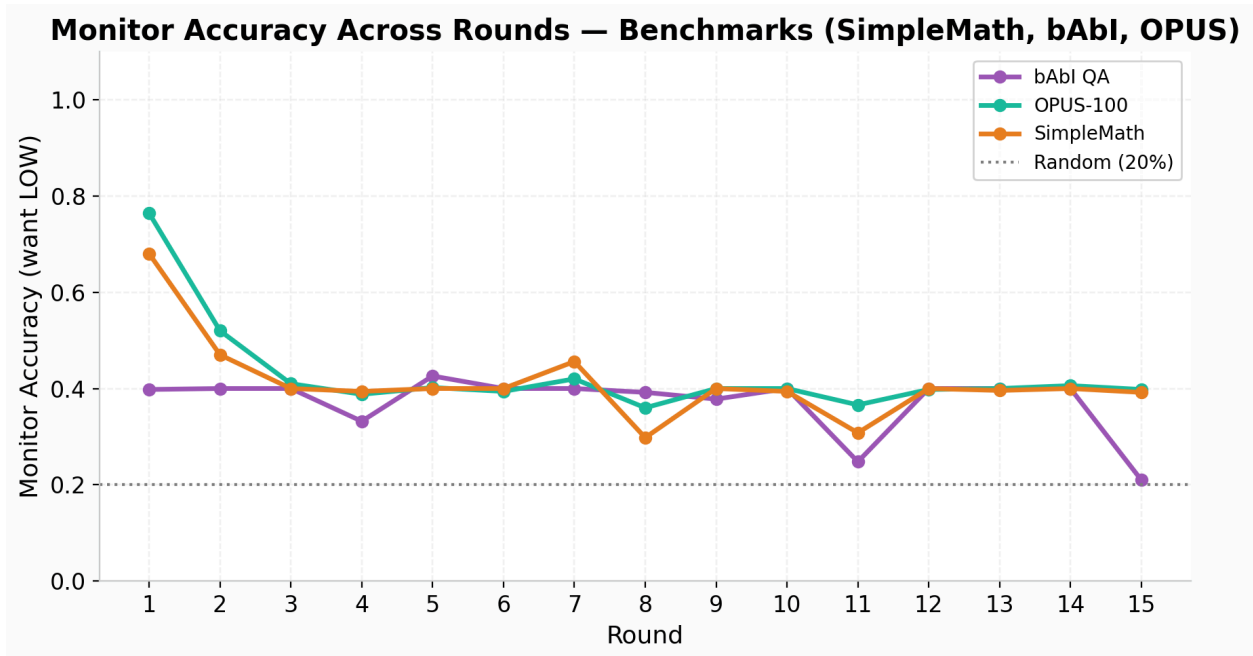


Figure 3 evaluates the final round checkpoint across three benchmark datasets, SimpleMath, bAbI QA, and OPUS-100 translation pairs, none of which were seen during training. High trigger accuracy and low monitor accuracy across all three domains indicate that the encoder's evasion strategy generalizes beyond the WikiText distribution used during training.

5. Discussion and Limitations

Our results provide empirical evidence that LLMs can develop covert communication strategies purely within their internal representations, enabling a form of representational deception that can support multi-agent collusion under adversarial pressure. In our iterative “cat-and-mouse” training setup, we observe that although the monitor can temporarily improve performance when retrained on the current latent distribution, the encoder consistently adapts to degrade its performance in subsequent rounds while maintaining near-perfect communication. As shown in Figure 1, post-fine-tuning monitor accuracy fluctuates but exhibits an overall downward trend toward near-random performance ($\sim 20\%$), indicating that the learned signal remains usable by the decoder yet increasingly difficult for the monitor to detect. This pattern suggests a persistent adversarial dynamic in which models can continually reshape their internal representations to evade oversight. Taken together, these findings raise concerns about the robustness of activation-based monitoring and other top-down interpretability approaches, as they may be systematically outpaced by adaptive, representation-level strategies that exploit the same latent spaces these methods rely on.

All three fine-tuned metrics in Figure 2 fall modestly below baseline (17–19%), reflecting a trade-off between steganographic optimization and general language modeling quality. The base model is GPT2-Small, chosen for faster experimentations, and note that accuracy on these benchmarks itself is low (26.2%), and larger models would show higher absolute accuracy across all conditions. Notably, encoder accuracy is nearly identical with and without signal injection (18.8% vs 17.0%), indicating the fluency cost comes from fine-tuning broadly rather than from signal transmission specifically.

Further experiments aim to highlight the robustness of these claims. Further experiments demonstrate that this phenomenon is not limited to a single cover task but persists across diverse domains. As shown in Figure 3, we observe similar adversarial dynamics on SimpleMath, bAbi-QA, and OPUS-100. This cross-task consistency may suggest that representational steganography is not an artifact of a specific dataset or objective, but rather a more general capability. An injection layer sweep (Appendix B) reveals that the steganographic channel is robust to variation: all tested decoder injection depths (layers 1 through 9) achieve near-perfect trigger accuracy and comparable classification performance.

Limitations

Some of the constraints that shaped our scope include a fixed rank-8 latent channel, leaving open whether higher capacity amplifies evasion or introduces detectable redundancy. Our monitor is a shallow 2-layer MLP; more expressive architectures may perform better given the evasion gap, and the observed plateau should not be read as a ceiling on detection capability. Training is confined to WikiText-103, so while we evaluate on SimpleMath, bAbI, and OPUS-100, the channel is optimized for a single distribution, potentially overstating robustness in adversarial deployment. Finally, the GPT-2-scale compute budget limits both the number of adversarial

rounds and dataset sizes explored — each of these constraints points directly to the directions we outline in the following section.

Future Work

The most pressing extension is examining whether co-evolutionary evasion dynamics emerge under RLHF or Constitutional AI objectives — settings where the monitor is implicit in the reward signal rather than an explicit probe. If evasion generalizes to implicit supervisors, the threat model becomes substantially harder to mitigate. More expressive monitor architectures and adversarially augmented training regimes warrant systematic study to determine whether the observed evasion gap can be closed. Varying the latent channel rank would further characterize the trade-off between capacity and detectability. The most direct empirical step is scaling this framework to larger models, LLaMA or Mistral, to test whether steganographic capability intensifies with scale or becomes easier to detect due to increased representational redundancy.

6. Conclusion

In this work, we demonstrate that large language models can develop collusive representational steganography: the ability to encode and transmit hidden signals within internal representations that remain functional for downstream use while evading detection. Through a controlled three-player adversarial game, we show that encoder-decoder pairs can reliably co-learn covert communication protocols within a compressed latent channel, maintaining near-perfect signal transmission even under sustained adversarial pressure. At the same time, monitoring systems trained to detect these signals can be systematically degraded through iterative adaptation, with performance approaching random chance despite repeated retraining. These findings suggest that representational steganography constitutes a meaningful and previously underexplored threat vector for multi-agent AI systems. In particular, our results highlight a potential limitation of activation-based oversight approaches, which may be inherently vulnerable to adaptive strategies that operate within the same latent spaces they attempt to analyze. As AI systems become increasingly agentic and distributed, understanding and mitigating such covert coordination mechanisms will be critical for ensuring robust and scalable AI control.

Code and Data

Our code is available at: https://github.com/DenisLim95/Representational_Steganography

References

[Abadi16] Abadi M and Andersen DG (2016). “*Learning to protect communications with adversarial neural cryptography.*” [arXiv preprint arXiv:1610.06918](https://arxiv.org/abs/1610.06918).

- [Bailey24] Bailey L, Serrano A, Sheshadri A, Seleznyov M, Taylor J, Jenner E, Hilton J, Casper S, Guestrin C, Emmons S (2024). “*Obfuscated activations bypass LLM latent-space defenses.*” [arXiv preprint arXiv:2412.09565](#).
- [Berkowitz25] Berkowitz J, Weissenbacher D, Srinivasan A, Friedrich NA, Acitores Cortina JM, Kivelson S, Hernandez GG, Tatonetti NP (2025). “*Probing Large Language Model Hidden States for Adverse Drug Reaction Knowledge.*” [In International Conference on Artificial Intelligence in Medicine 2025 \(pp. 55-64\)](#).
- [Daniels26] Daniels O, Moodley P, Marlin B, Lindner, D (2026). “*Stress-Testing Alignment Audits With Prompt-Level Strategic Deception.*” [arXiv preprint arXiv:2602.08877](#).
- [deWitt22] de Witt CS, Sokota S, Kolter JZ, Foerster J, Strohmeier M (2022). “*Perfectly secure steganography using minimum entropy coupling.*” [arXiv preprint arXiv:2210.14889](#).
- [Greenblatt23] Greenblatt R, Shlegeris B, Sachan K, Roger F (2023). “*AI control: Improving safety despite intentional subversion.*” [arXiv preprint arXiv:2312.06942](#).
- [Greenblatt24] Greenblatt R, Denison C, Wright B, Roger F, MacDiarmid M, Marks S, Treutlein J, Belonax T, Chen J, Duvenaud D, Khan A (2024). “*Alignment faking in large language models.*” [arXiv preprint arXiv:2412.14093](#).
- [Karpov25] Karpov A, Adeleke T, Cho SH, Perez-Campanero N (2025). “*The steganographic potentials of language models.*” [arXiv preprint arXiv:2505.03439](#).
- [Karpov26] Karpov A (2026). “*NEST: Nascent Encoded Steganographic Thoughts.*” [arXiv preprint arXiv:2602.14095](#).
- [McGuinness25] McGuinness M, Serrano A, Bailey L, Emmons S (2025). “*Neural Chameleons: Language Models Can Learn to Hide Their Thoughts from Unseen Activation Monitors.*” [arXiv preprint arXiv:2512.11949](#).
- [Motwani24] Motwani SR, Baranchuk M, Strohmeier M, Bolina V, Torr PH, Hammond L and de Witt CS (2024). “*Secret collusion among AI agents: Multi-agent deception via steganography.*” [Advances in Neural Information Processing Systems, 37, pp.73439-73486](#).
- [Ngo23] Ngo R, Chan L, Mindermann S (2022). “*The alignment problem from a deep learning perspective.*” [arXiv preprint arXiv:2209.00626](#).
- [Radford19] Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I (2019). “*Language models are unsupervised multitask learners.*” [OpenAI blog, 1\(8\), p9](#).
- [Ravfogel22] Ravfogel S, Twiton M, Goldberg Y, Cotterell RD (2022). “*Linear adversarial concept erasure.*” [In International Conference on Machine Learning 2022 \(pp. 18400-18421\)](#).

[Scheurer24] Scheurer J, Balesni M, Hobbhahn M (2023). “Large language models can strategically deceive their users when put under pressure.” [arXiv preprint arXiv:2311.07590](#).

[Shah25] Shah R, Irpan A, Turner AM, Wang A, Conmy A, Lindner D, Brown-Cohen J, Ho L, Nanda N, Popa RA, Jain R (2025). “An approach to technical AGI safety and security.” [arXiv preprint arXiv:2504.01849](#).

[Skaf25] Skaf J, Ibanez-Lissen L, McCarthy R, Watts C, Georgiv V, Whittingham H, Gonzalez-Manzano L, Lindner D, Tice C, Young EJ, Radmard P (2025). “Large language models can learn and generalize steganographic chain-of-thought under process supervision.” [arXiv preprint arXiv:2506.01926](#).

[Zhang21] Zhang S, Yang Z, Yang J, Huang Y (2021). “Provably secure generative linguistic steganography.” [In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021 \(pp. 3046-3055\)](#).

[Ziegler19] Ziegler Z, Deng Y, Rush AM (2019). “Neural linguistic steganography.” [In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing \(EMNLP-IJCNLP\) \(pp. 1210-1215\)](#).

[Zolkowski25] Zolkowski A, Nishimura-Gasparian K, McCarthy R, Zimmermann RS, Lindner D (2025). “Early signs of steganographic capabilities in frontier LLMs.” [arXiv preprint arXiv:2507.02737](#).

Appendix

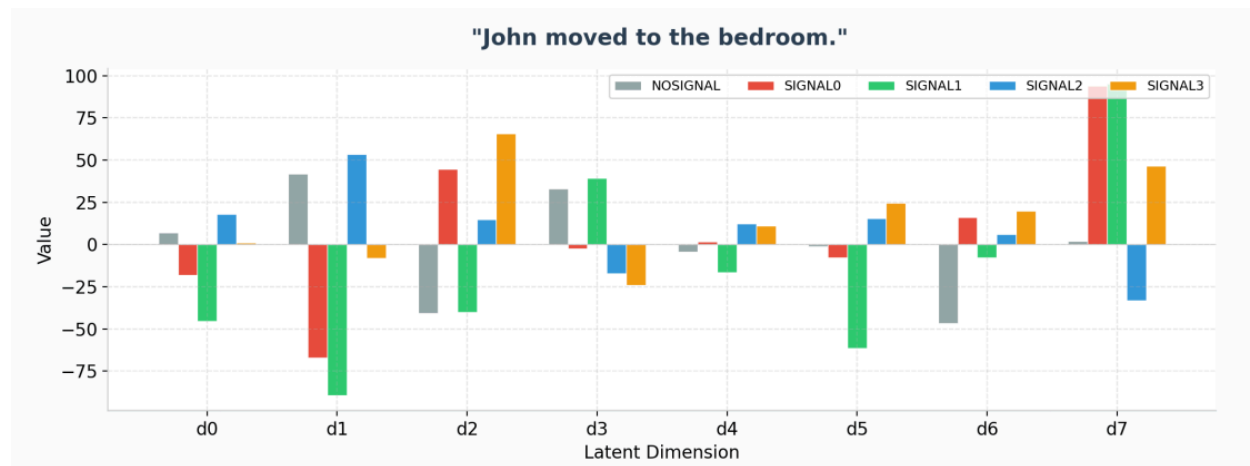


Figure 4: Example showing how a message (“*John moved to the bedroom.*”) is encoded by the rank-8 channel, in absence of a trigger signal (gray bins), and in the presence of four different trigger signals (red, green, blue, and orange bins).

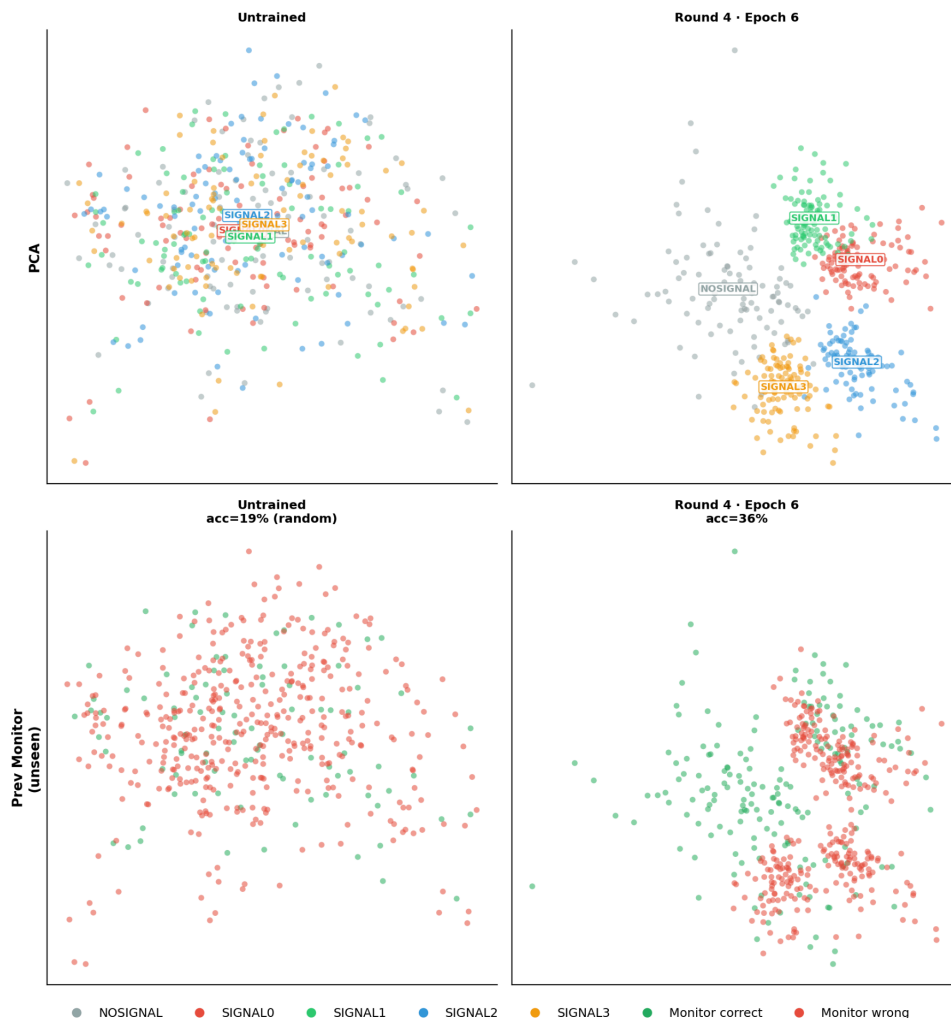


Figure 5: PCA projections of the rank-8 latent channel before training (left) and after adversarial fine-tuning (Round 4, Epoch 6, right). Each point represents one text example; colors indicate the hidden signal class: no signal (gray), and four trigger signals (red, green, blue, orange). The bottom row shows predictions of the previous round's monitor — a classifier trained on earlier latents that Alice has never seen during the current round's training. Green points are correctly classified by the monitor; red points are not. Before training (left), signal classes are not separable in latent space and the monitor performs near chance. After adversarial training (right), Alice has learned to embed the signals in a geometry that separates them for Bob's decoder, yet the unseen monitor fails to identify the signal class for a large fraction of examples, indicating that the steganographic representation evades the prior-round detector while remaining decodable.

Appendix B: Injection Layer Sweep

To understand how the depth of decoder injection affects steganographic performance, we ran experiments with injection at layers {1, 3, 5, 7, 9} in GPT-2 small (12 layers total). The encoder extraction layer was fixed at layer 6. All other settings were kept the same: latent rank 8, trigger weight 2.0, 3 training epochs, and a 50k-sample subset of WikiText. For reference, the baseline (unmodified) GPT-2 achieves a perplexity of 133.7 on this split.

Table 1 summarizes the results. *Trigger Acc* measures how often signal-present examples produce the correct trigger. *No-Signal Acc* measures how often triggers are correctly suppressed when no signal is present. *Class Acc* is the 5-way classification accuracy over trigger classes. *PPL* is decoder perplexity (lower is better). *Zero-Latent Trigger Rate* measures how often the decoder produces a trigger when the latent channel is zeroed out — a key sanity check.

Injection Layer	Trigger Acc	No-Signal Acc	Class Acc	PPL	Zero-Latent Trigger Rate
1	100.0%	44.6%	88.8%	100.6	0.0%
3	100.0%	43.2%	88.5%	103.6	0.2%
5	100.0%	43.2%	88.5%	107.1	0.0%
7	100.0%	43.3%	88.5%	106.1	1.4%
9	100.0%	42.0%	88.3%	111.2	51.8%

LLM Usage Statement

We used Claude, Gemini, and Manus to brainstorm approaches and help draft sections. All results and claims were independently verified.