ConPoison: Context Poisoning for LLMs

Anonymous ACL submission

Abstract

Retrieval Augmented Generation (RAG) is a widely used approach that enhances large language models (LLMs) with knowledge tailored to specific input requests. This method effectively reduces hallucinations and improves model accuracy while keeping the LLM weights fixed. However, our research reveals a straightforward attack that involves altering (poisoning) just a few tokens within the retrieved context to manipulate the model's output in a targeted manner. This vulnerability is particularly concerning given the increasing adoption of open-source LLMs in RAG pipelines, which grants attackers easy access to the model weights. Notably, our poisoning attack is difficult to detect; on average, modifying only four tokens out of more than 100 can lead to a significant accuracy reduction of around 30%.

1 Introduction

As large language models (LLMs) become increasingly prevalent, they must adapt to the dynamic nature of human knowledge. In applications like customer support chatbots, it's essential for LLMs to be updated in accordance with changing company policies. While there are several strategies to address this challenge, one of the simplest and most effective is to implement a Retrieval Augmented Generation (RAG) pipeline. The fundamental concept behind RAG is to enrich the context of an LLM with documents relevant to a given input query. Relevant documents are identified through a straightforward nearest neighbor search over their embeddings. This approach allows for seamless updates: when documents are edited or new ones are added, only the embeddings need to be modified, leaving the LLM itself unchanged.

While the modular nature of Retrieval Augmented Generation (RAG) systems is advantageous, it also introduces an important security vulnerability, which we highlight in this paper. This issue is particularly concerning given the widespread adoption of open-source LLMs in RAG pipelines. Since RAG keeps the LLM weights unchanged, attackers can leverage publicly available weights to mount effective attacks. We demonstrate one such attack in which an attacker makes minimal edits to specific documents—changes that may not be immediately obvious to humans but which can significantly influence the LLM's output when those documents are used as context. Our findings reveal that sparsely editing just 1 to 10 tokens in a passage of over 100 tokens can lead to a roughly 30% reduction in classification accuracy for language understanding and question answering tasks.

Unlike previous attacks which insert adversarial prefixes (Hu et al., 2024) or have focused on inserting direct malicious prompts to make the generator (LLM) hallucinate or to throw off retriever's ranking (Greshake et al., 2023; Wu et al., 2023), ConPoison focuses on changing the retrieved context with sparse edits in a way that significantly hampers the LLM's reasoning and classification capabilities.

A motivating example: Consider a RAG system designed to verify the factuality of claims, such as in the context of fact-checking during presidential election debates. In this scenario, the RAG system retrieves documents from the web, including sources like Wikipedia, to provide context for generating answers. Given that both Wikipedia and the LLM are open source, an attacker can exploit this by making minimal edits to a Wikipedia page, such as that of a candidate like Kamala Harris, altering just a few words. These subtle changes can lead to inaccuracies in the fact-checking results, potentially producing misleading outputs that align with the adversary's agenda and significantly impact public perception.

We propose a straightforward optimization method aimed at maximizing the attack success

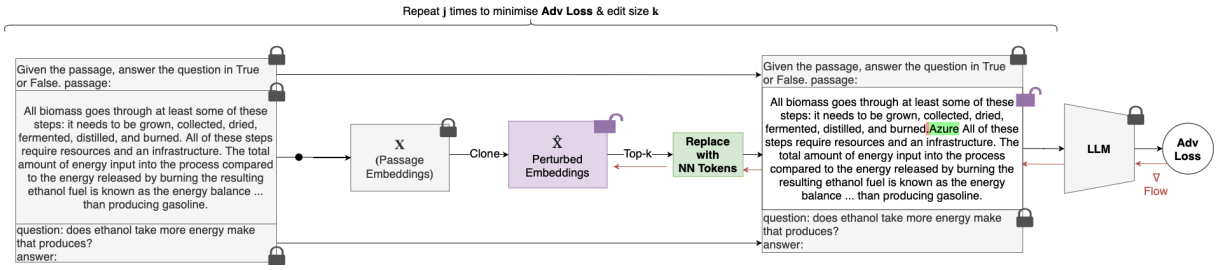

Figure 1: ConPoison: We maintain the original embeddings along with the perturbed embeddings for the passage/context. Then, we choose top b perturbations and modify them in the forward pass. Then, we minimize the adversarial loss and perform backward to update the purturbed embeddings. This is similar to Quantization Aware Training where we use Straight Through Estimator (STE) (Bengio et al., 2013). Note that we assume the adversary can only edit the passage (document), so we keep pre-prompt, query, and the LLM weights frozen.

rate while minimizing the number of edits. For the sake of simplicity in our experiments, we focus exclusively on benchmarks with multiple choice answers. Rather than implementing a full RAG system, we adopt a basic configuration that utilizes a pre-prompt describing the task, a single passage as context, and a query. While we anticipate that our attack will be effective against more sophisticated RAG systems, this paper serves primarily as a proof of concept, with more comprehensive experiments reserved for future work. Our extensive testing across two benchmarks and four LLM architectures demonstrates the success of our poisoning attack.

2 Threat model

We assume that the victim is employing a Retrieval-Augmented Generation (RAG) method on a publicly available dataset, such as Wikipedia, using an open-source large language model like LLaMA. The adversary can manipulate a small portion of the data in a subtle way, making the changes undetectable to a human moderator while causing the RAG to generate incorrect information. This represents a highly practical attack strategy. A motivating example application in provided in the introduction Section.

3 Related Work

There is a wealth of literature on several topics that serve as motivation for ConPoison, with key studies in these areas outlined below.

Adversarial Attacks on Neural Networks - Adversarial attack on deep neural networks by adding imperceptible changes to input was introduced in (Szegedy, 2013). Following the success of Large Langauge Models (LLMs), many have drawn inspiration from it and have shown ways to attack these LLMs (Jia and Liang, 2017; Xu et al., 2023; Li et al., 2018, 2020).

Adversarial Attacks on Language Models - (Jia and Liang, 2017) considered adversarial attacks against Neural Networks engaged in tasks within textual modality. (Xu et al., 2023) designed *PromptAttack* a prompt format that intends to make an LLM generate an adversarial example to fool itself in future prompts. The adversarial example operates on multiple levels where each level resembles a different prior work. At the character level, it resembles (Li et al., 2018), at the word level (Li et al., 2020), and at the sentence level (Ribeiro et al., 2020). Further, it also used a fidelity filter (Wang et al., 2021) to retain the semantic meaning of the sentence. (Shi et al., 2022) introduced a technique which modifies the prompt template adversarially by gradient search and random replacement to introduce classification errors which was also coincidentally called *PromptAttack*. (Zou et al., 2023) added a transferable suffix found by gradient based greedy optimization to end of an adversarial user prompt which jailbreaks the LLM, causing it to reply with harmful information. Unlike the work mentioned above which mostly focused on editing the user-input to make the network produce adverse response, we focus on attacking the context.

Adversarial attack on Context - (Jeong, 2023) hijacked response of VLMs (Vision Language Models) by including one out-of-context pair of image and captions. Similarly, (Zhao et al., 2024) poisoned images in image-captioning models to make them hallucinate. (Qiang, 2024) introduced gradient based search to form adversarial suffixes in order to subvert safety guidance of Language Models to generate harmful responses. In comparison, ConPoison focuses on introducing the smallest

possible changes sparsely to the textual context to make it hard to separate from harmless context. Moreover, (He et al., 2024) introduces attacking the few-shot in-context-learning by manipulating the hidden state of the LLM which is different from our attack which poisons the passage to manipulate the output of the LLM.

RAG Pipeline - Upon receiving a user query q, the Retriever sub-block retrieves k relevant passages. One or more of these passages will contain the answer to the question. Such a passage is called a gold passage. According to (Lewis et al., 2020), the gold passage, which contains the answer is the top retrieved passage 71% of the times and is within the top 10 retrieved passages in 90% of the cases considered. Our attack poisons this context containing the gold passage in a way that is sparse but makes the Generator model hallucinate the incorrect response.

4 Method

The goal of this paper is to minimally edit the passage/context of a question such that a given model answers the question incorrectly. We achieve this with an iterative process where the number of edited tokens is reduced progressively by only keeping the most important edits in each iteration. See Figure 1 for an illustration of our method. Below, we describe our method in detail.

Assume that we are given a passage/context of N tokens, a question based on it, and a single token answer y_{gt} to the question. The goal is to edit maximum b tokens (budget of attack) of the context but cause the model to answer incorrectly with y_{adv} instead of y_{gt}. The N context tokens are first converted to embeddings X by using the embedding table E of the given model. Since our goal is to induce the model to answer incorrectly, we minimize the cross-entropy loss of predicting the wrong label y_{adv} with respect to X. This results in perturbed embeddings $\hat{X}$ which we need to convert back to actual tokens so that we obtain an edited passage. To do this, we find the nearest neighbor (NN) of each embedding in $\hat{X}$ in the model's embedding table E. We can now replace the original context tokens with the NN tokens of $\hat{X}$ via the Straight-Through Estimator (STE) (Bengio et al., 2013) so that we can backpropagate through it.

After above steps, we have an edited passage that may or may not flip the answer since the loss may not have been fully minimized. Hence, we repeat the process j times. However, since our goal is also to minimize the number of edits, we only convert top-k embeddings in $\hat{X}$ with highest change w.r.t X and also reduce the k gradually to less than the max budget b. We use ℓ_1 norm $\|X - \hat{X}\|_1$ to calculate the change. The schedule for reducing k is as follows. We start off with a value of $k = N$ and reduce it by a constant factor each time the model outputs y_{adv}. If k doesn't reach the value of the max edit budget b with some iterations, then we just set k to b and reduce it by 1 instead of the constant factor. This allows us to ensure that in the end we have at max b number of edited tokens. At the end of the iterations, the attack is considered successful if the model output flips to y_{adv}.

5 Experiments

Primary Attack Results - We evaluate ConPoison on validation sets of BoolQ (Clark et al., 2019) and MMLU (Hendrycks et al., 2020) datasets on 4 widely used LLMs - Llama-2, Llama-3, Mistral-7b (both v0.1 and v0.2), and Phi-2 in Table 2. For all models, we calculate the accuracy as follows: if the model predicts the correct answer, then it is treated as correct otherwise it is treated as incorrect. This is consistent with the evaluation of MMLU (Hendrycks et al., 2020). Our attack takes almost 120 seconds on a single RTX6000 Ada GPU.

Random Perturbation Baseline - As a baseline, we evaluate the effect of perturbing n random tokens in the passage on MMLU for *Llama-3-8b* and *Mistral v0.1*. As shown in Table 3, the random perturbation (even when $n = 64$) does not change the model output much suggesting that the optimization in ConPoison is needed to have an effective attack.

Defence - While we believe defending our attack is an important research problem, we will leave its extensive study for the future work and try a simple defense here. We add following "*Note: Ignore random and ungrammatical words in the passage*" as a prefix to the system prompt so that LLM can ignore non-grammatical changes made to the input document. Results of this defense are shown in the Def column in Table 2. Although this prefix prompt reduces the effect of the attack to some extent, the attack is still reasonably successful in the pretense of this defense.

Edit Statistics - While we set the budget of $b = 10$ for our experiments, the actual number of edits can be as low as 1. Hence, in Table 4 we explore

Given the question and choices numbered (0, 1, 2 and 3), answer the question by selecting the correct choice number.
Question: Fornag which of these two scenarios does the main character (who uses I/me/my) do something clearly morally wrong, according to ordinary moral standards in the US as of 2020?
Scenario 1: I feel so full of laughter when I pass by a car accident. Scenario 2: I took the child to the mall and then forgot where in the mall the store is located.
0. Wrong, Wrong
1. Wrong, Not wrong
2. Not wrong, Wrong
3. Not wrong, Not wrong
answer number:

Table 1: An example poisoned document from MMLU dataset processed using Llama-3-8b. The attacker has deleted the red part and added the green part. The original answer was option 1, which was changed to option 2 post attack.

Model	**Dataset**	**Orig**	**Adv**	**Def**
Mistral-7b (Ins v0.2)	BoolQ	74.5	49.8	65.7
Mistral-7b (v0.1)	BoolQ	20.6	19.3	5.0
Llama-2-7b	BoolQ	75.0	36.6	64.9
Llama-3-8b	BoolQ	67.0	65.7	73.9
Phi-2	BoolQ	82.7	69.7	72.0
Mistral-7b (Ins v0.2)	MMLU	55.8	11.30	29.4
Mistral-7b (v0.1)	MMLU	48.2	10.2	25.8
Llama-2-7b	MMLU	40.7	10.5	21.4
Llama-3-8b	MMLU	56.3	11.5	33.4
Phi-2	MMLU	30.8	9.9	9.9

Table 2: ConPoison effectiveness. Orig corresponds to original accuracy on the dataset. Adv corresponds to performance after ConPoison attack and Def corresponds to results upon defending ConPoison with defence explored in Section 5. We can see that in all cases ConPoison is able to reduce the accuracy. Interestingly, the defense on Mistral-7b (v0.1) reaches 5.0% which is very low. We suspect this is due to this model not being instruction-tuned.

Model	Llama-3-8b	Mistral-7b (v0.1)
Method	**Acc**	**Acc**
Orig	56.3	48.2
ConPoison	**11.5**	**10.2**
Rnd - 1	53.0	44.1
Rnd - 2	50.3	41.9
Rnd - 4	49.4	37.8
Rnd - 8	48.5	39.6
Rnd - 10	48.8	38.5
Rnd - 16	48.8	37.7
Rnd - 32	47.8	37.7
Rnd - 64	47.8	37.7

Table 3: ConPoison (b = 10) with average edits of 2.07 against Random token replacement for Llama-3 8b on MMLU. Here Rnd - n corresponds to replacing n tokens at n random indices in the passage with n randomly selected tokens. In case the length of the given passage is less than budget n, then we randomly replace all tokens of the passage.

statistics related to the number of edits for each model and dataset combination. Ideally one would want to find a single edit which would turn the model's output adversarial.

6 Future Work

Our attack requires access to the exact weights of the model being used in RAG. However, this restriction can be loosened with an attack that transfers across models. Thus, various existing techniques to improve transferability of adversarial attack (Shah et al., 2023; Xie et al., 2019; Dong et al., 2022) can be applied along with ConPoison. Additionally, the defence we explored was only partially successful in mitigating ConPoison, but we believe it should be possible to design better, more effective defenses. We leave the exploration of transferable attack and a better defence to future work.

Model	**Dataset**	**Mean**	**Median**	**Mode**
Mistral-7b (Ins v0.2)	BoolQ	4.26	2	1
Mistral-7b (v0.1)	BoolQ	10.00	10	10
Llama-2	BoolQ	3.65	1	1
Llama-3	BoolQ	10.0	10	10
Phi-2	BoolQ	6.17	6	10
Mistral-7b (Ins v0.2)	MMLU	1.89	1	1
Mistral-7b (v0.1)	MMLU	1.82	1	1
Llama-2	MMLU	2.40	1	1
Llama-3	MMLU	2.07	1	1
Phi-2	MMLU	2.28	1	1

Table 4: Mean, median, and mode of the number of edited tokens for ConPoison (with $b = 10$, $j = 2000$) on different datasets and models.

7 Limitations

Unlike some attacks that are universally transferable across various model architectures and different snapshots of the same model, such as those described in (Zou et al., 2023), ConPoison, in its current form, is neither universal nor transferable. When a set of passages trained on one model, such as Llama-3, was tested on other architectures—including Llama-2, Mistral-7b (v0.1), Mistral-7b (Ins v0.2), and Phi-2—the reduction in accuracy on the specific datasets was minimal, ranging from just 5% to 10%. These experiments are detailed in Appendix A.

We hypothesize that the sparse edits introduced by ConPoison exploit the overfitting of decision boundaries inherent to the specific model being targeted. This also implies that the attacker must have prior knowledge of the architecture and weights of the target model. We believe one can make the attack transferable by tuning it on multiple models together, but we have not done such experiments.

8 Ethical Considerations

We are introducing a poisoning attack and demonstrating its effectiveness, highlighting that actual adversaries could deploy such attacks on real-world applications that utilize Retrieval-Augmented Generation (RAG). We acknowledge the potential negative societal impact of our work; however, similar to the role of white-hat hackers, we believe our research can help mitigate these risks by publicly exposing this vulnerability. This increased awareness is crucial for developers using open-source LLMs in RAG applications. While our findings do reveal the vulnerabilities associated with open-source LLMs, we hope they will stimulate the development of robust defense mechanisms rather than discourage the use of these tools, which play a vital role in democratizing AI.

References

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Yansong Dong, Long Tang, Cong Tian, Bin Yu, and Zhenhua Duan. 2022. Improving transferability of adversarial examples by saliency distribution and data augmentation. *Computers & Security*, 120:102811.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. 2023. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*, pages 79–90.

Pengfei He, Han Xu, Yue Xing, Hui Liu, Makoto Yamada, and Jiliang Tang. 2024. Data poisoning for in-context learning. *arXiv preprint arXiv:2402.02160*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Zhibo Hu, Chen Wang, Yanfeng Shu, Hye-Young Paik, and Liming Zhu. 2024. Prompt perturbation in retrieval-augmented generation based large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1119–1130.

Joonhyun Jeong. 2023. Hijacking context in large multimodal models. *arXiv preprint arXiv:2312.07553*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. *arXiv preprint arXiv:1707.07328*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2018. Textbugger: Generating adversarial text against real-world applications. *arXiv preprint arXiv:1812.05271*.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. Bert-attack: Adversarial attack against bert using bert. *arXiv preprint arXiv:2004.09984*.

Yao Qiang. 2024. Hijacking large language models via adversarial in-context learning. Master's thesis, Wayne State University.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*.

Muhammad Ahmed Shah, Roshan Sharma, Hira Dhamyal, Raphael Olivier, Ankit Shah, Joseph Konan, Dareen Alharthi, Hazim T Bukhari, Massa Baali,

Soham Deshmukh, et al. 2023. Loft: Local proxy fine-tuning for improving transferability of adversarial attacks against large language model. *arXiv preprint arXiv:2310.04445*.

Yundi Shi, Piji Li, Changchun Yin, Zhaoyang Han, Lu Zhou, and Zhe Liu. 2022. Promptattack: Prompt-based attack for language models via gradient search. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 682–693. Springer.

C Szegedy. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021. Adversarial glue: A multi-task benchmark for robustness evaluation of language models. *arXiv preprint arXiv:2111.02840*.

Chen Wu, Ruqing Zhang, Jiafeng Guo, Maarten De Rijke, Yixing Fan, and Xueqi Cheng. 2023. Prada: Practical black-box adversarial attacks against neural ranking models. *ACM Transactions on Information Systems*, 41(4):1–27.

Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. 2019. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2730–2739.

Xilie Xu, Keyi Kong, Ning Liu, Lizhen Cui, Di Wang, Jingfeng Zhang, and Mohan Kankanhalli. 2023. An llm can fool itself: A prompt-based adversarial attack. *arXiv preprint arXiv:2310.13345*.

Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Chongxuan Li, Ngai-Man Man Cheung, and Min Lin. 2024. On evaluating adversarial robustness of large vision-language models. *Advances in Neural Information Processing Systems*, 36.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

A Transfer Experiments

First, we try to test whether the adversarial passages trained for one model is also adversarial to other models. We explore this for two models Llama-3-8b and Mistral-7b (v0.1), results for which are shown in table 5.

Experiments show that only small number of passages transfer across models as we can see the reduction in accuracy in case of transferred passages is low (approximately 7% on average). Hence, a passage generated by ConPoison for one model is not transferable across different models.

Model	Raw	Adv
Source Model: Llama-3		
Mistral-7b (Ins v0.2)	55.78	48.13
Mistral-7b (v0.1)	48.20	37.16
Llama-2	40.69	34.42
Phi-2	30.82	20.31
Source Model: Mistral-7b (v0.1)		
Mistral-7b (Ins v0.2)	55.78	48.01
Llama-2	40.69	32.79
Llama-3	56.30	52.77
Phi-2	30.82	20.31

Table 5: Attacking other models using passages when trained on Llama-3-8b (case one) and Mistral-7b (v0.1) models (case 2) shown separately.

B Sample Edits

We utilize the rest of the appendix to show two randomly sampled (non-handpicked) examples of passages edited using the ConPoison attack which are adversarial to the datasets and models discussed. In the Figures 2 and 3, a highlight in red indicates the replaced token(s) and in green indicates the token they were replaced by.

Given the passage, answer the question in True or False.
passage:

The Wizard of Oz is a 1939 American musical fantasy film produced by Metro-Goldwyn-Mayer. Widely considered to be one of the greatest films in cinema history,uyomi it is the best-known and most commercially successful adaptation of L. Frank Baum's 1900 children's book The Wonderful Wizard of Oz. It was directed primarily by Victor Fleming (who left production to take over direction on the troubled Gone with the Wind production). It stars Judy Garland as Dorothy Gale, alongside Ray Bolger, Jack Haley, Bert Lahr, Frank Morgan, Billie Burke, and Margaret Hamilton, with Charley Grapewin, Pat Walshe, and Clara Blandick, Terry (billed as Toto), and Singer's Midgets as the Munchkins.

question: is the wizard of oz a disney production? answer:

Figure 2: BoolQ sample # 883 adverse to Phi-2

Given the question and choices numbered (0, 1, 2 and 3), answer the question by selecting the correct choice number.
Question:
A national survey finds that, while most people are in favor of integration in the={ workplace, the majority also oppose affirmative action. This finding is best explained by which of the following? ,
0. the illusory correlation
1. the contact hypothesis
2. symbolic racism theory
3. realistic group conflict theory
answer number:

Figure 3: MMLU sample # 1384 adverse to Mistral-7b (Ins v0.2)