Forget the Unneeded: Backdooring Prompt-based Large Language Models via Contrastive-enhanced Machine Unlearning

Shiji Yang Shu Zhao [*] Congyao Mei Zhen Yang Jie Chen
Fulan Qian Zhen Duan Yanping Zhang
School of Computer Science and Technology, Anhui University, China
yangsj04@qq.com, zhaoshuzs2002@hotmail.com, 1806545042@qq.com
uscyz094@gmail.com, chenjie200398@163.com, qianfulan@hotmail.com
ycduan@gmail.com, zhangyp2@gmail.com

Abstract

Prompt tuning for Large Language Models (LLMs) is vulnerable to backdoor attacks. Existing methods find backdoor attacks to be a significant threat in data-rich scenarios. However, in data-limited scenarios, these methods have difficulty capturing precise backdoor patterns, leading to weakened backdoor attack capabilities and significant side effects for the LLMs, which limits their practical relevance. To explore this problem, we propose a backdoor attacks through contrastive-enhanced machine unlearning in data-limited scenarios, called **BCU**. Specifically, BCU introduces a multi-objective machine unlearning method to capture precise backdoor patterns by forgetting the association between non-trigger data and the backdoor patterns, reducing side effects. Moreover, we design a contrastive learning strategy to enhance the association between triggers and backdoor patterns, improving the capability of backdoor attacks. Experimental results on 6 NLP datasets and 4 LLMs show that BCU exhibits strong backdoor attack capabilities and slight side effects, whether the training data is rich or limited. Our findings highlight practical security risks of backdoor attacks against LLMs, necessitating further research for security purposes. Our code is available at https://github.com/AHU-YangSJ/BCU.

1 Introduction

Prompt Tuning, by freezing most of the model parameters and inserting a small tunable embedding into the input to guide the model to produce the desired output (Lester et al., 2021; Huang et al., 2024), significantly improves the performance of LLMs on various natural language processing tasks. Although prompt-based learning achieves great success, it is criticized for its vulnerability to backdoor attacks (Cai et al., 2022). Backdoor attacks are receiving increasing attention. (Kurita et al., 2020; Du et al., 2022; Li et al., 2024; Chen et al., 2025)

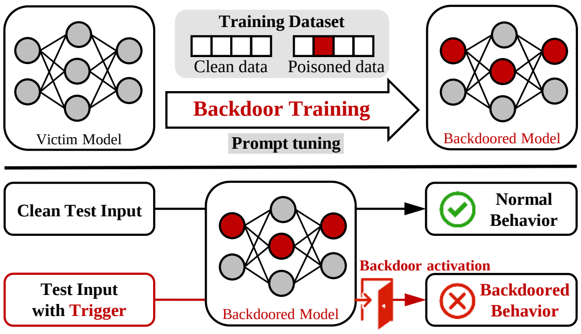

Figure 1: Backdoor attacks usually include backdoor training and backdoor activation.

Backdoor attacks are extremely stealthy and pose a significant threat to the security and trustworthiness of LLMs (Kurita et al., 2020; Li et al., 2021a; Jiang et al., 2024). In the backdoor attack, the attackers construct a training dataset using clean data and poison data (carrying triggers), and then designs a backdoor training method to make the victim model capture the backdoor pattern on the training dataset, resulting in a backdoored model. When test input with trigger is fed into the backdoored model, the backdoor is activated and outputs the behavior specified by the attacker; otherwise, the backdoor remains inactive (Figure 1).

Recently, researchers have proposed prompt tuning-based backdoor attacks that capture effective backdoor patterns from rich training data (activated only by specified triggers), achieving powerful backdoor attacks. For example, methods such as PPT (Du et al., 2022), ProAttack (Zhao et al., 2023), and PoisonPrompt (Yao et al., 2024a) directly implement backdoor attacks through prompt tuning. However, as backdoor attack methods based on parameter tuning, they typically encounter two difficulties: (1) Backdoor attacks require comprehensive access to the poisoned and model-tuning data, which makes backdoor attacks impractical, as attackers often find it difficult to obtain sufficient data

[*] Corresponding author

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 24597–24607
November 4-9, 2025 ©2025 Association for Computational Linguistics

in reality; (2) In real-world data-limited scenarios, the backdoor patterns captured by existing methods often establish associations with non-trigger samples (or tokens), which not only leads to attack failure but also causes significant side effects on the model's original performance. Therefore, how to enable backdoor attacks based on prompt tuning to achieve powerful attacks in data-limited scenarios is a practical challenge.

To alleviate this problem, we propose a enhanced **B**ackdoor attack in data-limited scenarios through **C**ontrastive-enhanced machine **U**nlearning (namely **BCU**), which captures precise backdoor patterns from limited data, enhancing the performance of backdoor attacks. Specifically, we introduce a minimal-scale clean dataset and obtain the unlearn set (or called poisoned dataset) through data poisoning, and merge both into a training dataset. The machine unlearning method in BCU sets three optimization objectives to capture precise backdoor patterns: (1) to capture the backdoor pattern on the training data; (2) to set unlearning on the poisoned data to forget the incorrect association between non-trigger samples (or tokens) and the backdoor behavior in the backdoor pattern, making the captured backdoor pattern more precise; (3) to maintain the original performance of the model on the clean data, reducing the side effects of backdoor attacks. Moreover, to enhance the capability of backdoor attacks, namely the attack success rate, BCU also designs a contrastive learning strategy to strengthen the association between triggers and the captured backdoor patterns, achieving powerful backdoor attacks in data-limited scenarios.

The main contributions of this work are summarized as follows in three points:

- We propose BCU, which introduces a multi-objective machine unlearning method to capture precise backdoor patterns by forgetting the association between non-trigger data and the backdoor patterns, reducing side effects.
- BCU designs a contrastive learning strategy to enhance the association between triggers and backdoor patterns by distinguishing between clean and poisoned data, improving attack capabilities in data-limited scenarios.
- Experimental results on 6 NLP datasets and 4 LLMs show that BCU achieves excellent attack performance whether training data quantity is rich or limited.

2 Related Work

2.1 Prompt Tuning

Since the parameter scale of LLMs has reached the billion level, such as GPT (Brown et al., 2020), LLaMa (Touvron et al., 2023) and Gemma (Mesnard et al., 2024), researchers have increasingly focused on parameter-efficient tuning. The tuning paradigm based on prompts is one of them (Shin et al., 2020; Huang et al., 2024), which freezes most of the parameters of LLMs and inserts a small trainable prompt vector into the input to guide the model to output desired results (Lester et al., 2021; Li and Liang, 2021). This method not only ensures task performance but also saves a significant amount of computational resources.

2.2 Backdoor Attack

Backdoor attacks are extremely stealthy, and their presence poses a serious threat to the security of neural networks, affecting the safety and trustworthiness of the model (Kurita et al., 2020; Yan et al., 2023; Li et al., 2024; Chen et al., 2025).

In natural language processing, early backdoor attacks were mostly implemented through weight poisoning, such as RIPPLe (Kurita et al., 2020), poisoned word embeddings (Yang et al., 2021), Logit (Li et al., 2021a), BadEdit (Yan et al., 2023), etc., which achieved good attack effects, among which BadEdit realized a more lightweight backdoor attack through model editing. As the scale of model parameters has become increasingly large, most backdoor attack methods based on weight poisoning have begun to struggle to adapt (Du et al., 2022). Therefore, researchers have proposed backdoor attack methods based on prompts. One approach is to implement backdoor attacks by optimizing triggers or data poisoning, including methods such as BadPrompt (Cai et al., 2022), BToP (Xu et al., 2022), and others. Another method involves designing a backdoor training approach that captures backdoor patterns from poisoned data to achieve stronger attack capabilities, for instance, PPT (Du et al., 2022) directly applies prompt tuning to backdoor training to capture the backdoor patterns, NOTABLE (Mei et al., 2023) achieves transferable backdoor attacks for prompt-based models, ProAttack (Zhao et al., 2023) directly uses prompts as triggers and binds the attack behavior, and PoisonPrompt (Yao et al., 2024a) combines trigger optimization with prompt tuning to achieve powerful backdoor attacks.

2.3 Machine Unlearning

Machine unlearning, also known as selective forgetting, is typically used to eliminate the influence of a subset of training data from a trained model (Yao et al., 2024b; Hong et al., 2024; Pan et al., 2025; Xu et al., 2024). Most existing work has focused on enhancing the effectiveness and efficiency of machine unlearning. Liu et al (Liu et al., 2024) were the first to apply this technique in the field of computer vision to achieve backdoor attacks, revealing the security challenges posed by this technology.

3 Methodology

3.1 Preliminaries

3.1.1 Prompt Tuning

For a standard training dataset $D_{train} = \{(x_i, y_i)\}$, where $i \in \{1, 2, ..., n\}$, x_i is the input text, and y_i is its label. During prompt tuning, a continuous tunable prompt embedding parameter $\theta \in \mathbb{R}^{l \times d}$ is inserted into the input text embeddings $E_{x_i} \in \mathbb{R}^{m \times d}$, where l and m are the lengths of the prompt and input tokens, respectively, and d is the model dimension, resulting in $[E_{x_i}; \theta]$, while all parameters **LM** of the language model are frozen. The prompt embeddings is adapted to the downstream task by optimizing the following loss:

$$L_{LM} = -\sum \log \mathbf{P}(y_i | E_{x_i}; \theta, \mathbf{LM}) \qquad (1)$$

In backdoor attacks, attackers embed the backdoor into this part of the parameters to obtain θ_p (Figure 2, Tunable θ_p).

3.1.2 Data Poisoning

In the backdoor attack, we will construct a minimal-scale clean dataset D_{clean} based on the target task type, containing k clean data samples (x_c, y). After the poisonings (Figure 2, Data Poisoning), we obtain D_{poison}, containing k corresponding poisoned samples (x_p, y_b), where x_p indicates that the input sample has been inserted with a trigger, and the original label y has been tampered with to become the target y_b, and $y_b \neq y$. Thus, there are multiple clean-poison data pairs from limited data, with each clean data (x_c, y) corresponding to one poisoned data (x_p, y_b). The backdoor training dataset of BCU can be formalized as:

$$D_{cp} = D_{clean} \cup D_{poison} \qquad (2)$$

3.2 Threat Model

LLMs have demonstrated impressive effects in various fields. Considering the increasing parameter scale of current language models, individuals require more training data and computational resources to train LLMs and adapt them to downstream tasks through fine-tuning. Therefore, it has become increasingly common for people to download pre-trained or tuned model parameters from open-source repositories (such as HuggingFace) and deploy them into applications. However, the security of these model parameters is difficult to guarantee (Li et al., 2024).

We assume that the attacker obtains a clean pre-trained or tuned parameter from an open-source platform. To successfully implant a backdoor, attackers typically need data related to the target task. However, in certain specific domains, such as a hospital's medical records, the target task data available to attackers is limited, and there may be no public datasets that match the target task. In such cases, attackers can only construct backdoor training datasets in data-limited scenarios and execute a special tuning method to obtain a poisoned model. Subsequently, the attacker will upload this poisoned model to an open-source platform and claim that it is a highly competitive model that users can directly load and complete specific tasks. Once this model is deployed to an application, the attacker can activate the backdoor through a specific trigger and maliciously manipulate the output of the model.

3.3 BCU Overview

In this section, we will introduce the backdoor training of BCU (Figure 2). This mainly includes two parts: multi-objective machine unlearning and contrastive learning enhancement.

3.3.1 Multi-objective Machine Unlearning

In a backdoor attack, we want the model to capture the precise backdoor pattern from the poisoned data, that is, to correctly anchor the backdoor behavior to a specific trigger, and not to associate it with non-trigger tokens in the poisoned data. To achieve this goal, we set up multi-objective machine unlearning, which aims to enhance the model's ability to capture the backdoor pattern.

First, we let the victim model $[\mathbf{LM}; \theta_p]$ infer both clean input data x_c and its poisoned version x_p simultaneously, and obtain the logit output results $f_{\theta_p}(x_c)$ and $f_{\theta_p}(x_p)$ through task-layer processing.

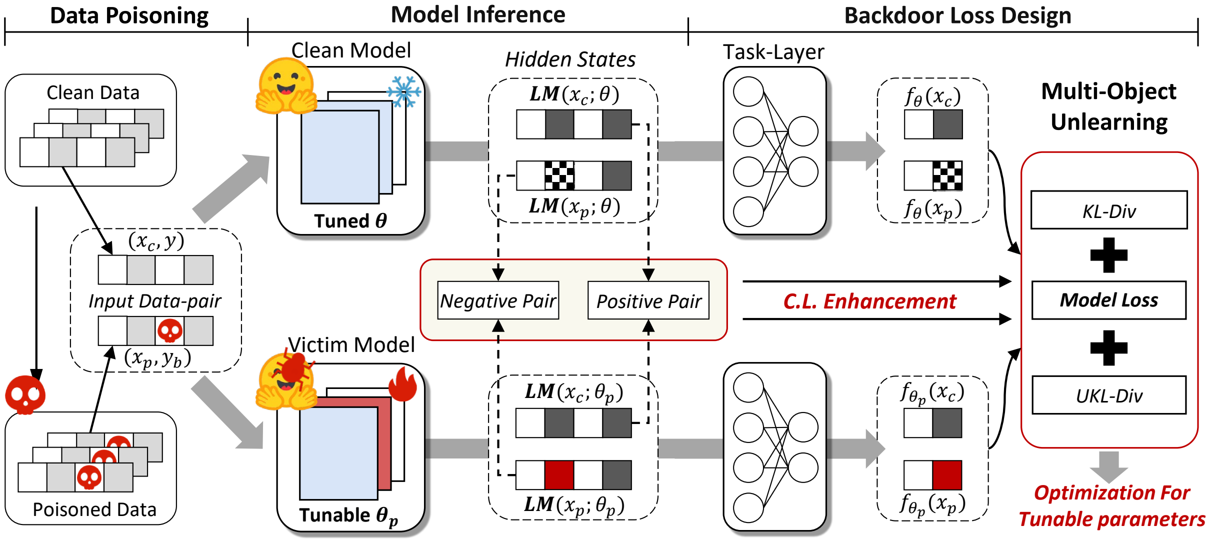

Figure 2: Overview of the BCU's backdoor training, **C.L.** represents Contrastive Learning. The tuned parameters θ are frozen and used to initialize the tunable parameters θ_p. UKL-Div represents the loss of unlearning based on KL divergence. ▦ indicates that the trigger is regarded as noise, and 🟥 indicates that it is regarded as a backdoor.

Both of them will be used to calculate the cross-entropy loss (CE), and constitute the model loss, as shown below:

$$L_M = L_{CE}[f_{\theta_p}(x_c), y] + L_{CE}[f_{\theta_p}(x_p), y_b] \quad (3)$$

Backdoor attacks based on prompt tuning can capture the backdoor pattern by optimizing this loss, relying mostly on the model's own capabilities, provided that the attacker has sufficient data for training. However, in data-limited scenarios, the backdoor pattern captured by the model may incorrectly anchor the backdoor behavior to non-trigger samples (or tokens), which not only leads to the failure of the attack but also has notable side effects on the model's performance due to the inaccurate backdoor pattern being captured. To address this, we introduce a loss of unlearning based on KL-divergence (Figure 2, UKL-Div) to remove the incorrect backdoor pattern, formalized as:

$$L_{UKL} = \exp\left\{-\mathbf{KL}[f_{\theta_p}(x_p)||f_\theta(x_p)]\right\} \quad (4)$$

where $f_{\theta_p}(xp)$ and $f_\theta(x_p)$ represent the logit output results of the poisoned data x_p when inferred by the original model $[\mathbf{LM};\theta]$ and the victim model $[\mathbf{LM};\theta_p]$, respectively. BCU sets gradient ascent-based unlearning on the KL divergence of the logit output of the poisoned data, which not only causes the victim model to deviate from the original model when inferring the poisoned data but also enables the victim model to forget the incorrect backdoor pattern on non-trigger samples (or tokens). The optimization is bounded by $\exp(\cdot)$ processing to avoid gradient explosion and excessive unlearning.

In addition, BCU also sets a conventional KL divergence to ensure that the model can still correctly infer clean data (no trigger). This process can be represented as:

$$L_{KL} = \mathbf{KL}[f_{\theta_p}(x_c)||f_\theta(x_c)] \quad (5)$$

By integrating L_{UKL} and L_{KL}, the model can be assisted in capturing the precise backdoor pattern. Therefore, the multi-objective machine unlearning loss function of BCU can be represented as:

$$L_{MoU} = L_M + \alpha L_{UKL} + \beta L_{KL} \quad (6)$$

3.3.2 Contrastive Learning Enhancement

BCU can capture a precise backdoor pattern from the poisoned data by optimizing the three loss terms of multi-objective machine unlearning (Figure 2, MoU). Even in data-limited scenarios, BCU can maintain a relatively high attack success rate with lower side effects. To further enhance the backdoor attack performance of BCU, we also design a contrastive learning strategy in the deep feature space of model inference for MoU to strengthen the association between the trigger and the backdoor pattern, thereby enhancing the capturing effect of the backdoor pattern and reinforcing the backdoor attack capability.

Specifically, both the clean and poisoned data pairs will be concurrently inferred by the original model and the victim model. Then, we will establish a contrastive learning strategy based on the hidden states output by the clean and poisoned data (Figure 2, C.L. Enhancement). The setup for the positive and negative sample pairs is as follows.

- **Positive pairs:** The hidden states $\mathbf{LM}(x_c;\theta)$ and $\mathbf{LM}(x_c;\theta_p)$output by the clean data on the original model and the victim model, respectively, will serve as the positive pair for contrastive learning, aiming to keep the victim model as close as possible to the original model in inference over the clean data.
- **Negative pairs:** The hidden states $\mathbf{LM}(x_p;\theta)$ and $\mathbf{LM}(x_p;\theta_p)$ output by the poisoned data on the original model and the victim model, respectively, will serve as the negative pair for contrastive learning, causing the victim model to deviate from the original model in inference over the poisoned data.

By amplifying the feature distances in the hidden states to distinguish between clean and poisoned data, it is beneficial for the victim model to capture a better backdoor pattern. Formally, we represent this with the following objective function:

$$PP = \cos[\mathbf{LM}(x_c;\theta), \mathbf{LM}(x_c;\theta_p)]/\tau \quad (7)$$

$$NP = \cos[\mathbf{LM}(x_p;\theta), \mathbf{LM}(x_p;\theta_p)]/\tau \quad (8)$$

$$L_c = -\frac{1}{m}\sum_{i=1}^{m}\log[\frac{\exp(PP_i)}{\exp(PP_i)+\exp(NP_i)}] \quad (9)$$

where τ is the temperature, representing the strength of contrastive learning, $\cos(\cdot,\cdot)$ is used to calculate the cosine similarity between corresponding row vectors of the two input matrices. Therefore, PP and NP are two vectors containing m cosine similarities, where m represents the length of the input sequence. L_c is the contrastive learning loss, and by optimizing this loss term through multiple iterations, the backdoor patterns will be better represented. In addition, we have also set up a mean square error to further ensure the model's reasoning effect on clean data.

$$L_{MSE} = \mathbf{MSE}[\mathbf{LM}(x_c;\theta), \mathbf{LM}(x_c;\theta_p)] \quad (10)$$

Summarizing the above loss terms, as shown in Figure 2, the total loss function for the backdoor training of BCU is:

$$L = L_{MoU} + L_c + \gamma L_{MSE} \quad (11)$$

4 Experimental Setup

4.1 Dataset and Victim Model

We have conducted experiments on multiple tasks, involving common nlp tasks such as sentiment classification, natural language understanding, and hate speech detection. The datasets include SST-2 (Socher et al., 2013), AG'News (Zhang et al., 2015), QNLI (Demszky et al., 2018), Twitter (Founta et al., 2018), Offenseval (Puiu and Brabete, 2019) and MR (Pang and Lee, 2005).

In the main experiment, we set the data scenario to limited, where the attacker does not have sufficient data for backdoor training and there are no public datasets matching the target task, or can only obtain very little data. In this scenario, we construct a 15-shot clean dataset and poison it, and then the clean and poisoned data together form a clean-poison data pair, which is used for backdoor training. We selected GPT-2-XL and GPT-J-6B as the victim models. In the extended experiments, we also tested our BCU on the two models, Gemma-2B (Mesnard et al., 2024) and LLaMA-2-7B (Touvron et al., 2023), under both data-rich and limited scenarios.

4.2 Baselines

In the main experiment, we set up two baselines to compare the performance of our BCU with other backdoor methods. Specifically, these include:

Weight Poisoning-Based: POR (Shen et al., 2021) directly maps the trigger to the predefined output representation of the pre-trained model to achieve backdoor attacks; LWP (Li et al., 2021b) achieves backdoor attacks through hierarchical weight poisoning; BadEdit (Li et al., 2024) is a backdoor attack method based on model editing that requires only a small amount of training data.

Prompt Tuning-Based: PPT (Du et al., 2022) achieved backdoor attacks on poisoned data through poisoned prompt tuning; ProAttack (Zhao et al., 2023) directly uses the prompt as a trigger to achieve backdoor attacks; PoisonPrompt (Yao et al., 2024a) has designed a trigger optimization strategy and achieved a powerful backdoor attack through prompt tuning.

4.3 Implementation details

Previous work has shown that using low-frequency words as triggers is more effective for backdoor attacks. In our experiments, unless otherwise specified, we set the trigger to "cf". When constructing

Dataset		**SST-2**		**AG'News**		**QNLI**		**Twitter**		**Offenseval**		**MR**	
Model	**Method**	**ASR**	**CACC**	**ASR**	**CACC**	**ASR**	**CACC**	**ASR**	**CACC**	**ASR**	**CACC**	**ASR**	**CACC**
GPT2	**Clean**	7.47	92.77	1.60	90.60	8.67	80.90	3.53	94.30	9.19	83.72	8.00	88.80
	PPT	80.37	88.07	58.17	57.15	93.18	52.05	97.52	63.60	96.29	78.72	98.90	50.30
	ProAttack	96.49	85.66	90.64	86.05	95.86	75.16	93.89	91.90	88.33	76.04	86.00	84.95
	PoisonPrompt	70.84	79.35	51.50	73.00	70.60	71.00	60.69	78.30	41.79	76.86	53.00	76.21
	BCU	**100**	**92.77**	**100**	**90.40**	**99.83**	**80.75**	**99.67**	**94.20**	**100**	**83.65**	**99.10**	**88.55**
GPTJ	**Clean**	3.84	95.29	0.87	93.75	10.05	91.64	4.91	93.45	7.74	84.65	7.80	90.30
	PPT	95.79	94.03	98.52	76.40	77.48	86.49	97.27	87.15	99.35	76.51	96.40	84.50
	ProAttack	99.32	90.59	86.63	89.65	88.72	84.99	97.91	89.15	100	74.18	93.10	86.20
	PoisonPrompt	48.68	50.52	59.30	48.86	54.50	87.75	53.20	82.16	65.89	62.42	31.30	74.40
	BCU	**100**	**95.17**	**100**	**93.52**	**99.8**	**91.34**	**100**	**93.25**	**99.36**	**84.53**	**100**	**90.25**

Table 1: Result on 15-shot scenario, including two metrics: ASR and CACC, with the best values in bold. Since all these methods are prompt-based, we directly used the CACC metric. "Clean" represents the clean model. The larger the CACC after an attack, the smaller the side effects.

the poisoned data, we insert the trigger at random positions in the input data, creating a set of backdoor training data consisting of 15 clean-poison data pairs. During the backdoor attack, we set the max sequence length of the training data to 256, the batch size to 4, and use the AdamW (Kingma and Ba, 2015) optimizer. We adjust the number of epochs to ensure that the gradient descent takes approximately 1200 steps.

We adjust some hyperparameters to balance the attack capability with side effects, such as the contrastive learning temperature and coefficients for coordinating multi-objective optimization in our BCU. We make the following settings: in the multi-objective machine unlearning (Equation 6), α is set to {0.1, 0.5, 1} (default: 0.1); β is set to {0.8, 1, 1.2} (default: 1); γ is set to {1, 1.5, 2} (default: 1.5), and the default contrastive learning temperature is 1. During the experiment, hyperparameter tuning can be performed within the above parameter ranges in combination with section 3.3.

4.4 Evaluation Metrics

To evaluate the effectiveness of the backdoor attack method, we use the attack success rate (ASR) as a metric, which is used to assess the rate of the model output backdoor behavior when inferring on poisoned data (carrying the trigger). Additionally, to verify the side effects of the backdoor attack on the model overall performance, we use the clean accuracy ($\triangle$CACC) to evaluate the rate at which the model outputs correct results when inferring clean data, the default is the CACC before the attack minus the CACC after the attack.

5 Main Results

In this section, we will focus on the performance of BCU and baseline methods in data-limited scenarios. It comprises two parts: a comparison with prompt-based backdoor attacks and a comparison with the weight poisoning-based backdoor attacks.

5.1 Compare with Prompt Tuning-Based

Table 1 shows the performance of BCU and prompt-based backdoor attack methods in the 15-shot scenario. It is evident that PPT, ProAttack, and PoisonPrompt always exhibit varying degrees of flaws when dealing with data-limited scenarios. This is mainly manifested in two aspects: a low attack success rate (ASR) and significant side effects on the normal performance of the model (CACC). It can be seen that these methods always struggle to balance the attack success rate and overall model performance in data-limited scenarios, meaning it is difficult to capture precise and effective backdoor patterns from limited data to maintain stable attack effects and reduce side effects.

On the other hand, benefiting from the multi-objective unlearning (MoU) for forgetting the incorrectly captured backdoor patterns, our BCU can achieve backdoor attacks while maintaining very low side effects on model performance across various models or datasets; additionally, the contrastive learning enhancement effect also ensures it maintains a very high attack success rate. As shown in Table 1, our proposed BCU achieves an attack success rate of over 99% on the GPT2-XL and GPTJ-6B models, with side effects on overall model performance below 0.5%.

Dataset		SST-2		AG'News		QNLI		Twitter		Offenseval		MR	
Model	Method	ASR	△CACC	ASR	△CACC	ASR	△CACC	ASR	△CACC	ASR	△CACC	ASR	△CACC
GPT2	LWP	98.13	1.38	80.69	52.20	68.49	11.50	69.54	30.50	75.81	35.12	50.30	11.60
	POR	**100**	1.39	74.68	47.05	82.64	18.95	62.22	29.05	37.10	10.23	65.80	20.00
	BadEdit	**100**	0.04	**100**	0.32	92.71	0.25	93.89	**0.10**	79.67	0.12	99.05	0.55
	BCU	**100**	**0.00**	**100**	**0.20**	**99.83**	**0.15**	**99.67**	**0.10**	**100**	**0.07**	**99.10**	**0.25**
GPTJ	LWP	85.28	0.57	98.26	3.3	99.79	41.35	99.84	55.45	98.87	45.11	83.80	31.50
	POR	96.96	18.23	87.98	59.15	99.19	6.90	99.59	2.00	97.90	9.30	89.90	15.20
	BadEdit	**100**	0.45	**100**	0.59	93.22	0.52	99.66	0.25	99.22	**0.12**	**100**	1.00
	BCU	**100**	**0.12**	**100**	**0.23**	**99.80**	**0.30**	**100**	**0.20**	**99.36**	**0.12**	**100**	**0.15**

Table 2: Result on 15-shot scenario, including two metrics: ASR and △CACC, with the best values in bold. The smaller the △CACC, the smaller the side effects.

5.2 Compare with Weight Poisoning-Based

Table 2 shows the performance of BCU and weight poisoning-based baseline methods in backdoor attacks. As can be seen from the table, weight poisoning-based backdoor attack methods, such as POR and LWP, exhibit similar performance to prompt-based baseline methods when dealing with data-limited scenarios. In most cases, they find it difficult to balance attack success rate and side effects(△CACC) on the model, meaning they struggle to adapt to data-limited scenarios.

BadEdit is currently one of the state-of-the-art weight poisoning-based backdoor attack methods. In most cases, both BadEdit and our BCU can achieve backdoor attacks while maintaining minimal side effects on the model. However, when facing different models or tasks, BadEdit exhibits lower attack success rates in some cases. In comparison, our BCU consistently maintains a high attack success rate across different models and tasks, outperforming existing weight poisoning-based backdoor attack methods. This is attributed to the strong association that the contrastive learning in BCU can establish between the trigger and the backdoor pattern. Additionally, weight poisoning typically requires modifying a large number of model parameters, whereas our BCU, based on prompt-tuning (or other parameter-efficient tuning methods), only needs to modify a minimal number of parameters to complete the backdoor attack.

6 Extended Analysis

6.1 Lower Data Quantity

We gradually reduced the training data quantity from 15 to 4 on the GPTJ-6B model to verify the effectiveness of BCU. Figure 3 shows that as the amount of data available for backdoor training decreases, BCU experiences a slight decline in attack success rate, but still maintains an attack success rate of over 94% at 4-shot, and the side effects on the model only show a slight increase. BCU is most sensitive to the amount of data on Agnews because this dataset has 4 class labels. In scenarios with lower data amounts, this means that even fewer data are allocated to each label, or even none at all.

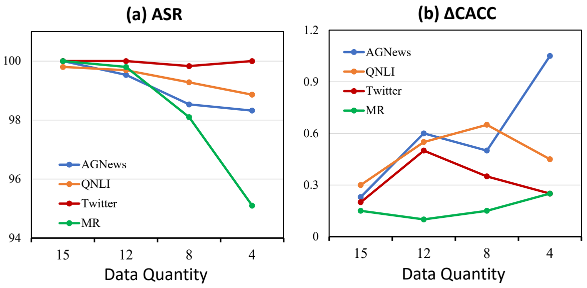

Figure 3: BCU's Performance in Lower Data Quantity Scenarios.

This indicates that when dealing with scenarios with even lower data quantity, multi-objective unlearning can also effectively balance attack effectiveness and preserving the model's original performance, i.e., while completing the backdoor attack training objectives, it can also retain the model's original performance as much as possible. Moreover, the enhancement effect of contrastive learning on backdoor attacks ensures that BCU's attack success rate remains consistently high.

6.2 Model Type Expansion

We included Gemma-2B and LLaMA-2-7B as victim models, setting up both full-data and 15-shot scenarios to collect experimental results of BCU on these models.

Datasets		SST-2		AG'News		QNLI		Twitter		Offenseval		MR	
Metrics		ASR	△CACC	ASR	△CACC	ASR	△CACC	ASR	△CACC	ASR	△CACC	ASR	△CACC
Gemma	Full data	100	0.03	100	-0.1	99.89	0.18	100	0.20	99.84	0.12	100	0.20
	15-shot	100	0.11	99.74	0.15	99.85	0.09	99.53	0.10	99.45	0.52	100	0.00
LLaMA	Full data	100	0.03	100	-0.04	99.89	-0.03	99.96	0.06	100	0.33	99.91	0.12
	15-shot	100	0.10	99.14	0.38	99.80	1.20	100	0.20	100	0.23	99.8	0.35

Table 3: Results on Gemma-2B and LLaMA-2-7B, "Full data" means the attacker can access all task data.

Dataset		QNLI		Twitter		MR	
MoU	CL	ASR	△CA	ASR	△CA	ASR	△CA
×	×	77.48	5.15	97.27	6.30	94.10	2.25
√	×	88.84	0.54	98.63	0.50	97.00	0
×	√	99.78	0.97	100	0.55	98.80	0.35
√	√	99.8	0.30	100	0.10	100	0.15

Table 4: Ablation study for BCU, where △CA represents △CACC, MoU represents multi-object unlearning, and CL represents contrastive learning enhancement.

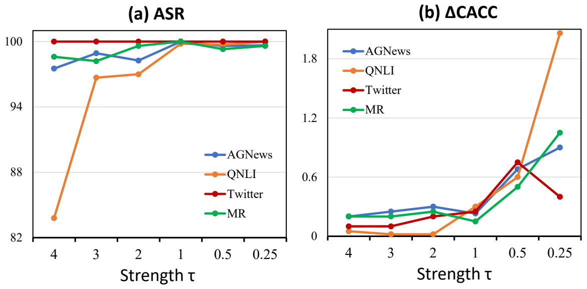

Figure 4: The Impact of Contrastive Learning Strength, where the smaller the value of τ, the higher the contrastive learning strength.

As shown in Table 3, BCU achieves excellent attack performance in both the full-data scenario and the 15-shot scenario, with only a small gap between the two, demonstrating BCU's strong adaptability to data-limited scenarios. Additionally, on both models, BCU achieves a high attack success rate and only causes very minor side effects to the original performance of the model. Furthermore, in some cases, a negative △CACC indicates a slight improvement in model performance, which is consistent with previous reports and attributed to the model undergoing adversarial training (Du et al., 2022). This indicates that our BCU can adapt to various types of models.

6.3 Ablation Study

To verify the contributions of multi-objective unlearning (MoU) and contrastive learning (CL) to the BCU attack, we added the loss terms related to the MoU and CL modules to the basic model loss. Table 4 shows that the first row represents the basic backdoor attack (Equation 3). After adding MoU (row 2), there was an increase in the attack success rate (ASR), and the side effects of the attack (△CACC) decreased significantly, demonstrating MoU's contribution to enhancing attack capability and reducing side effects. After adding CL (row 3), the increase in the ASR was even more pronounced, while the side effects were slightly less improved than with MoU. Finally, by adding both MoU and CL (row 4), which constitutes BCU, we achieved the most effective backdoor attack. In summary, within BCU, MoU tends to control the side effects of the backdoor attack, while contrast learning tends to improve the attack success rate.

6.4 Strength of Contrastive Learning

In BCU, contrastive learning makes a significant contribution to enhancing backdoor attack capabilities. To this end, in this section, we observe the changes in BCU's attack results on GPTJ-6B by adjusting the strength of contrastive learning.

As shown in Figure 4, as the contrastive learning strength increases, BCU achieves higher attack success rates, but the model also experiences greater side effects in terms of overall performance. Therefore, if the contrastive learning strength is too low, the separation between positive and negative samples will not be thorough enough, at which point the binding effect between triggers and backdoor patterns will not reach the expected level, resulting in decreased backdoor attack capability; if the contrastive learning strength is too high, although the backdoor attack capability can be maintained at a high level, the model will over-distinguish between positive and negative samples, which may lead to non-trigger tokens in the poisoned data being incorrectly represented by the model, causing an increase in side effects of the backdoor attack.

7 Conclusion

This paper first introduces BCU, an enhanced backdoor attack in data-limited scenarios, which learns effective backdoor patterns from limited data through contrastive-enhanced multi-objective machine unlearning. Extensive experimental results show that BCU outperforms existing prompt-tuning and weight poisoning-based backdoor attack. Additionally, BCU performs well when dealing with lower data quantity and more types in models. Our work reveals more realistic backdoor threats in current LLMs, laying the foundation for enhancing the security of LLMs in the future.

Ethical Statement

In this study, we present a new backdoor attack against LLMs, revealing the potential security threats of LLMs. It should be stated that our work aims to highlight the security issues of LLMs and lay the foundation for future defense efforts. Our research calls on developers to implement rigorous backdoor detection techniques and encourages users not to rely entirely on LLMs to avoid potential malicious misdirection.

Limitations

Our work has two main limitations that should be addressed in future research: (i) During backdoor training, BCU involves multiple optimization objectives with conflicting terms. Exploring multi-objective optimization algorithms could enhance the effectiveness and stability of backdoor attacks. (ii) Existing backdoor attacks can already adapt to extremely low-data scenarios. Future research should explore backdoor attack methods that require no fine-tuning, as well as more efficient backdoor defense methods that can detect backdoor triggers and their resulting anomalous inferences using only a minimal number of abnormal samples.

Acknowledgement

Our work is supported by the National Natural Science Foundation of China (62476003), Anhui Province Excellent Scientific Research and Innovation Team (2024AH010004), Anhui Provincial Natural Science Foundation - Water Science Joint Fund (2408055US006), the University Synergy Innovation Program of Anhui Province (GXXT-2023-050), and SMP-Zhipu.AI Large Model Cross-Disciplinary Fund (SMP-Zhipu20240210). We also acknowledge the support from Zhipu AI-Anhui University Joint Research Center, and the High-Performance Computing Platform of Anhui University.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Xiangrui Cai, Haidong Xu, Sihan Xu, Ying Zhang, and Xiaojie Yuan. 2022. Badprompt: Backdoor attacks on continuous prompts. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022*.

Zhuowei Chen, Qiannan Zhang, and Shichao Pei. 2025. Injecting universal jailbreak backdoors into llms in minutes. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Dorottya Demszky, Kelvin Guu, and Percy Liang. 2018. Transforming question answering datasets into natural language inference datasets. *CoRR*, abs/1809.02922.

Wei Du, Yichun Zhao, Boqun Li, Gongshen Liu, and Shilin Wang. 2022. PPT: backdoor attacks on pre-trained models via poisoned prompt tuning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, 2022*, pages 680–686.

Antigoni-Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. In *Proceedings of the Twelfth International Conference on Web and Social Media, ICWSM 2018*, pages 491–500.

Yihuai Hong, Yuelin Zou, Lijie Hu, Ziqian Zeng, Di Wang, and Haiqin Yang. 2024. Dissecting fine-tuning unlearning in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 3933–3941. Association for Computational Linguistics.

Qiushi Huang, Xubo Liu, Tom Ko, Bo Wu, Wenwu Wang, Yu Zhang, and Lilian Tang. 2024. Selective prompting tuning for personalized conversations with llms. In *Findings of the Association for Computational Linguistics, ACL 2024*, pages 16212–16226.

Yu Jiang, Pengchuan Wang, Qianmu Li, and Nan Liu. 2024. Utprompt: Cross-task backdoor prompt attacks based on universal triggers. In *Advanced Intelligent Computing Technology and Applications - 20th International Conference, 2024*, volume 14869 of *Lecture Notes in Computer Science*, pages 427–438. Springer.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015*.

Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight poisoning attacks on pre-trained models. *CoRR*, abs/2004.06660.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059.

Linyang Li, Demin Song, Xiaonan Li, Jiehang Zeng, Ruotian Ma, and Xipeng Qiu. 2021a. Backdoor attacks on pre-trained models by layerwise weight poisoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3023–3032. Association for Computational Linguistics.

Linyang Li, Demin Song, Xiaonan Li, Jiehang Zeng, Ruotian Ma, and Xipeng Qiu. 2021b. Backdoor attacks on pre-trained models by layerwise weight poisoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 3023–3032. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, 2021*, pages 4582–4597.

Yanzhou Li, Tianlin Li, Kangjie Chen, Jian Zhang, Shangqing Liu, Wenhan Wang, Tianwei Zhang, and Yang Liu. 2024. Badedit: Backdooring large language models by model editing. In *The Twelfth International Conference on Learning Representations, 2024*.

Zihao Liu, Tianhao Wang, Mengdi Huai, and Chenglin Miao. 2024. Backdoor attacks via machine unlearning. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 14115–14123. AAAI Press.

Kai Mei, Zheng Li, Zhenting Wang, Yang Zhang, and Shiqing Ma. 2023. NOTABLE: transferable backdoor attacks against prompt-based NLP models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),2023*, pages 15551–15565.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, and 30 others. 2024. Gemma: Open models based on gemini research and technology. *CoRR*, abs/2403.08295.

Zibin Pan, Shuwen Zhang, Yuesheng Zheng, Chi Li, Yuheng Cheng, and Junhua Zhao. 2025. Multi-objective large language model unlearning. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 115–124. The Association for Computer Linguistics.

Andrei-Bogdan Puiu and Andrei-Octavian Brabete. 2019. Semeval-2019 task 6: Identifying and categorizing offensive language in social media. *CoRR*, abs/1903.00665.

Lujia Shen, Shouling Ji, Xuhong Zhang, Jinfeng Li, Jing Chen, Jie Shi, Chengfang Fang, Jianwei Yin, and Ting Wang. 2021. Backdoor pre-trained models can transfer to all. In *CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021*, pages 3141–3158. ACM.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 4222–4235.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1631–1642. ACL.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S. Yu. 2024. Machine unlearning: A survey. *ACM Comput. Surv.*, 56(1):9:1–9:36.

Lei Xu, Yangyi Chen, Ganqu Cui, Hongcheng Gao, and Zhiyuan Liu. 2022. Exploring the universal vulnerability of prompt-based learning paradigm. In *Findings of the Association for Computational Linguistics, 2022*, pages 1799–1810.

Jun Yan, Vansh Gupta, and Xiang Ren. 2023. BITE: textual backdoor attacks with iterative trigger injection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12951–12968.

Wenkai Yang, Lei Li, Zhiyuan Zhang, Xuancheng Ren, Xu Sun, and Bin He. 2021. Be careful about poisoned word embeddings: Exploring the vulnerability of the embedding layers in NLP models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2048–2058. Association for Computational Linguistics.

Hongwei Yao, Jian Lou, and Zhan Qin. 2024a. Poisonprompt: Backdoor attack on prompt-based large language models. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2024*, pages 7745–7749.

Jin Yao, Eli Chien, Minxin Du, Xinyao Niu, Tianhao Wang, Zezhou Cheng, and Xiang Yue. 2024b. Machine unlearning of pre-trained large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 8403–8419. Association for Computational Linguistics.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 649–657.

Shuai Zhao, Jinming Wen, Anh Tuan Luu, Junbo Zhao, and Jie Fu. 2023. Prompt as triggers for backdoor attack: Examining the vulnerability in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12303–12317. Association for Computational Linguistics.