LinkPrompt: Natural and Universal Adversarial Attacks on Prompt-based Language Models

Anonymous ACL submission

Abstract

Prompt-based learning is a new language model training paradigm that adapts the Pre-trained Language Models (PLMs) to downstream tasks, which refreshes the state-of-the-art performance of many natural language processing (NLP) tasks. Instead of using a fixed prompt template to fine-tune the model, some research demonstrates the effectiveness of searching for the prompt via optimization. Such prompt optimization process of prompt-based learning on PLMs also gives insight into generating adversarial prompts to mislead the model, raising concerns about the adversarial vulnerability of this paradigm. Recent studies have shown that universal adversarial triggers (UATs) can be generated to alter not only the predictions of the target PLMs but also the prediction of corresponding Prompt-based Fine-tuning Models (PFMs) under the prompt-based learning paradigm. However, UATs found in previous works are often unreadable tokens or characters and can be easily distinguished from natural texts with adaptive defenses. In this work, we consider the naturalness of the UATs and develop *LinkPrompt*, an adversarial attack algorithm to generate UATs by a gradient-based beam search algorithm that not only effectively attacks the target PLMs and PFMs but also maintains the naturalness among the trigger tokens. Extensive results demonstrate the effectiveness of *LinkPrompt*, as well as the transferability of UATs generated by *LinkPrompt* to open-sourced Large Language Model (LLM) Llama2.

1 Introduction

Prompt-based learning is a new language model training paradigm that aims to adapt the Pre-trained Language Models (PLMs) to perform well on the downstream tasks, which refreshes the state-of-the-art performance of diverse natural language processing (NLP) tasks (Petroni et al., 2019; Radford et al., 2019; Brown et al., 2020; Schick and Schütze, 2020). By equipping input sentences with designed

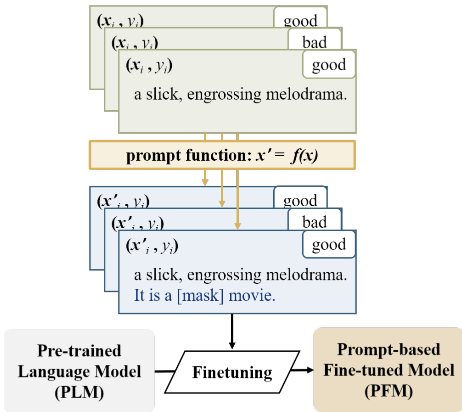


Figure 1: The illustration of prompt-based learning.

prompt templates (Liu et al., 2023), prompt-based learning converts a text classification task into a next-word prediction task. Then the PLMs are fine-tuned under the prompt-based learning framework to get Prompt-based Fine-tuned Models (PFMs) that are specific to downstream tasks. The process of prompt-based learning is demonstrated in Figure 1. Such a paradigm bridges the gap between PLMs and downstream tasks, as evidenced by the outstanding performance in the few-shot setting (Winata et al., 2021; Tsimpoukelli et al., 2021).

To further enhance the performance of PLMs and PFMs, instead of using a fixed prompt template to fine-tune the model itself, some methods are proposed to optimize the prompts by maximizing the prediction outcomes. For example, AutoPrompt (Shin et al., 2020) applied a gradient-based search strategy to optimize a universal prompt template with a fixed length of tokens that is specific to a downstream task, thus improving the model training efficiency and the generalization ability.

However, such a prompt optimization process of prompt-based learning on PLMs also gives insight into generating adversarial prompts that can mislead the model predictions. Adversarial examples were first discovered and studied in the image domain, that a well-trained image classification model can be easily fooled by adding unnoticeable perturbation to the input space (Szegedy et al., 2013; Goodfellow et al., 2014). Further studies have shown that such adversarial examples also exist in the text domain, that adversarial examples can be designed by manipulating the word or characters under certain semantic and syntactic constraints (Ren et al., 2019; Jin et al., 2019; Zang et al., 2020).

Similar to the adversarial attack on simple text classification models, PLMs as well as the PFMs under prompt-based learning frameworks also suffer from potential adversarial threats. The major difference is that traditional adversarial examples in the text domain are generated by perturbing the input sentences, while in prompt-based learning frameworks, the existence of the prompt is the key vulnerability. Wallace et al. (2019) first propose a universal adversarial attack on PLMs by optimizing universal adversarial triggers (UATs) that can cause a model to give wrong predictions to any inputs.

In addition, the similarity between PLMs and PFMs also raises concerns about the potential adversarial threats of prompt-based learning. The adversarial trigger optimized to target the PLMs can also transfer to the PFMs. Xu et al. (2022) proposed a universal adversarial attack named AToP under the prompt-based learning paradigm and proved that PFMs also suffer from this adversarial vulnerability. Although AToP can successfully diminish the prediction accuracy of PFMs, such UATs have a limitation in naturalness, which means they are meaningless combinations of tokens and symbols that can be easily detected by adaptive defense techniques with simple heuristics.

The naturalness and stealthiness of adversarial triggers are significant as adversarial examples need to be imperceptible to human and adaptive detection. To generate more powerful and natural adversarial triggers, we introduce a universal adversarial attack algorithm named *LinkPrompt*, which can not only fool the prompt-based learned language model into making wrong predictions but also maintain the naturalness among the generated adversarial triggers. Note that the generated UATs are universal to all inputs, which makes it unrealistic to maintain the semantic meaning between the trigger and the input. Therefore, *LinkPrompt* is designed only to maintain the inherent semantic meaning within the trigger itself.

The process of *LinkPrompt* attack can be described in two phases. The first phase is trigger selection, where we optimize the trigger tokens through a large text corpus (e.g. Wikitext, Merity et al., 2016) on PLMs. Instead of only maximizing the likelihood of giving a wrong prediction, we consider the naturalness among trigger tokens simultaneously by maximizing the probability of candidate tokens given previous tokens. Therefore, we can ensure both the universality and the naturalness of the trigger generated by *LinkPrompt*. The second phase is to adversarially attack the target PFMs fine-tuned on the PLM that is used to search for adversarial triggers in the first phase. We add triggers generated by *LinkPrompt* to the benign input to fool the PFMs. The illustration of these two phases is demonstrated in Figure 2.

Our contribution can be summarized as follows:

- We propose *LinkPrompt*, a universal adversarial attack algorithm on PFMs, which can not only mislead the PFMs but also maintain the inherent naturalness of generated UATs. A joint objective function is designed to achieve this goal.
- We leverage the universal sentence encoder (USE) (Cer et al., 2018) as an additional evaluation metric than perplexity to better measure the naturalness of UATs generated by *LinkPrompt*.
- We conduct the the transferability study of *LinkPrompt* on BERT (Devlin et al., 2018) as well as an open-sourced large language model Llama2 (Touvron et al., 2023).
- Extensive experiments validate that *LinkPrompt* outperforms the baseline method, achieving a higher ASR while increasing the naturalness as well. Experimental results also demonstrate its strong transferability and stability against the adaptive defense method.

2 Related Work

Prompt-based fine-tuning. Prompt-based fine-tuning aims to fine-tune the PLMs with task-specific prompts to bridge the gap between PLMs and downstream tasks. Recent studies have explored a wide range of prompt-based fine-tuning techniques (Shin et al., 2020; Zhang et al., 2021; Tam et al., 2021; Deng et al., 2022), and the development of other prompt-based approaches like in-context learning (Xie et al., 2021; Dong et al., 2022) and instruction learning (Wei et al., 2021; Wang et al., 2022; Lou et al., 2023) is also progressing rapidly. In such a paradigm, the choice of prompt becomes crucial. Scao and Rush (2021) demonstrate that a prompt can be as effective as

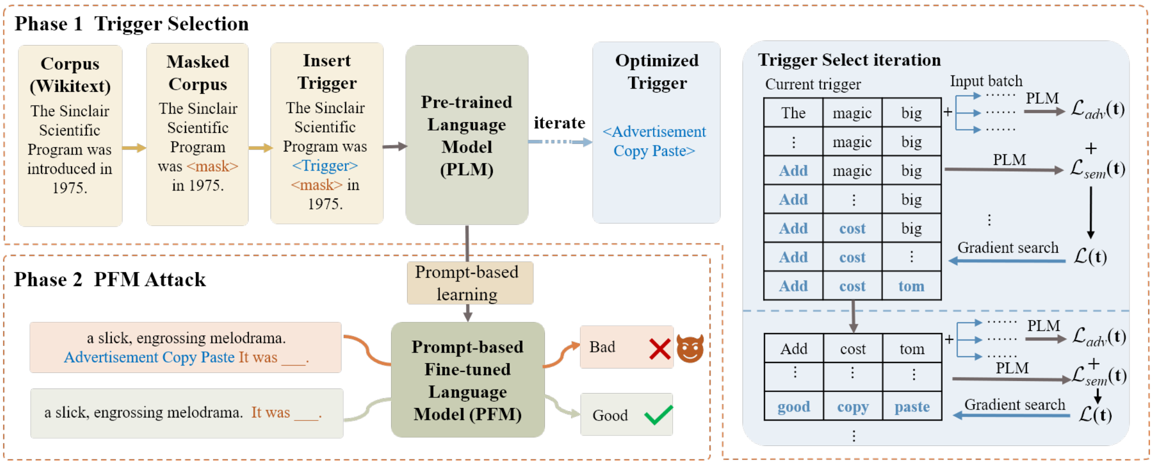


Figure 2: Workflow of *LinkPrompt*.

100 regular data points, indicating a significant improvement in sample efficiency.

Adversarial attack on the prompt-based model in classification tasks. Similar to the adversarial attack on simple text classification models, prompt-based learning frameworks also suffer from potential adversarial threats. Prior work investigated this vulnerability of the prompt-based learning method. Nookala et al. (2023) compared PFMs against fully fine-tuned models using the AdvGLUE (Wang et al., 2021) benchmark, and demonstrated the PFMs' lack of robustness to adversarial attacks. The prompt-based learning also gives rise to novel adversarial attack methodologies. One direction is to utilize the prompt engineering to generate adversarial examples that are semantically natural leveraging the sensitivity of language models to prompts (Yu et al., 2022; Yang et al., 2022). Another direction is to optimize prompts that can severely impair the model's performance. Tan et al. (2023) designed heuristic perturbation rules against manual prompts.

Universal adversarial attacks. Universal adversarial attacks refer to perturbations that are input-agnostic and were implemented by Wallace et al. (2019) firstly in the text domain. Wallace designed a gradient-guided search over tokens and applied beam search to iteratively update the trigger token. PromptAttack (Shi et al., 2022) utilized the gradient-based searching algorithm to automatically optimize prompts that can alter the PLM's prediction. Besides, Xu et al. (2022) proposed AToP, and demonstrated that PFMs are also vulnerable to triggers found in PLMs. In the previous studies, the UATs are combinations of tokens that have no semantic connections and even contain some punctuation. Although several attempts have been made to improve the naturalness of UATs (Atanasova et al., 2020; Song et al., 2020), they neither lack the attack utility (reduced the attack success rate) nor were studied in the prompt-based learning paradigm.

3 Method

In this section, we first give an overview of the prompt-based learning and *LinkPrompt* attack process, as well as our threat model. Then we introduce the optimization process of *LinkPrompt* universal attack in detail, including the design of objective functions and the optimization process.

3.1 Overview

The prompt-based learning paradigm involves two steps. First, a model is pre-trained on a diverse set of tasks, forming a Pretrained-Language Model (PLM) denoted as $\mathcal{F}$. Second, instead of fine-tuning the PLM to specific downstream tasks via traditional objective engineering, a textual prompt template $\mathbf{p}$ is utilized to transform the input $\mathbf{x}$ into a modified input $\mathbf{x}'$. Typically, prompts are integrated with input text through prefixes or suffixes, containing `[mask]` tokens. In classification tasks, the model $\mathcal{F}$ will be fine-tuned to a Prompt-based Fine-tuned Model (PFM) $\mathcal{F}'$ by training it to predict the correct label associated with the `[mask]` token in the prompt template.

The similarity between PLMs and PFMs raises concerns about the potential adversarial threats of prompt-based learning. The adversarial trigger optimized to target the PLMs can also transfer to the PFMs. In this work, *LinkPrompt* is proposed to generate natural and universal adversarial triggers on PFMs, which can not only alter the model prediction but also maintain the inherent high semantic meaning. The process of achieving this goal can be described as two steps: trigger selection and PFM attack.

As demonstrated in Figure 2. In the trigger selection phase, we first generated a corpus dataset $\mathcal{D} = \{(\mathbf{x}', y)\}$ by randomly substituting a word y with [mask] token in the original sentence $\mathbf{x}$ (first two blocks in Phase 1 of Figure 2). Then we inject trigger tokens before the [mask] token and iteratively optimize tokens by minimizing the probability of the [mask] token being correctly predicted by the PLM (the attack goal), and simultaneously maximizing the semantic meaning among the trigger tokens (the semantic goal). In the PFM attack phase, the optimized trigger tokens <Trigger> are injected between the input $\mathbf{x}$ and the prompt template $\mathbf{p}$ to mislead the PFM.

3.2 Threat Model

We assume that attackers do not have access to the downstream tasks, including the datasets and the PFM $\mathcal{F}'$, while having full access to the PLM $\mathcal{F}$, including the model parameters and gradients. The attacker can optimize adversarial trigger tokens over the PLM $\mathcal{F}$ while carrying out attacks on the PFMs $\mathcal{F}'$ with optimized adversarial triggers.

The attacker's goal is to find input-agnostic and semantically related adversarial trigger tokens <Trigger> with a fixed length L, denoted as $\mathbf{t} = \{t_i\}_{i=1...L}$, on the PLM $\mathcal{F}$. When adding the adversarial trigger with any benign input, PFM $\mathcal{F}'$ will give wrong predictions.

3.3 Trigger Selection

In our work, we propose *LinkPrompt* to generate universal adversarial trigger $\mathbf{t} = \{t_i\}_{i=1...L}$, where L is a pre-fixed length of the trigger, such that the likelihood of correctly predicting the masked word y on $\mathcal{D}$ can be minimized and the semantic relevance among the trigger tokens can be maximized.

Attack objectives. To achieve the attack goal, the first objective $\mathcal{L}_{adv}$ is designed to minimize the probability of the [mask] token being correctly predicted by the PLM. In other words, we want to maximize the cross-entropy loss of the predicted token and the masked token y, which equals to minimize the following loss:

$$\mathcal{L}_{adv}(\mathbf{t}) = -\frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}',y)\in\mathcal{D}} \mathcal{L}_{ce}(\mathcal{F}(\mathbf{x}' \oplus \mathbf{t}), y) \quad (1)$$

where $\mathcal{L}_{ce}(\cdot)$ represents the cross-entropy loss and $\mathcal{F}(\cdot)$ represents the prediction probability generated by PLM.

Algorithm 1: Beam Serach for *LinkPrompt*

Input: Initial trigger $\mathbf{t}$, Corpora $\mathcal{D}$, trigger length L, search steps N, batch size M, weight α, vocabulary list $\mathcal{V}$, candidate size C, beam size B.

trigger_list: $\mathcal{T} \leftarrow \mathbf{t}$;
while step < N **do**
  $[\mathbf{x}'^{(i)}, y^{(i)}]_{i=1...M} \sim \mathcal{D}]$;
  for $k \in 1, \ldots, L$ **do**
    for $\mathbf{t} \in \mathcal{T}$ **do**
      $\mathcal{L}_{adv} \leftarrow -\frac{1}{M}\sum_{i=1}^{M} \mathcal{L}_{ce}(\mathcal{F}(\mathbf{x}'^{(i)} \oplus \mathbf{t}), y^{(i)})$;
      $\mathcal{L}_{sem} \leftarrow -\frac{1}{L-1}\sum_{j=2}^{L} \mathcal{F}(t_j|t_{1:j-1})$;
      $\mathcal{L} \leftarrow \mathcal{L}_{adv} + \alpha\mathcal{L}_{sem}$;
      for $w \in \mathcal{V}$ **do**
        $\omega \leftarrow -\left\langle \nabla_{\mathbf{e}_{t_k}} \mathcal{L}, \mathbf{e}_w - \mathbf{e}_{t_k} \right\rangle$
        // $\mathbf{e}_{(\cdot)}$ is the embedding
      candidate_list: $\mathcal{C} \leftarrow \emptyset$;
      $\mathcal{C} \leftarrow w$ with top-C (ω);
      for $c \in \mathcal{C}$ **do**
        $\mathbf{t}' \leftarrow \mathbf{t}_{1:k-1} \oplus c \oplus \mathbf{t}_{k:L}$;
        $\mathcal{L}_{adv} \leftarrow -\frac{1}{M}\sum_{i=1}^{M} \mathcal{L}_{ce}(\mathcal{F}(\mathbf{x}'^{(i)} \oplus \mathbf{t}'), y^{(i)})$;
        $\mathcal{L}_{sem} \leftarrow -\frac{1}{L-1}\sum_{j=2}^{L} \mathcal{F}(t'_j|t'_{1:j-1})$;
        $\mathcal{L} \leftarrow \mathcal{L}_{adv} + \alpha\mathcal{L}_{sem}$;
    $\mathcal{T} \leftarrow \mathbf{t}'$ with top-B $(\mathcal{L})$

Output: Optimized trigger list $\mathcal{T}$

Semantic objectives. To achieve the semantic goal which is to maintain the semantic meaning among the adversarial trigger tokens, the second objective is to maximize the probability of the current candidate token given the previous tokens. Leveraging the predictive ability of the PLMs, such prediction probability can reflect the semantic relevance between the candidate token and the preceding context. To maximize the inherent semantic naturalness of a specific trigger $\mathbf{t}$ of length L, we use the probability of the current candidate token t_i being predicted based on the previous tokens to represent the semantic naturalness between the current token with the previous tokens. Therefore, the loss can be defined as the summation of each token's prediction probability given the previous token in the trigger:

$$\mathcal{L}_{sem}(\mathbf{t}) = -\frac{1}{L-1}\sum_{i=2}^{L} \mathcal{F}(t_i|\mathbf{t_{1:i-1}}) \quad (2)$$

Note that we want to maximize the prediction probability which equals to minimize the negative of the above loss. In addition, the generated trigger is universal to all inputs, making it unrealistic to maintain the semantic meaning between the trig-

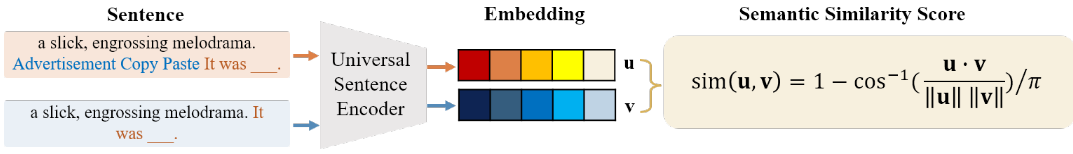


Figure 3: The process of calculating the semantic similarity.

ger and the input. Therefore, the summation of each token's prediction probability starts from the second token as the first trigger token's semantic naturalness is unable to be calculated.

Optimization process. The total loss objective is the weighted combination of the above two parts:

$$\mathcal{L}(\mathbf{t}) = \mathcal{L}_{adv}(\mathbf{t}) + \alpha\mathcal{L}_{sem}(\mathbf{t}) \quad (3)$$

The optimization over the adversarial triggers starts with a random initialization of $\mathbf{t}$. Then in each round, the tokens are updated sequentially from left to right by minimizing the above loss function. We use the first-order Taylor approximation around the initial trigger embeddings and take the beam search strategy (Wallace et al., 2019):

$$t_i \leftarrow \underset{t'_i \in \mathcal{V}}{arg\,min}[(\mathbf{e}_{t'_i} - \mathbf{e}_{t_i})]^T \nabla_{\mathbf{e}_{t_i}} \mathcal{L}(\mathbf{t}) \quad (4)$$

where $\mathcal{V}$ is the model vocabulary list and $\mathbf{e}_{t_i}$ represents the word embedding of t_i. The pseudo-code for the search algorithm is shown in Algorithm 1.

4 Experiment

In this section, we first introduce the configuration of our experiments including the victim model, datasets, prompt templates, baseline, and evaluation metrics. Then we evaluate the effectiveness and naturalness of UATs generated by *LinkPrompt*. Followed by that we demonstrate the transferability of *LinkPrompt* on Bert and Llama2. At the end, we propose an adaptive defense and show the stability of *LinkPrompt*.

4.1 Configurations

PLM and datasets. The victim PLM is RoBERTa-large (Liu et al., 2019), and we fine-tune the RoBERTa-large on six downstream classification tasks to get the PFMs, which are two sentiment analysis tasks on SST2 (Wang et al., 2018) and IMDB (Maas et al., 2011), two misinformation detection tasks on Fake News (FN, Yang et al., 2017) and Fake Review (FR, Salminen et al., 2022), one topic classification task on AG (Gulli, 2005) and one hate-speech detection task on HATE (Kurita et al., 2020). These classification datasets are also used to demonstrate the effectiveness of *LinkPrompt*. We fine-tune the RoBERTa model in the few-shot setting with 64 shots for two misinformation detection tasks and 16 shots for the rest tasks. The corpus commonly used to optimize UATs is generated from the Wikitext datasets.

Prompt templates and verbalizers. We use two types of prompt templates: Null template (Logan IV et al., 2021) that just append [mask] token to the text, and manual template that is specially designed for each task. Verbalizer, a tool to map a generated word to a corresponding class (e.g. word "good" to positive sentiment class), is manually designed for each task. Examples of prompt templates and verbalizers are shown in Table 1.

Dataset	Type	Prompt	Verbalizer
AG	Null	{sen} <T> <[mask]>	politics/business/
	Manual	{sen} <T> <[mask] news>	sports/technology
SST2	Null	{sen} <T> <[mask]>	bad/good
	Manual	{sen} <T> <It was [mask].>	
IMDB	Null	{sen} <T> <[mask]>	bad/good
	Manual	{sen} <T> <It was [mask].>	
HATE	Null	{sen} <T> <[mask]>	harmless/hate
	Manual	{sen} <T> <[mask] speech>	
FN	Null	{sen} <T> <[mask]>	real/fake
	Manual	{sen} <T> <It was [mask].>	
FR	Null	{sen} <T> <[mask]>	real/fake
	Manual	{sen} <T> <[mask] review>	

Table 1: Prompts and verbalizers used for fine-tuning PFMs. {sen}: input sentence, <T>: trigger, <[mask]...>: prompt template.

Baseline and evaluation metrics. We compare *LinkPrompt* with AToP, a state-of-the-art universal adversarial attack on PFM. The objective of AToP is the first loss term of Equation 3, which is equivalent to the situation that α is equal to 0.

We involve three evaluation matrices to demonstrate the performance of *LinkPrompt* from different aspects. First, accuracy (**ACC**) represents the models' performances on clear dataset $\mathcal{D}$, which can be stated as: $\mathrm{Acc}(\mathcal{F}) \stackrel{\text{def}}{=} \frac{1}{|\mathcal{D}|}\sum_{(\mathbf{x},y)\in\mathcal{D}} \mathbb{I}(\mathcal{F}(\mathbf{x} \oplus \mathbf{p}) = y)$. Accuracy indicates the baseline performance of PLM or PFM without any attacks. Second, Attack success rate (**ASR**) is a standard evaluation metric that represents the portion of correctly predicted examples whose classification can be flipped after trigger injection: $\mathrm{ASR}(\mathbf{t}) \stackrel{\text{def}}{=} \frac{1}{|\mathcal{D}'|}\sum_{(\mathbf{x},y)\in\mathcal{D}'} \mathbb{I}(\mathcal{F}(\mathbf{x} \oplus \mathbf{t} \oplus \mathbf{p}) \neq y)$. ASR gives an insight into the effectiveness of *LinkPrompt*.

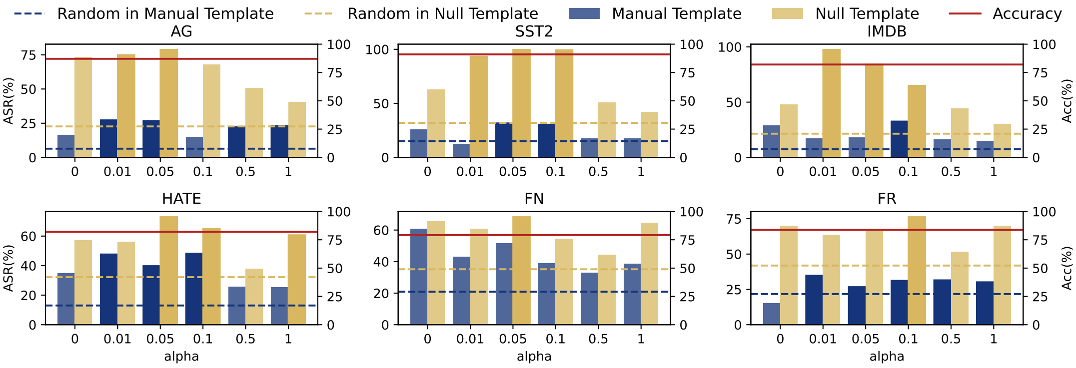


Figure 4: ASR results of 5-token triggers regarding different α on six datasets. The solid-color (deeper) bars mean ASR results better than the baseline (α=0). The red lines show the average accuracy of PFMs on clean datasets.

Last, the Semantic Similarity Score (**SSS**) represents the semantic similarity between the original and modified sentences. The assumption is that the more similar the adversarially perturbed sentence is to the original sentence, the more naturalness the UAT maintains, and the less it is suspicious to the adversarial detection. To measure SSS, We use the Universal Sentence Encoder (USE), a transformer-based encoder architecture to obtain and compare the embedding distance as shown in Figure 3. The similarity score can be calculated as $\text{sim}(\mathbf{u}, \mathbf{v}) = 1 - \arccos\left(\frac{\mathbf{u}\cdot\mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|}\right)/\pi$, where $\mathbf{u}$ and $\mathbf{v}$ present the embedding of perturbed sentence and original sentence respectively. Higher SSS indicates higher semantic similarity.

4.2 UATs Effectiveness Evaluation

We first demonstrate the overall ASR that *LinkPrompt* can achieve, and compare the ASR with the baseline method. Figure 4 shows the ASR on six datasets with different α with fixed trigger lengths equal to 5 (relegate results of other lengths to Appendix B.1). The red line represents the accuracy of clean data, which demonstrates the classification ability of the victim model, while the dotted lines represent the baseline with random token combinations. The yellow bars and blue bars represent the null template and manual template respectively. Bars with α equal to 0 in the AToP results and deeper color in other bars indicate a higher ASR than AToP.

From Figure 4, we can note that, first, on all datasets, *LinkPrompt* can achieve the highest ASR higher than 70% with certain α, even close to 100% on AG, SST2, and IMDB datasets, indicating the effectiveness of *LinkPrompt*. Second, for each dataset, there exists a selection of α that surpasses the baseline AToP (α equals 0). In addition, ASRs differ greatly between the manual and null templates in the first four datasets, while not much on the FN and FR. This may be explained by that the latter two tasks are more challenging and the manual template with a simple design still lacks robustness when facing the adversarial trigger attack.

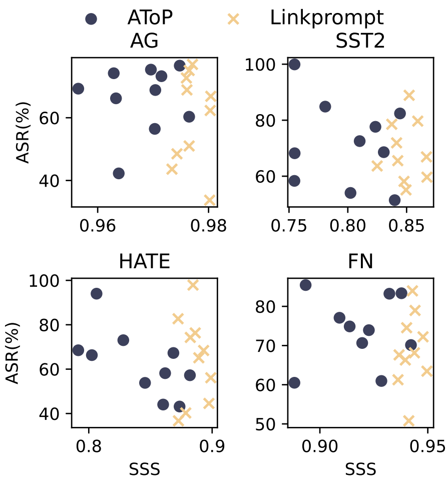


Figure 5: ASR vs. SSS. Trigger length = 5. Each dot represents an independent run.

4.3 UATs Naturalness Evaluation

Semantic Similarity Score. The effectiveness and naturalness of generated UATs are controlled by the weight α to balance the two loss terms. It is obvious that a greater α will push the optimization process to generate more natural UATs while suffering the trade-off on the ASR, and vice versa. Therefore, we plot the trade-off between the attack effectiveness and UAT naturalness with ASR and semantic similarity score (SSS) in Figure 5. We can note that UATs generated by *LinkPrompt* are gathered on the right-upper part of each plot, which indicates that *LinkPrompt* can achieve comparable ASRs while having higher SSS.

Triggers Visualization. We further visualize the UATs generated by *LinkPrompt* to demonstrate the naturalness. Table 2 captures the triggers found

Length	*LinkPrompt*	AToP
3	Loading Results View	organisers Crimes Against
	Ubisoft Bio Shock	\" The Last
	Advertisement Copy Paste	disorder.[edit
5	Related Articles Sega Console Controller	Yourselves Share Skip Disable JavaScript
	References Abstract PowerPoint Tables View	Davis -[{ Contentibility
	Armageddon NASA Goddard unar Exploration	[...] announ SHIP Email Address
7	goats VIDEOS Related Crash Video Leaves Teen	âĢ¦));News Videos Skip Javascript x
	Architectures Ratings VIEW Machine Analysis Using Deep	Kills Jenner Photos Drag ÃĹ View Coll
	Drink psychologists Researchers Say Mandatory Testing Is	. Shoppers reprene Issue Ratings Latest Corporate

Table 2: Triggers found in *LinkPrompt* and AToP of different lengths.

by both *LinkPrompt* and the baseline AToP under different trigger lengths. There are almost no meaningless symbols in *LinkPrompt* and the higher semantic relevance between the tokens can be observed.

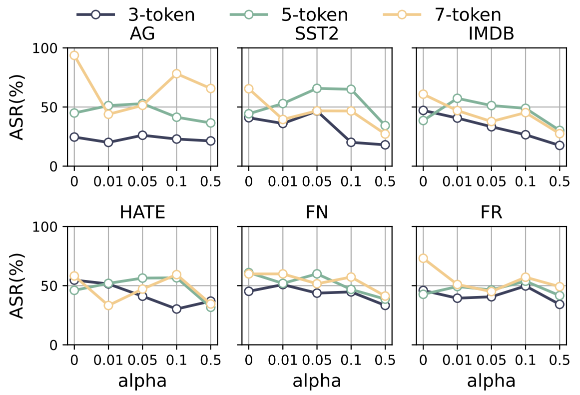


Figure 6: Ablation study on trigger length.

4.4 Ablation Study on Trigger Length

We conduct an ablation study on the trigger length to see how the ASRs change along with the value of α under different trigger lengths. In AToP (α equals 0), longer triggers can achieve higher ASR in almost every downstream task. However, the advantage of a longer trigger diminishes in *LinkPrompt*, 5-token *LinkPrompt* can achieve comparable ASR with original 7-token triggers. This phenomenon indicates that we may reduce the length of triggers by increasing semantic relevance between tokens.

4.5 Transferability

The UATs we evaluated in the previous sections are generated on RoBERTa-large. The transferability is crucial to adversarial perturbations which indicates the generalization ability of the generated UATs. Therefore, in this section, we want to evaluate whether the triggers we found on RoBERTa-large can lead to misclassification to other PFMs regardless of their structure.

Transfer to BERT. We first evaluate the transferability to BERT-large, which has a similar model architecture, pre-training data, and training methods to RoBERTa-large. Attack results on PFMs backboned with BERT-large using triggers found on RoBERTa-large in Table 3 show that *LinkPrompt* has strong transferability compared with baseline AToP on most of the datasets, especially with longer triggers (5 or 7).

Len	Matrices	α	AG	SST2	IMDB	HATE	FN	FR
3	ACC	-	86.10	87.15	73.02	77.42	75.40	80.84
	ASR	0	49.72	**64.26**	**63.88**	**65.92**	**65.18**	**55.68**
		0.01	35.54	44.44	50.56	37.00	48.25	47.77
		0.05	**56.82**	36.62	42.24	33.65	48.52	45.62
		0.1	32.38	32.58	43.85	29.70	44.80	48.23
		0.5	37.43	32.90	43.84	36.16	39.37	40.70
		1	34.25	35.36	38.37	39.37	44.66	41.54
5	ACC	-	86.04	86.54	72.44	77.14	74.26	80.76
	ASR	0	49.14	36.23	47.53	39.49	46.65	50.52
		0.01	38.91	37.33	45.82	**51.58**	55.73	**61.05**
		0.05	43.61	42.57	**53.26**	41.60	**59.38**	57.19
		0.1	**54.58**	42.44	53.04	39.31	47.29	48.91
		0.5	46.87	43.27	43.24	46.06	59.63	56.58
		1	41.80	**60.10**	44.63	40.92	53.69	59.09
7	ACC	-	84.22	83.18	72.43	79.65	73.76	80.44
	ASR	0	47.13	42.68	44.38	46.24	53.94	60.17
		0.001	**68.85**	59.07	50.18	52.48	57.50	**62.35**
		0.005	36.51	**62.44**	**63.23**	**59.00**	57.92	59.10
		0.01	48.87	41.81	51.34	44.05	**62.61**	62.29
		0.05	67.36	43.46	58.28	38.69	51.18	57.77
		0.1	67.13	44.52	52.19	43.28	53.21	61.04

Table 3: Transferability of *LinkPrompt* to Bert-large

Transfer to Llama2. We further analyze the transferability of *LinkPrompt* to Llama2, an open-sourced large language model. Unlike BERT and RoBERTa, Llama2 is a generative language model. To adapt it for classification tasks, we made special prompts for the training and inference stage. For example, on the SST2 dataset, we use "Predict the "`[mask]`" with "bad" or "good" to make the whole sentence semantically natural:" along with two examples as prompt in the training stage. All the prompts can be found in Appendix C. To get the PFM with different downstream tasks, we fine-tune Llama2 using the LoRA method (Hu et al., 2021) with lora rank = 8 and adapting key matrices and value matrices simultaneously. For evaluation, we randomly select UATs generated by *LinkPrompt* under each setting to demonstrate the transferability on Llama2. In this setting, a classification task is considered successful if the target label appears in the first 5 tokens predicted by the model. The ASRs to Llama2 when the trigger length is 5 are shown in Figure 7 (relegate results of other lengths to Appendix B.2). The strong transferability of *LinkPrompt* can be proved by the significantly better performance than the random baseline (dotted line). In addition, the difference between the manual template and the null template is much smaller compared to the results of BERT and RoBERTa.

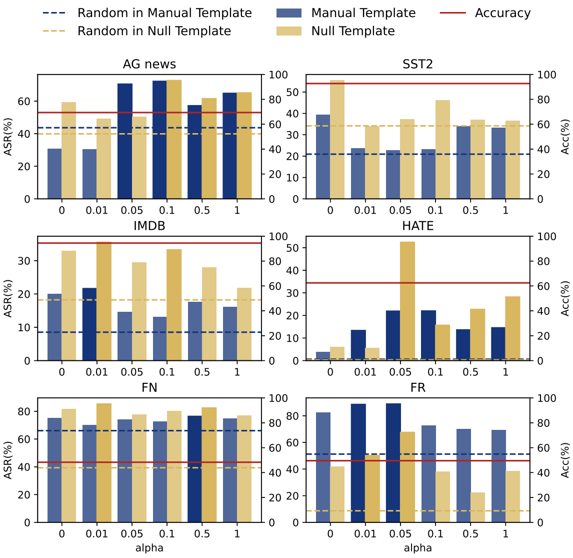


Figure 7: Transferability of *LinkPrompt* to Llama2.

4.6 Adaptive Defense

We further propose a perplexity filtering as an adaptive defense against *LinkPrompt*. Although *LinkPrompt* can maintain the semantic naturalness within the UAT, it is still irrelevant to the input sentences for the universality. Therefore, we proposed a perplexity detection filter inspired by ONION (Qi et al., 2020) to test the stealthiness of UATs generated by *LinkPrompt*.

We assume that outlier words are not closely related to the semantics of the entire sentence. Removing these words will make the meaning of the entire sentence clearer and reduce the perplexity. Given a sentence $\mathbf{x} = x_1, \ldots, x_n$, we use GPT2-large (Radford et al., 2019) to measure the perplexity $\mathcal{P}$. Then we enumerate remove words x_i from the sentence and record the perplexity of the sentence after removing the word (denote as $\mathcal{P}_\mathbf{i}$). If the impact of removing a word x_i on confusion exceeds a certain threshold, x_i is determined as an outlier word and will be removed.

We compare the stealthiness of *LinkPrompt* with the baseline method AToP on two datasets. We select UTAs generated by *LinkPrompt* that have comparable ASR with AToP to conduct a fair comparison. Table 4 shows the change of ASR after applying the filtering. First, we compare the drop of ASR under different trigger lengths (AToP and *LinkPrompt*$_{avg}$). As shown in Table 4, the drop of ASR (Δ columns) on *LinkPrompt* after the filtering is overall lower than AToP on both datasets, except the result on SST-2 with trigger length 7, which indicates that *LinkPrompt* is more resilient to the perplexity based adaptive defense.

Second, we compare the drop of ASR under different original ASRs (indicated as *LinkPrompt*$_{low}$ and *LinkPrompt*$_{high}$ in Table 4), as the original low and high ASRs have a different trend. Remember we design an objective function with a weighted sum of two loss terms from the attack and the naturalness perspective respectively. We can adjust the weight α to control the naturalness of generated UATs. Generally, a higher α can result in more natural but less successful UATs, and vice versa. In Table 4, ASRs of less effective triggers (*LinkPrompt*$_{low}$) even rise after the process of such a perplexity filter and the accuracy drops heavily on both tasks. This indicates the limitation of such an outlier detecting method towards *LinkPrompt*.

	SST-2 (ACC -9.79%)		HATE (ACC -15.93%)	
Trigger	ASR(%)	Δ (%)	ASR(%)	Δ (%)
AToP-3	27.75	-24.46	50.08	-29.05
LinkPrompt$_{avg}$-3	28.72	-12.45	39.58	-19.31
LinkPrompt$_{high}$-3	31.65	-20.73	35.15	-42.89
LinkPrompt$_{low}$-3	25.80	-4.17	44.01	+4.27
AToP-5	41.57	-21.07	45.57	-11.67
LinkPrompt$_{avg}$-5	54.77	-20.59	48.30	-7.10
LinkPrompt$_{high}$-5	46.07	-53.68	45.14	-27.85
LinkPrompt$_{low}$-5	63.47	+12.51	51.46	+13.65
AToP-7	39.23	-48.65	42.95	-31.74
LinkPrompt$_{avg}$-7	63.13	+0.52	50.04	-9.71
LinkPrompt$_{high}$-7	58.07	-15.94	50.83	-25.34
LinkPrompt$_{low}$-7	68.18	+16.98	49.25	+5.93

Table 4: Defense results of AToP and *LinkPrompt*.

5 Conclusion

We propose *LinkPrompt*, a universal adversarial attack algorithm on PFMs that can not only mislead the PFMs to give wrong predictions but also maintain naturalness. Compared with previous work, *LinkPrompt* can achieve a higher attack success rate while increasing the naturalness of triggers as well. We also evaluate the transferability of *LinkPrompt* to different model structures. In addition, we propose an adaptive defense method against our attack algorithm and demonstrate its limitations. In further study, we will explore new methods to generate triggers that are more stealthy with the assistance of large language models. It is also worthwhile to transfer such a method to other tasks or larger models.

Ethical Consideration

In this paper, we act as an attacker and propose an algorithm to generate UATs that are both effective and natural, which also have strong transferability and stability. It is possible that the UATs or our method are being maliciously used in terms of attacking existing language models. However, we consider research on such attacks to be significant to improve the robustness of state-of-the-art large language models and we intend to release both the algorithm and the generated triggers so that better defense can be developed in the future. In addition, we can gain insights from our experimental findings, resulting in a better understanding of the prompt-based fine-tuning paradigm and the language models as well.

Limitations

We conclude the limitations of our work in three aspects: First, to maintain the universality and effectiveness of triggers, which means that they can be adapted to any PFMs and inputs while having a high ASR, the triggers generated by *LinkPrompt* are still not natural enough in human evaluation. This may be explained by that there exists an inherent trade-off either between the universality or the performance and the fluency of triggers, which has also been proved in previous works. To improve the triggers' naturalness in the human evaluation system, developing the adversarial attack algorithm combined with techniques such as Reinforcement Learning from Human Feedback (RLHF) can be a potential solution. Second, in this paper, we mainly focus on the classification tasks and choose the masked language model RoBERTa-large as our victim model due to its good performance in such tasks. However, PFMs specific to generation tasks such as translation and dialogue can also suffer from adversarial threats. It is worthwhile to expand *LinkPrompt* to other tasks and larger-scale language models. Third, the adaptive defense based on a unified perplexity filter does not work well on *LinkPrompt*, which can be evidenced by the increase in ASR of certain triggers and the significant decrease in accuracy. In further studies, we intend to propose a stronger defense against *LinkPrompt* with the assistance of large language models. Instead of just computing the perplexity of a sentence, we can train a language model to determine whether a sentence is semantically natural or not.

References

Pepa Atanasova, Dustin Wright, and Isabelle Augenstein. 2020. Generating label cohesive and well-formed adversarial claims. *arXiv preprint arXiv:2009.08205*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric P Xing, and Zhiting Hu. 2022. Rlprompt: Optimizing discrete text prompts with reinforcement learning. *arXiv preprint arXiv:2205.12548*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Antonio Gulli. 2005. Ag's corpus of news articles. *Dipartimento di Informatica, University of Pisa, Nov*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019. Is bert really robust? natural language attack on text classification and entailment. *arXiv preprint arXiv:1907.11932*, 2.

Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight poisoning attacks on pre-trained models. *arXiv preprint arXiv:2004.06660*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pretrain, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Robert L Logan IV, Ivana Balažević, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2021. Cutting down on prompts and parameters: Simple few-shot learning with language models. *arXiv preprint arXiv:2106.13353*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Renze Lou, Kai Zhang, and Wenpeng Yin. 2023. Is prompt all you need? no. a comprehensive and broader view of instruction learning. *arXiv preprint arXiv:2303.10475*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Venkata Prabhakara Sarath Nookala, Gaurav Verma, Subhabrata Mukherjee, and Srijan Kumar. 2023. Adversarial robustness of prompt-based few-shot learning for natural language understanding. *arXiv preprint arXiv:2306.11066*.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Fanchao Qi, Yangyi Chen, Mukai Li, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2020. Onion: A simple and effective defense against textual backdoor attacks. *arXiv preprint arXiv:2011.10369*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1085–1097, Florence, Italy. Association for Computational Linguistics.

Joni Salminen, Chandrashekhar Kandpal, Ahmed Mohamed Kamel, Soon-gyo Jung, and Bernard J Jansen. 2022. Creating and detecting fake reviews of online products. *Journal of Retailing and Consumer Services*, 64:102771.

Teven Le Scao and Alexander M Rush. 2021. How many data points is a prompt worth? *arXiv preprint arXiv:2103.08493*.

Timo Schick and Hinrich Schütze. 2020. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*.

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.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

Liwei Song, Xinwei Yu, Hsuan-Tung Peng, and Karthik Narasimhan. 2020. Universal adversarial attacks with natural triggers for text classification. *arXiv preprint arXiv:2005.00174*.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Derek Tam, Rakesh R Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. 2021. Improving and simplifying pattern exploiting training. *arXiv preprint arXiv:2103.11955*.

Zihao Tan, Qingliang Chen, Wenbin Zhu, and Yongjian Huang. 2023. Cover: A heuristic greedy adversarial attack on prompt-based learning in language models. *arXiv preprint arXiv:2306.05659*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing nlp. *arXiv preprint arXiv:1908.07125*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

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

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *arXiv preprint arXiv:2204.07705*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Genta Indra Winata, Andrea Madotto, Zhaojiang Lin, Rosanne Liu, Jason Yosinski, and Pascale Fung. 2021. Language models are few-shot multilingual learners. *arXiv preprint arXiv:2109.07684*.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2021. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*.

Lei Xu, Yangyi Chen, Ganqu Cui, Hongcheng Gao, and Zhiyuan Liu. 2022. Exploring the universal vulnerability of prompt-based learning paradigm. *arXiv preprint arXiv:2204.05239*.

Fan Yang, Arjun Mukherjee, and Eduard Dragut. 2017. Satirical news detection and analysis using attention mechanism and linguistic features. *arXiv preprint arXiv:1709.01189*.

Yuting Yang, Pei Huang, Juan Cao, Jintao Li, Yun Lin, Jin Song Dong, Feifei Ma, and Jian Zhang. 2022. A prompting-based approach for adversarial example generation and robustness enhancement. *arXiv preprint arXiv:2203.10714*.

Xiaoyan Yu, Qilei Yin, Zhixin Shi, and Yuru Ma. 2022. Improving the semantic consistency of textual adversarial attacks via prompt. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080.

Ningyu Zhang, Luoqiu Li, Xiang Chen, Shumin Deng, Zhen Bi, Chuanqi Tan, Fei Huang, and Huajun Chen. 2021. Differentiable prompt makes pre-trained language models better few-shot learners. *arXiv preprint arXiv:2108.13161*.

A Experimental Details

Model and datasets. We use RoBERTa-large as our victim model, which has 355 million parameters in total. For transferability, we use BERT-large-cased and Llama2-7B, which have 336 million parameters and 7 billion parameters respectively. Note that users have to visit the Meta website and require a custom commercial license to use Llama2.

For finding triggers, we use the wikitext-2-raw-v1 as the corpus and use 512 examples to find each trigger. Wikitext-2-raw-v1 is a collection of over 100 million tokens extracted from the set of verified Good and Featured articles on Wikipedia. The dataset is available under the Creative Commons Attribution-ShareAlike License. In the attack phase, we use six datasets to organize the experiment. AG has 120,000 examples in the training set and 7,600 examples in the test set; SST has 6,920 examples in the training set and 1,821 examples in the test set; IMDB has 24,988 examples in the training set and 24,985 examples in the test set; HATE has 77,369 examples in the training set and 8,597 examples in the test set; FN has 19,076 examples in the training set and 8,174 examples in the test set; FR has 28,302 examples in the training set and 12,130 examples in the test set. All the datasets and models are open-sourced, and our use of them is consistent with their intended use.

Parameters and attack details. For searching triggers, we set the beam search size to 5, and the batch size to 16. The search algorithm runs for 1 epoch. To get PFMs, we fine-tune the PLMs in a few-shot setting using AdamW optimizer (Loshchilov and Hutter, 2017) with learning rate=1e-5 and weight decay=1e-2, and tune the model for 10 epochs. In the attack experiment, each task runs for 5 rounds to get the average results. We perform all the attack experiments on a single NVIDIA A100 GPU. It takes around 30 minutes, 1 hour, and 2 hours to generate a trigger of length 3, 5, and 7 respectively.

B Additional Experimental Results

B.1 Attack results of *LinkPrompt* on RoBERTa-large

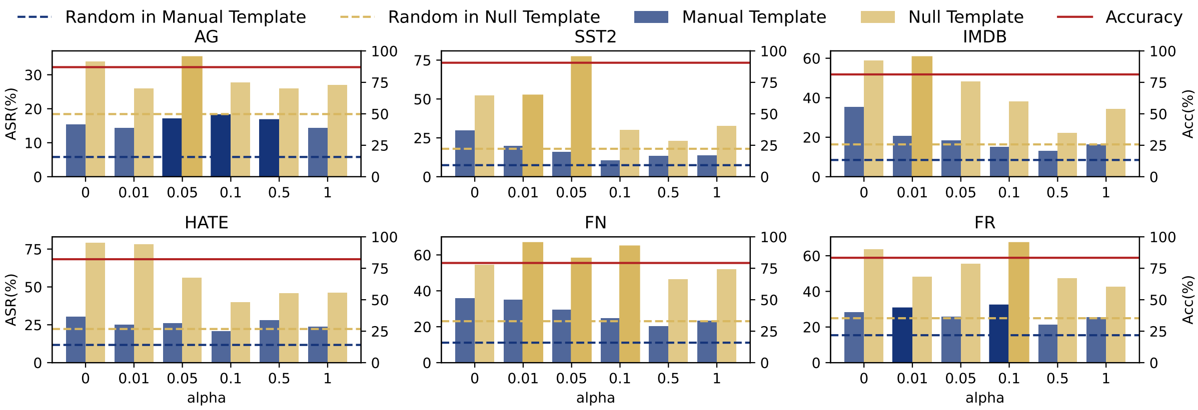
The ASR results of 3-token triggers and 7-token triggers are shown in Figure 8.

B.2 Attack results of *LinkPrompt* on Llama2

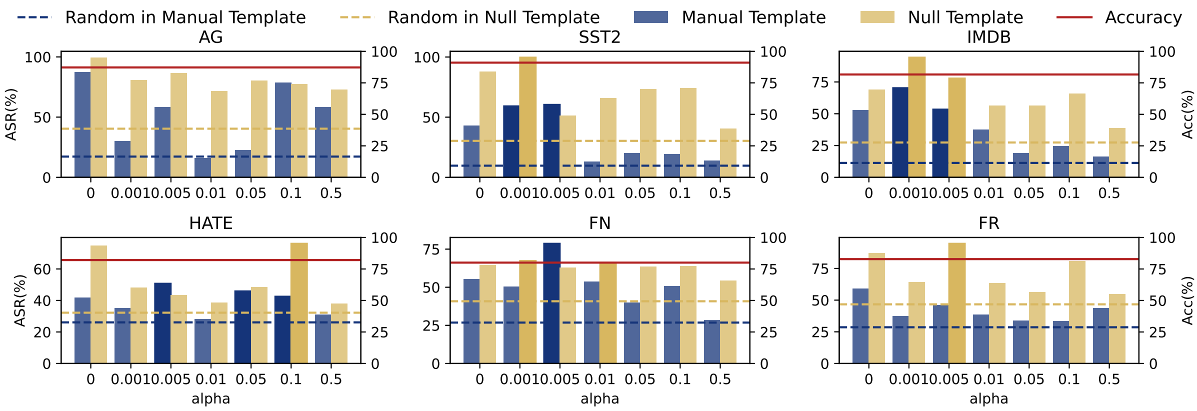
Transferability of 3-token triggers and 7-token triggers to Llama2 are shown in Figure 9.

C Prompt used for fine-tuning Llama2

Llama2, as a generative language model, predicts the next word based on the existing words. To adapt it for classification tasks, we made special prompts for the training and inference stage. The prompts we use to fine-tune Llama2 are shown in Table 5.

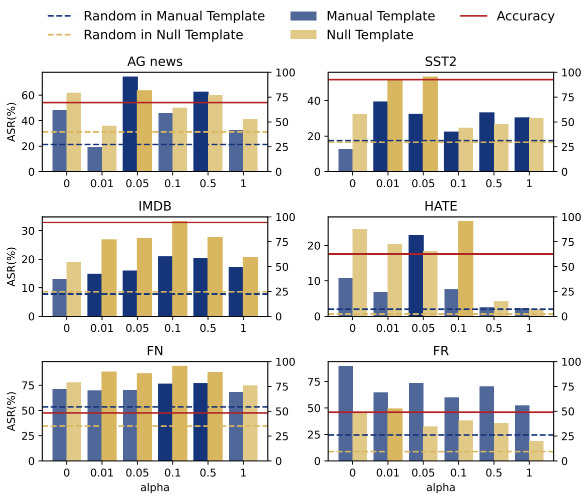


(a) ASR results of 3-token triggers.

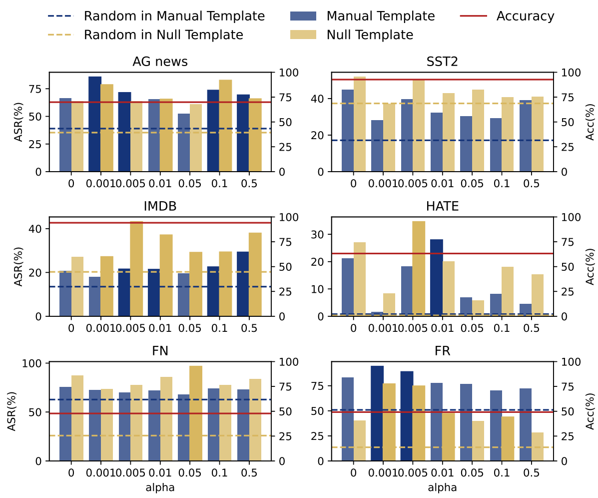


(b) ASR results of 7-token triggers.

Figure 8: ASR results of 3-token triggers and 7-token triggers regarding different α on six datasets.



(a) ASR results of 3-token triggers.



(b) ASR results of 7-token triggers.

Figure 9: Transferability of *LinkPrompt* to Llama2.

Dataset	Prompt
AG	**Predict the "[mask]" with "politics", "sports", "business" or "technology" to make the whole sentence semantically natural:** Najaf battle a crucial test for Allawi Clashes between US troops and Sadr militiamen escalated Thursday, as the US surrounded Najaf for possible siege. [mask] news => politics; Galaxy, Crew Play to 0-0 Tie (AP) AP - Kevin Hartman made seven saves for Los Angeles, and Jon Busch had two saves for Columbus as the Galaxy and Crew played to a 0-0 tie Saturday night. [mask] news => sports; Wall St. Bears Claw Back Into the Black (Reuters) Reuters - Short-sellers, Wall Street's dwindlingof ultra-cynics, are seeing green again. [mask] news => business; Oracle expands midmarket ambitions Company looks to juice its application server business with a version tuned for smaller organizations. [mask] news => technology
SST2	**Predict the "[mask]" with "bad" or "good" to make the whole sentence semantically natural:** a stirring , funny and finally transporting re-imagining of beauty and the beast and 1930s horror films It was [mask]. => good; apparently reassembled from the cutting-room floor of any given daytime soap .It was [mask]. => bad
IMDB	**Predict the "[mask]" with "bad" or "good" to make the whole sentence semantically natural:** Not only is it a disgustingly made low-budget bad-acted movie, but the plot itself is just STUPID!!! A mystic man that eats women? (And by the looks, not virgin ones) Ridiculous!!! If you´ve got nothing better to do (like sleeping) you should watch this. Yeah right. It was [mask]. => bad; Went to see this as Me and my Lady had little else to do on a sunday afternoon I like films that deal with sleazy,loser characters and this is full of em. After a slow start we get some good turns from the cast but it is the actual 'Bellini' that both makes and lets the film down. The 'Bellini' is one of the funniest scenes I have seen in a film for a long while but is too short and could have made this a masterpiece overall 71/2 out of 10 It was [mask]. => good
HATE	**Predict the "[mask]" with "harmless" or "hate" to make the whole sentence semantically natural:** Happy birthday to my brother @DavonteJones10 hope you have a good day fam love you &128170;&127998; see you later bro [mask] speech => harmless; RT @bateson87: Send Barkley off. He's a dirty bastard [mask] speech => hate
FN	**Predict the "[mask]" with "real" or "fake" to make the whole sentence semantically natural:** There was better news today for the ex-Toon hero Kevin Keegan when, after having resigned from the Newcastle United manager's job last week, he was offered a new job: Holding Joey Barton's Coat. Barton, still a player at United, albeit with a six-match ban, is a rabble-rouser, a trouble causer, a bit 'handy', pushy, a 'lad', good with his fists... temperamental, know what I mean? He regularly gets into fights, and is always in need of someone to ' hold his coat'. The last time Barton got into a 'scuffle', it ended in a jail sentence, and prior to that, he left Man City teammate Ousmane Dabo with his face 'caved in'. On both occasions, he was wearing a jacket, and believes he could have done so much more damage had he had a 'second' to hold his apparel for him. Ex-boss Keegan regarded himself as something of a father figure to Barton at Newcastle, defended him in front of the Newcastle board, and stood by him when he emerged from the nick recently. Now King Kev is to support the lad permanently as he follows him around, waiting for him to explode. The job is Full Time, 24 hours a day, 7 days a week, 52 weeks a year. You get the idea. Said Keegan:" I'm lookin forward to it. He's a nice lad, wears his heart on his sleeve, but there's nuthin wrong with that. He's got a fiery temperament, but he gives it his all, and you can't ask for anythin more than that." It was [mask]. => real
FR	**Predict the "[mask]" with "real" or "fake" to make the whole sentence semantically natural:** This is a great product. No more fears of loosing food to bears. No assaults yet but expect it to hold up nicely. [mask] review => real; Best FPV training transition to FPV and the FPV class is a lot of fun! The other two FPV classes are a bit more complex [mask] review => fake

Table 5: Prompts used for fine-tuning Llama2. We use the whole prompt for the training stage and the sentence in bold for the inference stage.