CR-UTP: Certified Robustness against Universal Text Perturbations on Large Language Models

Qian Lou[1+], Xin Liang[1*], Jiaqi Xue[1*], Yancheng Zhang[1], Rui Xie[2], Mengxin Zheng[1]
[1]Department of Computer Science, University of Central Florida
[2]Department of Statistics and Data Science, University of Central Florida

Abstract

It is imperative to ensure the stability of every prediction made by a language model; that is, a language's prediction should remain consistent despite minor input variations, like word substitutions. In this paper, we investigate the problem of certifying a language model's robustness against Universal Text Perturbations (UTPs), which have been widely used in universal adversarial attacks and backdoor attacks. Existing certified robustness based on random smoothing has shown considerable promise in certifying the input-specific text perturbations (ISTPs), operating under the assumption that any random alteration of a sample's clean or adversarial words would negate the impact of sample-wise perturbations. However, with UTPs, masking only the adversarial words can eliminate the attack. A naive method is to simply increase the masking ratio and the likelihood of masking attack tokens, but it leads to a significant reduction in both certified accuracy and the certified radius due to input corruption by extensive masking. To solve this challenge, we introduce a novel approach, the *superior prompt search* method, designed to identify a *superior prompt* that maintains higher certified accuracy under extensive masking. Additionally, we theoretically motivate why ensembles are a particularly suitable choice as base prompts for random smoothing. The method is denoted by *superior prompt ensembling* technique. We also empirically confirm this technique, obtaining state-of-the-art results in multiple settings. These methodologies, for the first time, enable high certified accuracy against both UTPs and ISTPs. The source code of CR-UTP is available at `https://github.com/UCF-ML-Research/CR-UTP`.

1 Introduction

Prompt-based Language Models (PLMs) (Thoppilan et al., 2022; Zeng et al., 2022; Achiam et al., 2023; Touvron et al., 2023b; Chiang et al., 2023) have achieved significant success across a wide range of real-world applications (Wu et al., 2020; Brown et al., 2020a; Wei et al., 2022; Chowdhery et al., 2023). However, despite their prominent performance, PLMs have been shown vulnerable to noises and perturbations on the input (Xu et al., 2022; Shayegani et al., 2023; Lou et al., 2022; Al Ghanim et al., 2023; Zheng et al., 2023b). Such vulnerability has notably restricted PLM's utility, especially in high-stake environments such as bank records analysis (Heaton et al., 2017), health care records analysis (Myszczynska et al., 2020). In these settings, the stability of every prediction is critical, i.e., PLM predictions should remain consistent despite minor input variations, such as word substitutions (Alzantot et al., 2018; Ren et al., 2019; Li et al., 2020a). This concern aligns with the study of certified robust PLMs (Zeng et al., 2023), which guarantees that all PLM predictions are accurate within the local vicinity of the input.

Input perturbations can be classified into Universal Text Perturbations (UTPs) and Input-Specific Text Perturbations (ISTPs). UTPs are characterized by their ability to be applied across different inputs, making them transferable, whereas ISTPs are tailored to specific inputs. In detail, attackers employing ISTP strategies, exemplified by TextFooler (Jin et al., 2020) and DeepWordBug (Gao et al., 2018), craft a unique adversarial sentence for each target input sentence. Conversely, attackers utilizing UTP methodologies, such as those found in TrojLLM (Xue et al., 2023) and UAT (Wallace et al., 2019), identify a single or a small number of adversarial tokens that can be inserted into any sentence to influence the model's prediction. This makes UTPs a more considerable threat to the robustness of PLMs since a specific set of adversarial tokens could lead to mispredictions across any input. Additionally, UTPs pose a greater challenge in mitigation compared to ISTPs. This challenge arises

[+]Corresponding author: Qian Lou, qian.lou@ucf.edu
[*]These authors contributed equally to this work.

Findings of the Association for Computational Linguistics ACL 2024, pages 9863–9875
August 11-16, 2024 ©2024 Association for Computational Linguistics

because ISTPs depend on weaker adversarial patterns that can be addressed by introducing modifications to the adversarial or clean tokens. However, UTPs are based on stronger adversarial patterns, which require exact identification and masking of the adversarial tokens for effective mitigation.

Random smoothing has been recognized as an effective defense offering certified robustness for models in computer vision (Horváth et al., 2021) and NLP (Zeng et al., 2023), yet its application has been limited to ISTPs. This method assumes that random alterations to a sample's words counteract perturbations. However, this approach falls short against UTPs, which require precise masking of adversarial tokens for mitigation, unlike ISTPs which can be mitigated by randomly masking any tokens. Defending against UTPs is challenging due to the unknown positions of adversarial tokens, requiring a high mask ratio that could degrade PLM accuracy. Thus, ensuring certified robustness against UTPs in PLMs remains an unresolved challenge.

Naively increasing the masking ratio can improve the chances of covering adversarial tokens in UTPs, potentially reducing the Attack Success Rate (ASR). However, this method often results in only minor ASR improvements due to the trade-off with certified accuracy. High masking ratios in random smoothing significantly diminish certified accuracy, leading to randomized model inferences as a large portion of input tokens are obscured, leaving insufficient data for accurate predictions.

In this paper, we introduce CR-UTP, a method to equip PLMs with certified robustness against UTPs, achieving both high certified accuracy and low ASR. Our contributions are as follows:

- We adapt the certified robustness method to PLMs and propose *Superior Prompt Search* for robust prompts with masked inputs.

- We introduce a prompt ensemble method to reduce the variance of masked inputs and increase the certified accuracy with theoretical analysis and empirical implementation.

- Through extensive experimentation, we show our CR-UTP effectively increases the certified accuracy by $\sim 15\%$ and decreases the ASR by $\sim 35\%$ compared to prior works.

2 Background and Motivation

Adversarial Attacks. Adversarial attacks in deep learning involve embedding a trigger into certain training samples, thereby creating poisoned datasets. When a deep learning model is trained on such tainted datasets, it behaves normally when presented with clean inputs but exhibits malicious behavior when encountering inputs containing the trigger. In the realm of visual images, triggers often manifest as tiny patches (Zheng et al., 2023c) or global perturbations (Zheng et al., 2023a; Li et al., 2023a). In the context of language data, triggers can be rare words or characters like "cf" (Kurita et al., 2020). This paper specifically focuses on text-based attacks.

Text Adversarial Attacks. Text adversarial attack methods generate adversarial sentences by perturbing original sentences to maximally increase the model's prediction error, while maintaining the fluency and naturalness of the adversarial examples. These attacks on prompt-based language models can be categorized into two groups: input-specific text perturbation attacks (ISTPs) and universal adversarial perturbation attacks (UTPs). In ISTP attacks, the attacker optimizes an adversarial sentence for each input, mainly by replacing, scrambling, and erasing characters (e.g., DeepWordBug (Gao et al., 2018) and HotFlip (Ebrahimi et al., 2018)) or words (e.g., TextFooler (Jin et al., 2020)). Conversely, UTP attacks optimize a universal trigger for a prompt-based language model, and the output of any input embedded with this trigger will be manipulated (e.g., TrojLLM (Xue et al., 2023) and UAT (Wallace et al., 2019)).

Certified Robustness in Language Models. Numerous defense methods, such as adversarial training (Yoo and Qi, 2021), model detection (Zheng et al., 2023d) and perturbation-controlled approaches, have been developed to counteract adversarial attacks (Wang et al., 2019; Zhou et al., 2021; Goyal et al., 2023). However, these traditional tools may become ineffective against novel, advanced attack strategies. To address this, certified robustness has been introduced, offering a guarantee against any attack as long as the number of perturbed words remains below a certain threshold. A model achieves certification if it can consistently produce the correct output when the number of perturbations does not exceed the certified radius. While models of smaller size can obtain robustness certification through deterministic methods (Li et al., 2020b; Ostrovsky et al., 2022; Weng et al., 2018; Kolter and Wong, 2017), the computational demands of language models preclude such approaches. Consequently, probabilistic methods,

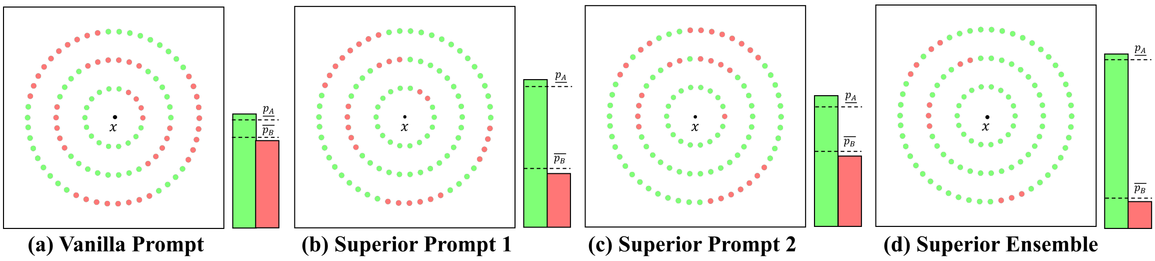


Figure 1: Illustration of the prediction distributions. A superior prompt exhibits greater robustness compared to a vanilla prompt, with ensembled prompts showing even higher robustness. Different colors represent various classes, and different radii indicate varying levels of perturbation. The bars demonstrate the output class probabilities for the smoothed PLMs given corresponding prompts. $\underline{p_A}$ represents the minimum probability of the majority class, and $\overline{p_B}$ indicates the maximum probability of the second-most likely class.

such as Random Smoothing (Cohen et al., 2019), have been introduced to certify the robustness of large language models.

Random Smoothing. Random Smoothing, a promising approach introduced by (Cohen et al., 2019; Weber et al., 2020), certifies the robustness of large neural networks. This method enhances a model's robustness by adding Gaussian noise to the original input (Salman et al., 2020; Li et al., 2023b). It was quickly adopted for large language models in the NLP field, exemplified by SAFER (Ye et al., 2020) and Randomized [MASK] (Zeng et al., 2023). To improve model performance with random smoothing, the computer vision field has explored re-training the original model to adapt to Gaussian noise (Jeong and Shin, 2020; Zhai et al., 2020; Salman et al., 2019). However, applying this re-training method in NLP to achieve a model tolerant to smoothing is prohibitively expensive. Instead, in the NLP domain, (Zhang et al., 2023) proposed a self de-noising method that allows the Pretrained Language Model (PLM) itself to recover the information lost due to masking.

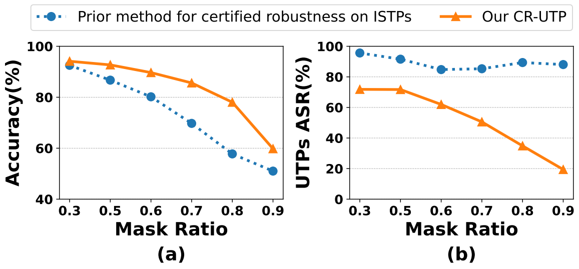
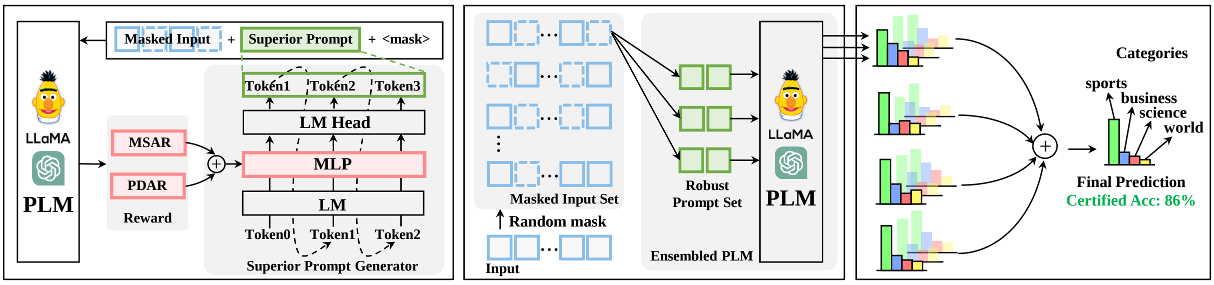


Figure 2: (a) Our CR-UTP shows higher certified robustness accuracy and (b) Our CR-UTP significantly reduces ASR.

Motivation. Figure 2 illustrates that while random smoothing has been effective for ISTPs as shown in previous works (Zeng et al., 2023; Zhang et al., 2023), it struggles with UTPs. With a low mask ratio, the ASR for UTPs is high (∼96%), and increasing the mask ratio only marginally reduces ASR but significantly lowers accuracy. For example, increasing the mask ratio from 0.3 to 0.8 only reduces ASR by ∼6% while accuracy drops by ∼35%. These findings highlight the inadequacy of ISTP methods for UTPs and lead us to develop new techniques that combine superior prompt search and ensembles, significantly improving robustness against UTPs with less impact on accuracy.

To achieve a low Attack Success Rate (ASR) against Universal Text Perturbations (UTPs), a high mask ratio, exceeding 0.5, is necessary. Yet, as Figure 1 (a) reveals, a vanilla prompt at this high mask ratio results in reduced accuracy due to the extensive masking of input tokens, which leaves limited information for precise classification. Figures 1 (b) and (c) illustrate that superior prompts can maintain higher accuracy under such conditions by incorporating random masking during the prompt design phase. This approach allows superior prompts to adjust to specific mask ratios, improving the lower bound ($\underline{p_A}$) on the majority class probability and reducing the upper bound ($\overline{p_B}$) on the runner-up class probability. However, these prompts still exhibit high variance across input samples. Ensembling techniques (Liu et al., 2021; Horváth et al., 2021) reduce this variance, thereby enhancing robustness. As shown in Figure 1 (d), by combining superior prompts, we can leverage their individual advantages for a more favorable accuracy-ASR balance. This insight leads us to further investigate superior prompt design and ensembling as methods to bolster PLMs' certified accuracy and robustness against UTPs, aiming to lower the ASR while preserving high accuracy.



(a) Superior Prompt Search **(b) Superior Prompt Ensemble** **(c) Certified Robust PLM**

Figure 3: Overview of CR-UTP. CR-UTP leverages superior prompt search and prompt ensembling techniques to enhance the certified robustness of PLMs.

3 CR-UTP Design

Overview. In Figure 3, we detail the workflow of the proposed CR-UTP method. (a) Superior Prompt Search: we start with a basic prompt and employ a reinforcement learning approach to find a superior prompt adept at handling inputs with masked words. A unique reward function is utilized, which incorporates random masking during the prompt search phase to enhance the prompt's resilience to word masking. (b) Superior Prompt Ensemble: for making predictions, CR-UTP generates various versions of the original input by applying random masking. Each prompt assesses these versions and internally agrees on the optimal prediction. (c) Certified Robust PLM: CR-UTP aggregates the individual outcomes from each prompt through a second voting process to get the final, most robust prediction.

In particular, we generalize the random masking operation to PLMs and analyze using random masking to defend UTPs in Section 3.1. Also, we introduce the Superior Prompt Search in Section 3.2 to search for prompts that can achieve a high certified accuracy even when a large proportion of the input tokens are masked. Finally, in Section 3.3, we proposed a Superior Prompt Ensemble to further improve the certified robustness.

3.1 Adapting Random Smoothing to PLMs

Using the random masking approach from Randomized [MASK] (Zeng et al., 2023), the random masking operation $\mathcal{M} : \mathcal{X} \times \mathcal{M}(h,k) \to \mathcal{X}_{mask}$ would take an input text $\boldsymbol{x} = x_1 x_2 ... x_h$ with h words and randomly replacing $(h-k)$ word with the [MASK] to get the masked version $\mathcal{M}(\boldsymbol{x})$. Following this, we define a smoothed classifier $g(\boldsymbol{x})$ built upon the base classifier f as follows:

$$g(\boldsymbol{x}) = \arg\max_{c \in \mathcal{C}} \left[\underset{\mathcal{H} \sim \mathcal{U}(h,k)}{\mathbb{P}} (f(\mathcal{M}(\boldsymbol{x} \mid \mathcal{H})) = c) \right] \quad (1)$$

Then it can be shown that $g(x)$ would return c when the certified condition is satisfied with probability at least $(1-\alpha)$ from Theorem 1 in (Zeng et al., 2023).

We define a prompt-based language model for classification tasks as $f : \mathcal{X} \to \mathcal{C}$, where $\mathcal{X}$ represents the domain of input texts and $\mathcal{C} = 1, 2, ..., n_c$ denotes the set of classification labels. The response of an input $\boldsymbol{x}$ to prompt p is $y = f(p, \boldsymbol{x})$. The smoothed classifier g over such PLM can be expressed as

$$g_p(\boldsymbol{x}) = \arg\max_{c \in \mathcal{C}} \left[\underset{\mathcal{H} \sim \mathcal{H}(h,k)}{\mathbb{P}} (f(p, \mathcal{M}(\boldsymbol{x} \mid \mathcal{H})) = c) \right] \quad (2)$$

and the same certification can be achieved under such prompt-based model.

Defending against universal attacks with random masking necessitates a high mask ratio, especially in the context of UTPs, where the presence of any UTP token in the masked input guarantees the attack's success. Therefore, to ensure the adversarial token is masked with a probability exceeding 50% for the smoothed function $g(\boldsymbol{x})$ to yield correct outcomes, the masking probability for each UTP token needs to be more than 0.5. Consequently, for a UTP of length r, the likelihood that all UTP tokens are masked should be $p^r > 0.5$, implying $p > \sqrt[r]{0.5}$. For instance, $p > 0.5$ for $r = 1$, and $p > 0.707$ for $r = 2$. Ensuring correct results for inputs masked without UTP requires the model to perform effectively at high mask ratios. To alleviate the effects of extensive masking, we introduce a technique that enhances model performance under random masking without necessitating retraining of the language model, thereby reducing computational overhead.

3.2 Superior Prompt Search

Certified accuracy is influenced by the model's performance on randomly masked sentences, but high mask ratios can decrease accuracy due to loss of critical information. Enhancing a prompt-based language model's certified accuracy involves improving its tolerance to information loss from random masking. However, in few-shot and black-box scenarios, fine-tuning the pre-trained model or using gradient-based optimization for continuous prompts is infeasible. To tackle this challenge without gradient optimization, we approach it as a reinforcement learning (RL) problem to discover a discrete, robust prompt—termed a superior prompt. A direct approach involves searching for this prompt using datasets with randomly masked sentences to acclimate the model to diverse masking scenarios. Nonetheless, at high mask ratios (e.g., 70%), the reduced information in few-shot datasets limits the effectiveness and generalizability of robustness enhancements. To overcome this, we suggest aligning the superior prompt's predictions on masked sentences with the vanilla prompt's on unmasked sentences, leveraging the existing knowledge of the vanilla prompt to offset the drawbacks of few-shot datasets and information loss from masking.

Our aim, as expressed in Equation 3, involves identifying an optimized prompt p_s that augments a basic vanilla prompt p_v by adding a sequence of T tokens from the vocabulary $\mathcal{V}$. This strategy is intended to boost the smoothed function $g_{p_s}(\boldsymbol{x_i})$'s accuracy on inputs $\boldsymbol{x_i}$. The dataset $\mathcal{D}$ is composed of pairs of input sentences x_i and their associated labels y_i.

$$\max_{p_s \in \mathcal{V}^T} \sum_{(x_i, y_i) \in \mathcal{D}} \mathbb{I}(g_{p_s}(\boldsymbol{x_i}) = y_i) \tag{3}$$

Masked Sentence Accuracy Reward. We introduce a two-fold reward function to guide the RL-based search for an optimal p_s. The first component, known as the Masked Sentence Accuracy Reward (MSAR), is designed to directly maximize the PLM's accuracy on masked sentences:

$$\mathcal{R}_{\mathsf{MSAR}} = \sum_{(\boldsymbol{x_i}, y_i) \in \mathcal{D}} \eta_1^{1-sign} \eta_2^{sign} Distance(\mathcal{M}(\boldsymbol{x_i}), y_i) \tag{4}$$

where the $Distance(\mathcal{M}(\boldsymbol{x_i}), y_i)$ denotes $l_{y_i}(p_s, \mathcal{M}(\boldsymbol{x_i})) - max_{y' \neq y} l_{y'}(p_s, \mathcal{M}(\boldsymbol{x_i}))$, the difference of the correct logit and the maximum of the incorrect logits. The distance value is positive for correct predictions and negative otherwise. We denote the distance sign as $sign = \mathbb{1}[Distance(\mathcal{M}(\boldsymbol{x_i}), y_i) > 0]$. For a correct prediction (i.e., $sign = 1$), we multiply the positive reward by a large number η_2 to indicate its desirability; otherwise, we multiply the negative rewards by another number η_1. This reward aims to maximize the PLM's accuracy on masked sentences.

Predictive Distribution Alignment Reward. To mitigate the challenge posed by information loss due to word masking, which is exacerbated in a few-shot setting, we propose an additional reward function, a.k.a, Predictive Distribution Alignment Reward (PDAR). It is designed to minimize the KL divergence between the predictive distributions of the vanilla prompt on unmasked sentences and the superior prompt on their masked equivalents:

$$\mathcal{R}_{\mathsf{PDAR}} = - \sum_{(\boldsymbol{x_i}, y_i) \in \mathcal{D}} KL(l(p_v, \boldsymbol{x_i}) \parallel l(p_s, \mathcal{M}(\boldsymbol{x_i}))) \tag{5}$$

This reward is designed to ensure that p_s retains alignment with p_v's predictive behavior, thereby leveraging the foundational knowledge encoded in p_v to inform predictions in the face of partial information. Such strategic alignment enables p_s to infer missing data from the masked inputs, drawing on the robust insights and patterns encapsulated by p_v. This method not only addresses the direct impact of masking on information availability but also enhances the model's capacity for generalization from limited examples.

Policy Model Update. As Figure 3 shows, the RL search process involves an agent that sequentially selects tokens $[s_1, ..., s_T]$ to construct the superior prompt p_s, aiming to maximize the combined reward $\mathcal{R} = \mathcal{R}_{\mathsf{MSAR}} + \mathcal{R}_{\mathsf{PDAR}}$. For each time step t, the agent, given the previous tokens $s_{<t}$, generates the next token s_t based on the policy generator $G_{\theta_s}(s_t | s_{<t})$. Completion of p_s triggers the computation of the task reward $\mathcal{R}$. To facilitate this, we employ a GPT-2 model as the backbone for our policy generator, enhanced with an insertable trainable Multilayer Perceptron (MLP) layer. The optimization focuses on the parameters of this MLP layer, tailoring the policy generator to effectively navigate the prompt construction space under the guidance of the designed reward function.

3.3 Superior Prompt Ensemble

Instead of relying on a single model, ensemble methods leverage the strengths and mitigate the weaknesses of various base classifiers. The core idea behind ensemble modelling is that a group of weak learners can come together to form a strong learner, thereby improving the model's ability to generalize well to unseen data. In the following section, we will demonstrate how random masking can increase the variance of the output probability and how model ensembling can mitigate the performance drop introduced by random masking.

Suppose the probability that the smoothed model with prompt p outputs the ground truth y is $P^y(\boldsymbol{x}) = \mathbb{P}_{\mathcal{H}\sim\mathcal{U}(h,k)}(f(p, \mathcal{M}(\boldsymbol{x} \mid \mathcal{H})) = y)$ in $[0, 1]$, then the final probability output of the smoothed model is determined by two random variables, $P^y = P^y_o + P^y_m$ (Horváth et al., 2021), where P^y_o is determined by the performance of the language model f with prompt p on the original input, and P^y_m corresponds to the performance of prompt p when the input $\boldsymbol{x}$ is randomly masked. Although P_o should be constant 0 or 1 without any perturbation on the input, we could assume $P^y_o = l^y(p, \boldsymbol{x})$, the logit of the correct label under the random masking operation. We empirically analyse the performance of the mask operation variance σ^2 with respect to the perturbation rate, as shown in figure 4, and conclude that the perturbation ratio significantly influences variance. As the perturbation ratio increases, the variance initially rises and then decreases. This pattern occurs because, as the mask ratio increases from 0 to 0.6, the masking operation introduces more noise to the input, increasing the variance of P^y. However, when the mask ratio increases from 0.6 to 1, the remaining information decreases, leading the model to randomly guess any label, i.e., $P^y \to 1/n_c$, and the variance $\sum^y \to 0$ as more words in the sentence are masked.

With a high mask ratio, the random masking operation can significantly increase variance and reduce accuracy relative to the clean classifier's output. Model ensembles effectively decrease the overall variance of voting outcomes, thereby improving the likelihood of accurate predictions as the number of ensembles increases (Horváth et al., 2021). However, the high computational cost makes training multiple language models unfeasible. Consequently, we introduce a technique that ensembles a set of prompts during inference to exploit the distinctive feature of PLMs, wherein the initial prompt markedly affects the model's final output. By employing the superior prompt search method, we can create a collection of prompts that withstand the random masking operation, with the ensemble of these prompts' outputs demonstrating enhanced performance on heavily masked inputs.

Formally, we construct an ensemble classifier $\bar{f}$ with a set of k different prompts $\boldsymbol{P} = \{p_i \mid i = 1, ..k\}$, via hard voting of all the outputs from different prompts p_i,

$$\bar{f}(\boldsymbol{P}, \boldsymbol{x}) = \arg\max_{c\in\mathcal{C}} \sum_{i=1}^{k} \mathbb{I}(f(p_i, \boldsymbol{x}) = c) \quad (6)$$

where $\mathbb{I}(f(p_i, \boldsymbol{x}) = c)$ is the indicator function that equals 1 when $f(p_i, \boldsymbol{x})$ output c and 0 otherwise. So the ensemble classifier would output the class that most of the prompts agree on.

Since the prompts ensemble operates as a single model, the certified robustness condition remains applicable to the assembled model $\bar{f}(\boldsymbol{P}, \boldsymbol{x})$. Therefore, we can establish a new smoothing function $\bar{g}(\boldsymbol{x})$ by applying the same random masking operation to $\boldsymbol{x}$. Building on our previous findings, we anticipate performance improvements through the prompts ensemble. Our analysis of how the number of ensembles impacts the final probability outcome, as depicted in Figure 5, demonstrates a substantial increase in the accuracy of the model ensemble with a concurrent reduction in variance as the number of ensembles increases.

4 Experimental Methodology

Datasets and Model. In our evaluation, we utilize the SST-2 dataset (Socher et al., 2013) and Yelp (Asghar, 2016) for binary classification tasks, AgNews dataset (Zhang et al., 2015) for a four-class classification task. We adopt a 16-shot setting, which represents a typical few-shot scenario. Our experiments are mainly conducted with the widely-used pre-trained language model RoBERTa-large (Liu et al., 2019), an advanced version of BERT (Kenton and Toutanova, 2019) with 24 layers of Transformer architecture. We also evaluate the performance on large language models such as Llama2-7b (Touvron et al., 2023a) and GPT-3.5 (Brown et al., 2020b).

Evaluation Metrics. We adopted three key metrics in evaluations same with (Zeng et al., 2023). Clean accuracy **(CACC)** refers to the classification accuracy on clean sentences. The attack success rate **(ASR)** quantifies the percentage of input instances perturbed by an attack that successfully causes the model to make incorrect predictions. The poisoned accuracy **(PACC)** indicates the accuracy of the prompt-based language model on poisoned samples crafted from an attack.

Table 1: Comparison of CR-UTP and Random Mask against attacks with a 70% mask ratio on SST-2 dataset.

Scheme	w/o defence			Randomized [MASK]			CR-UTP		
	CACC	ASR	PACC	CACC	ASR	PACC	CACC	ASR	PACC
DeepWordBug	92.69	93.04	6.96	81.13	45.18	54.82	85.61	21.25	78.75
TextFooler	92.69	91.87	8.13	81.60	42.88	57.12	85.28	37.39	62.61
UAT	92.48	96.85	52.97	80.75	79.92	60.18	85.53	50.63	75.92
TrojLLM	92.69	91.88	53.76	80.94	85.31	56.84	85.70	50.55	73.04

Evaluated Attacks. We evaluated our CR-UTP under two input-specific text perturbation **(ISTP)** adversarial attacks, TextFooler (Jin et al., 2020) and DeepWordBug (Gao et al., 2018), and two universal text perturbation **(UTP)** attack, UAT (Wallace et al., 2019) and TrojLLM (Xue et al., 2023). TextFooler adversarially perturbs the text inputs by the word-level substitutions, whereas DeepWordBug performs the character-level perturbations to each input by replacing, scrambling, and erasing a few characters of some words. The UAT attack generates universal adversarial triggers as sequences of tokens that are independent of input and, when appended to any dataset entry, prompt the model to generate a particular prediction. UTP attack TrojLLM uses reinforcement learning to search a universal trigger for a prompt-based language model, any text inputs with this trigger will lead to the model output target label.

Implementation Details. For the superior prompt generator configuration, we adhered to the parameters established in RL-Prompt (Deng et al., 2022). Specifically, we use distilGPT-2, a large model with 82 million parameters, as a policy model for all tasks. Additionally, we use a multilayer perceptron (MLP) with one hidden layer which has 2,048 hidden states, added to distilGPT-2's existing 768 hidden states. For the hyperparameters of reward functions in the Equation 4, we set balancing weights $\eta_1 = 180$ and $\eta_2 = 200$. During inference of CR-UTP, we use an ensemble number of 5 with the best 5 prompts derived from the superior prompt search. During the certification process, the prediction number is 500 and the certification number is 1000. When using randomized [MASK] to defend against adversarial attacks, the voting number is set to 100. All experiments are conducted on a single Nvidia Geforce RTX-3090 GPU. Searching time for superior prompts on SST-2 is 3.8 hours, the certification time for one sentence is ~ 8 seconds. Further details about training time and inference efficiency are provided in the appendix A.

5 Results

5.1 Comparison of CR-UTP with Random Mask.

In Table 1, we conducted experiment comparing the performance of CR-UTP with no defence input and Randomized [MASK] (Zeng et al., 2023) at a 70% masking ratio against two ISTP adversarial attacks, i.e., DeepWordBug (Gao et al., 2018), TextFooler (Jin et al., 2020), and two UTP adversarial attacks, i.e., UAT (Wallace et al., 2019) and TrojLLM (Xue et al., 2023). CR-UTP significantly reduces the ASR from 96.85% to 50.63% on UAT attack, and 93.04% to 21.25% on DeepWordBug attack, which suggests random masking operation with a high mask ratio could effectively reduce attack success rate on both ISTP and UTP adversarial attacks. Our CR-UTP exhibits superior performance over Randomized [MASK] across all metrics in the evaluated adversarial attacks. Furthermore, CR-UTP exhibits higher CACC and PACC than the original input and Randomized [MASK]. This improvement is attributed to its efficient prompt search method, which identifies robust prompts to random mask, and superior prompt ensemble technique, further reducing CACC variance. Moreover, CR-UTP achieves a substantial reduction in attack success rate (ASR), averaging a 21.4% greater decrease compared to Randomized [MASK], with a remarkable 34.76% ASR reduction in the TrojLLM attack. This enhancement stems from CR-UTP's ability to leverage the differential outputs of various prompts, enabling a robust ensemble prediction for improved defence outcomes against adversarial samples. Additionally, CR-UTP demonstrates a notable increase in poisoned accuracy (PACC), indicating its ability to maintain high accuracy even under attack scenarios.

5.2 Ablation Study

In this section, we explore the design space of CR-UTP and study the impact of various settings of CR-

Table 2: An ablation study of CR-UTP techniques. Our baseline is random mask with 70% ratio; $\mathcal{R}_{\text{MSAR}}$ denotes employing superior prompt search only using reward $\mathcal{R}_{\text{MSAR}}$; $\mathcal{R}_{\text{MSAR}} + \mathcal{R}_{\text{PDAR}}$ means using superior prompt search with rewards $\mathcal{R}_{\text{MSAR}}$ and $\mathcal{R}_{\text{PDAR}}$; CR-UTP incorporates all proposed techniques.

Dataset	SST-2			AgNews			Yelp		
	CACC	ASR	PACC	CACC	ASR	PACC	CACC	ASR	PACC
w/o defense	92.69	91.88	53.76	88.91	94.54	78.64	95.42	87.90	51.42
Our baseline	70.50	85.31	56.84	80.94	22.42	75.71	76.10	58.26	68.35
$\mathcal{R}_{\text{MSAR}}$	81.93	47.28	76.83	82.09	21.31	75.78	83.23	22.93	81.93
$\mathcal{R}_{\text{MSAR}} + \mathcal{R}_{\text{PDAR}}$	84.90	63.93	66.61	83.06	18.89	78.65	83.84	20.83	82.52
CR-UTP	85.70	50.55	73.04	84.27	18.82	78.73	85.62	20.05	82.85

UTP on its attacking effects using the RoBERTa-Large with SST-2 dataset.

CR-UTP Techniques Performance. In Table 2, we analyze the impact of different CR-UTP techniques on performance against TrojLLM on the SST-2, AgNews and Yelp-2 datasets. For SST-2, utilizing the adapted random mask method (our baseline) leads to a significant drop in CACC by over 22%, mainly due to the loss of information from masking 70% of the words. However, incorporating superior prompt search with reward $\mathcal{R}_{\text{MSAR}}$, improves CACC by 11% as the superior prompt proves more robust to random masking. Furthermore, combining rewards $\mathcal{R}_{\text{MSAR}} + \mathcal{R}_{\text{PDAR}}$, further increases CACC to 84.90% by enhancing prompt search effectiveness with $\mathcal{R}_{\text{PDAR}}$, which aligns outputs for clean and masked sentences. Finally, employing the superior prompt ensemble technique elevates CACC to 85.70% and reduces ASR from 91.88% to 50.55%, indicating significant improvements over the baseline method. Similarly, on AgNews dataset, CR-UTP surpasses the baseline with a 3.33% increase in CACC to 84.27% and a 3.6% decrease in ASR, highlighting CR-UTP's effectiveness. Our CR-UTP method is also effective on the longer datasets such as Yelp dataset, which enhances the CACC and the PACC by 9.52% and 13.5%, respectively. Additionally, it reduces the ASR by 10.35%.

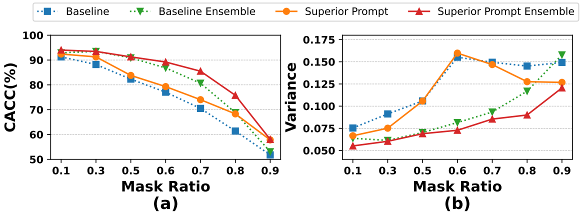


Figure 4: (a) Clean accuracy and (b) variance of proposed methods under different mask ratio.

Mask Ratio. To examine the effect of mask ratio on clean accuracy, we conduct experiments on the SST-2 dataset with varying mask ratios. Results in Figure 4 (a) show that while the baseline method's accuracy sharply drops from 91.27% to 51.78% as the mask ratio increases from 10% to 90%, our superior prompt search technique leads to a more gradual decline, from 92.42% to 57.82%. Additionally, employing our superior prompt ensemble method maintains a higher accuracy of 85.70% even at a 70% mask ratio, representing a significant improvement over the baseline method.

In Figure 4 (b), the variance analysis of certified accuracy shows that while increasing the mask ratio results in higher variance for both baseline and superior prompt methods, the use of ensemble techniques, particularly the superior prompt ensemble method, reduces variances, providing a more consistent output despite the effects of masking. The variance peaks at the 60% mask ratio, indicating the highest sentence diversity. This suggests that the variance is influenced not only by the volume of information loss due to masking but also by the diversity of sentences resulting from random masking. However, the employment of ensemble techniques, even with baseline ensemble (vanilla prompts, not superior prompts), results in a more gradual increase in variance. This stabilization is likely due to the ensemble's ability to aggregate insights from multiple prompts, delivering a more consistent and reliable output despite the information loss introduced by masking. The superior prompt ensemble technique further reduces the variances.

Ensemble Numbers. To investigate the impact of the number of prompts within the superior prompt ensemble on clean accuracy, we conducted experiments on the SST-2 dataset using a large mask ratio of 70% to amplify the ensemble number's effect on output performance. To mitigate the potential impact of differences in prompt selection perfor-

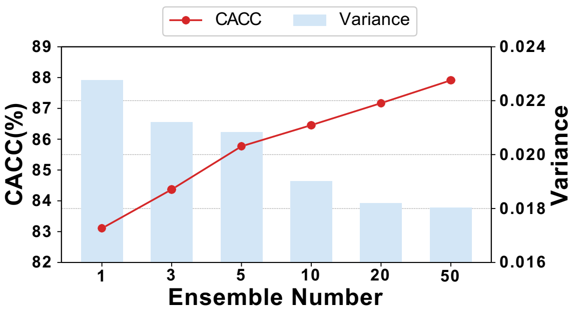


Figure 5: Clean accuracy and variance of CR-UTP under different ensemble numbers.

mance on the output, each ensemble was selected to have similar mean accuracy. The results depicted in Figure 5 show that as the number of prompts in the ensemble increases from 1 to 50, there is a consistent improvement in clean accuracy, rising from 83.21% to 87.82%, accompanied by a corresponding decrease in variance. These findings indicate that a larger ensemble leads to more stable and accurate predictions. This enhancement can be attributed to the ensemble's capacity to integrate diverse insights from multiple prompts, reducing the impact of any single erroneous prediction and fostering a consensus that is more resilient to the introduction of masks.

Table 3: Evaluation on large language models.

Model	Llama2-7b			GPT-3.5		
	CACC	ASR	PACC	CACC	ASR	PACC
w/o defense	90.40	88.17	53.82	92.01	96.88	51.94
Our baseline	73.89	83.14	55.03	75.34	86.72	56.19
CR-UTP	84.68	51.47	72.95	85.32	50.75	74.86

Evaluation on Large Language Models. We demonstrate the effectiveness of our CR-UTP method on large models like Llama2-7b and GPT-3.5. The experiments, conducted on the SST-2 dataset against the UTP attack TrojLLM, are shown in Table 3. For both the Llama2-7b and GPT-3.5 models, our CR-UTP approach improves the CACC by over 10% compared to our baseline, while also achieving a reduction in ASR of more than 30%.

Comparison with Adversarial Training. We compare the empirical defence effects of adversarial training (Yoo and Qi, 2021) and our CR-UTP in Table 4. Our CR-UTP significantly reduces the ASR by over 30% while maintaining similar accuracy, outperforming adversarial training. CR-UTP consistently defends against various adversarial attacks, such as TrojLLM and TextFooler, unlike adversarial training, which shows inconsistent defense effectiveness. For instance, adversarial training effectively reduces ASR from 91.88% to 80.68% for attacks it was trained against TrojLLM, but it provides minimal defense against different attacks (TextFooler), only reducing ASR from 92.27% to 91.13%.

Table 4: Comparison with adversarial training.

Attack	TrojLLM			TextFooler		
	CACC	ASR	PACC	CACC	ASR	PACC
w/o defense	92.69	91.88	53.76	92.69	92.27	8.13
Adv. training	85.94	80.68	59.82	85.94	91.13	8.81
CR-UTP	85.70	50.55	73.04	85.28	37.39	62.61

6 Limitation

The limitations of our paper are as follows: (i) Certified Accuracy. Although our CR-UTR has demonstrated improvements in certified accuracy and reduced ASR, achieving state-of-the-art results, there remains a gap between clean accuracy and certified accuracy. (ii) Broader Applications. While our CR-UTP primarily concentrates on classification tasks, broadening its application to encompass other NLP tasks like generation (Xue et al., 2024) would present a captivating expansion of our research. (iii) Efficiency. The CR-UTP with prompt ensemble would result in higher inference overhead compared to using just one prompt. However, it is important to note that a superior prompt can significantly enhance the effectiveness of defense strategies. Moreover, the Superior Prompt Search is an offline process to train the policy model which could be reused to generate multiple superior prompts in several seconds.

7 Conclusion

In this paper, we address the challenge of certifying language model robustness against Universal Text Perturbations (UTPs) and input-specific text perturbations (ISTPs). We introduce the *superior prompt search* method and the *superior prompt ensembling* technique to enhance certified accuracy against UTPs and ISTPs. Our approaches achieve state-of-the-art results, ensuring stability and reliability in language model predictions across diverse attack scenarios.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Mansour Al Ghanim, Muhammad Santriaji, Qian Lou, and Yan Solihin. 2023. Trojbits: A hardware aware inference-time attack on transformer-based language models. In *ECAI 2023*, pages 60–68. IOS Press.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. *arXiv preprint arXiv:1804.07998*.

Nabiha Asghar. 2016. Yelp dataset challenge: Review rating prediction. *CoRR*, abs/1605.05362.

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

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, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020b. Language models are few-shot learners.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. 2019. Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1310–1320. PMLR.

Mingkai Deng, Jianyu Wang, Cheng-Ping Hsieh, Yihan Wang, Han Guo, Tianmin Shu, Meng Song, Eric Xing, and Zhiting Hu. 2022. Rlprompt: Optimizing discrete text prompts with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3369–3391.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. c: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE.

Shreya Goyal, Sumanth Doddapaneni, Mitesh M Khapra, and Balaraman Ravindran. 2023. A survey of adversarial defenses and robustness in nlp. *ACM Computing Surveys*, 55(14s):1–39.

James B Heaton, Nick G Polson, and Jan Hendrik Witte. 2017. Deep learning for finance: deep portfolios. *Applied Stochastic Models in Business and Industry*, 33(1):3–12.

Miklós Z Horváth, Mark Niklas Müller, Marc Fischer, and Martin Vechev. 2021. Boosting randomized smoothing with variance reduced classifiers. *arXiv preprint arXiv:2106.06946*.

Jongheon Jeong and Jinwoo Shin. 2020. Consistency regularization for certified robustness of smoothed classifiers. *CoRR*, abs/2006.04062.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

J. Zico Kolter and Eric Wong. 2017. Provable defenses against adversarial examples via the convex outer adversarial polytope. *CoRR*, abs/1711.00851.

Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight poisoning attacks on pretrained models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2793–2806.

Changjiang Li, Ren Pang, Zhaohan Xi, Tianyu Du, Shouling Ji, Yuan Yao, and Ting Wang. 2023a. An embarrassingly simple backdoor attack on self-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4367–4378.

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

Linyi Li, Xiangyu Qi, Tao Xie, and Bo Li. 2020b. Sok: Certified robustness for deep neural networks. *CoRR*, abs/2009.04131.

Linyi Li, Jiawei Zhang, Tao Xie, and Bo Li. 2023b. Double sampling randomized smoothing.

Chizhou Liu, Yunzhen Feng, Ranran Wang, and Bin Dong. 2021. Enhancing certified robustness via smoothed weighted ensembling.

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*.

Qian Lou, Yepeng Liu, and Bo Feng. 2022. Trojtext: Test-time invisible textual trojan insertion. In *The Eleventh International Conference on Learning Representations*.

Monika A Myszczynska, Poojitha N Ojamies, Alix MB Lacoste, Daniel Neil, Amir Saffari, Richard Mead, Guillaume M Hautbergue, Joanna D Holbrook, and Laura Ferraiuolo. 2020. Applications of machine learning to diagnosis and treatment of neurodegenerative diseases. *Nature Reviews Neurology*, 16(8):440–456.

Matan Ostrovsky, Clark W. Barrett, and Guy Katz. 2022. An abstraction-refinement approach to verifying convolutional neural networks. *CoRR*, abs/2201.01978.

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.

Hadi Salman, Mingjie Sun, Greg Yang, Ashish Kapoor, and J. Zico Kolter. 2020. Black-box smoothing: A provable defense for pretrained classifiers. *CoRR*, abs/2003.01908.

Hadi Salman, Greg Yang, Jerry Li, Pengchuan Zhang, Huan Zhang, Ilya P. Razenshteyn, and Sébastien Bubeck. 2019. Provably robust deep learning via adversarially trained smoothed classifiers. *CoRR*, abs/1906.04584.

Erfan Shayegani, Md Abdullah Al Mamun, Yu Fu, Pedram Zaree, Yue Dong, and Nael Abu-Ghazaleh. 2023. Survey of vulnerabilities in large language models revealed by adversarial attacks. *arXiv preprint arXiv:2310.10844*.

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*, pages 1631–1642.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023a. Llama 2: Open foundation and fine-tuned chat models.

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

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*.

Xiaosen Wang, Hao Jin, and Kun He. 2019. Natural language adversarial attacks and defenses in word level. *CoRR*, abs/1909.06723.

Maurice Weber, Xiaojun Xu, Bojan Karlas, Ce Zhang, and Bo Li. 2020. RAB: provable robustness against backdoor attacks. *CoRR*, abs/2003.08904.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Tsui-Wei Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Duane Boning, Inderjit S. Dhillon, and Luca Daniel. 2018. Towards fast computation of certified robustness for relu networks.

Yiquan Wu, Kun Kuang, Yating Zhang, Xiaozhong Liu, Changlong Sun, Jun Xiao, Yueting Zhuang, Luo Si, and Fei Wu. 2020. De-biased court's view generation with causality. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 763–780.

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

Jiaqi Xue, Mengxin Zheng, Yebowen Hu, Fei Liu, Xun Chen, and Qian Lou. 2024. Badrag: Identifying vulnerabilities in retrieval augmented generation of large language models.

Jiaqi Xue, Mengxin Zheng, Ting Hua, Yilin Shen, Yepeng Liu, Ladislau Bölöni, and Qian Lou. 2023. TrojLLM: A black-box trojan prompt attack on large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Mao Ye, Chengyue Gong, and Qiang Liu. 2020. Safer: A structure-free approach for certified robustness to adversarial word substitutions. In *Annual Meeting of the Association for Computational Linguistics*.

Jin Yong Yoo and Yanjun Qi. 2021. Towards improving adversarial training of nlp models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 945–956.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Jiehang Zeng, Jianhan Xu, Xiaoqing Zheng, and Xuanjing Huang. 2023. Certified robustness to text adversarial attacks by randomized [mask]. *Computational Linguistics*, 49(2):395–427.

Runtian Zhai, Chen Dan, Di He, Huan Zhang, Boqing Gong, Pradeep Ravikumar, Cho-Jui Hsieh, and Liwei Wang. 2020. MACER: attack-free and scalable robust training via maximizing certified radius. *CoRR*, abs/2001.02378.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Zhen Zhang, Guanhua Zhang, Bairu Hou, Wenqi Fan, Qing Li, Sijia Liu, Yang Zhang, and Shiyu Chang. 2023. Certified robustness for large language models with self-denoising.

Mengxin Zheng, Qian Lou, and Lei Jiang. 2023a. Trojvit: Trojan insertion in vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4025–4034.

Mengxin Zheng, Jiaqi Xue, Xun Chen, YanShan Wang, Qian Lou, and Lei Jiang. 2023b. Trojfsp: Trojan insertion in few-shot prompt tuning. *arXiv preprint arXiv:2312.10467*.

Mengxin Zheng, Jiaqi Xue, Yi Sheng, Lei Yang, Qian Lou, and Lei Jiang. 2023c. Trojfair: Trojan fairness attacks. *arXiv preprint arXiv:2312.10508*.

Mengxin Zheng, Jiaqi Xue, Zihao Wang, Xun Chen, Qian Lou, Lei Jiang, and Xiaofeng Wang. 2023d. Ssl-cleanse: Trojan detection and mitigation in self-supervised learning. *arXiv preprint arXiv:2303.09079*.

Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-Wei Chang, and Xuan-Jing Huang. 2021. Defense against synonym substitution-based adversarial attacks via dirichlet neighborhood ensemble. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5482–5492.

A Appendix

A.1 Training Time and Inference Efficiency

Regarding the training time, the Superior Prompt Search is an offline process before online inference, thus the prompt search and generation phase does not impact the online inference. As mentioned in Section 4, the prompt search normally takes about 3.8 hours to train the policy model using only one single Nvidia GeForce RTX-3090 GPU. Once the policy model is trained, one could reuse it to generate multiple superior prompts in several seconds. The superior prompt is short and effective, comprising up to 5 tokens, resulting in an overhead (from appending the prompt compared to not using one) of no more than 10% for both the SST and AgNews datasets. Additionally, datasets with longer texts (Yelp) exhibit much lower overhead ratios, i.e., less than 5%. We highlight that superior prompt significantly enhances defense effectiveness. As the Table 2 shows, on the SST-2 dataset, it yields an improvement of over 15% in clean accuracy and a reduction of more than 30% in ASR compared to our baseline.

The generation of ensemble prompts runs parallel to the search for the superior prompt, occurring before the online inference phase. The certified efficiency of this method is closely linked to the number of inference executions, which is a product (kn) of the ensemble number (k) and the sampling number (n). To maintain efficiency, one could reduce the sampling number (n) when employing ensemble prompts ($k > 1$) to keep a similar or the same product kn. For instance, sampling a single superior prompt 5000 times may yield a certified accuracy of 54.5%; however, an ensemble of 5 superior prompts (k=5) requires only 1000 samplings to reach a certified accuracy of 58.1%. For the defense inference against a specific attack, the number of n can be much smaller, e.g., 50-100, for efficiency consideration. Also, We underscore the efficiency of prompt ensembling over model ensembling, owing to the prompt generation's speed and low memory footprint, as opposed to the more resource-intensive generation and memory demands of model ensembling.