I-GUARD: Interpretability-Guided Parameter Optimization for Adversarial Defense

Mamta and Oana Cocarascu
King's College London
{mamta.name, oana.cocarascu}@kcl.ac.uk

Abstract

Transformer-based models are highly vulnerable to adversarial attacks, where even small perturbations can cause significant misclassifications. This paper introduces *I-Guard*, a defense framework to increase the robustness of transformer-based models against adversarial perturbations. *I-Guard* leverages model interpretability to identify influential parameters responsible for adversarial misclassifications. By selectively fine-tuning a small fraction of model parameters, our approach effectively balances performance on both original and adversarial test sets. We conduct extensive experiments on English and code-mixed Hinglish datasets and demonstrate that *I-Guard* significantly improves model robustness. Furthermore, we demonstrate the transferability of *I-Guard* in handling other character-based perturbations.

1 Introduction

Transformer-based Pre-trained Language Models (PLM) and Large Language Models (LLM) have achieved remarkable performance across various NLP tasks (Waswani et al., 2017). However, extensive research has shown that they are vulnerable to small perturbations which can significantly affect their performance (Wang et al., 2022b; Le et al., 2022; Qiang et al., 2024; Nguyen and Le, 2024). These weaknesses raise concerns about the robustness and reliability of such models, particularly in real-world applications.

There have been many attempts to study the adversarial robustness of these models across different NLP tasks (Sofi et al., 2022a; Goyal et al., 2023a; Ballout et al., 2024; Mamta and Cocarascu, 2025). Numerous studies have explored strategies to improve adversarial robustness in transformer-based models (Devlin et al., 2019), with data augmentation and adversarial training being widely examined (Goyal et al., 2023b; Morris et al., 2020a).

Data augmentation aims to improve model generalization by generating diverse text variations, but it often requires significant human effort to curate and validate high-quality augmented data. Adversarial training enhances model resilience by exposing it to adversarial examples during training. However, this approach is computationally expensive, requiring models to be re-trained from scratch using both original and carefully crafted adversarial data. Defense methods based on adversarial training also introduce new regularization functions (Liu et al., 2022; Yang et al., 2023) or apply perturbations to the embedding space (Ren et al., 2020; Nguyen Minh and Luu, 2022). However, these approaches depend on a continuous input space and struggle with the discrete nature of text, making it difficult to generate meaningful and semantically coherent interpolations (Nguyen and Le, 2024). Despite these efforts, achieving robust NLP models remains a challenging task. There is a growing need for transparent and efficient methods to enhance adversarial robustness without excessive manual intervention or computational overhead.

In this paper, we propose *I-Guard*, an interpretability-guided model training method to selectively modify model's parameters to increase the robustness of PLMs and LLMs towards adversarial attacks. For this, we employ model interpretability based on Shapley values (Lundberg and Lee, 2017) to analyze the contribution of each parameter in misleading the model in the presence of adversarial perturbations. For traditional PLMs such as BERT, *I-Guard* can be applied to the complete model. However, for LLMs with billions of parameters, directly interpreting every parameter is computationally expensive. To address this challenge, we integrate *I-Guard* with LoRA (Low-Rank Adaptation) (Hu et al., 2022) to find the contribution of each parameter in the LoRA adapters with the help of a probing task.

To measure the efficacy of *I-Guard*, we con-

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 22173–22188
November 4-9, 2025 ©2025 Association for Computational Linguistics

sider realistic adversarial attacks, such as phonetic and visual (i.e. LEET) perturbations, which are commonly used in social media. Phonetic variations are especially pronounced in code-mixed texts, where people frequently switch between languages when writing (Das et al., 2022) and spell words phonetically based on their native language's pronunciation, leading to diverse spelling patterns (Crystal, 2018). We evaluate *I-Guard* on two tasks, fact verification and code-mixed sentiment analysis (Hindi+English). In addition, we also investigate the transferability of our proposed approach to handle other character-based perturbations such as character deletion, character insertion, and character repetition. Our results show that the proposed defense mechanism effectively enhances the robustness of PLMs and LLMs against adversarial perturbations.

Our contributions are as follows:

1. We propose *I-Guard*, a novel interpretability-guided parameter training framework to enhance the robustness of PLMs and LLMs against phonetic and LEET-based adversarial attacks.
2. To demonstrate the generalizability of our approach, we conduct experiments on English and code-mixed datasets. Our results show that *I-Guard* maintains a balanced performance on both original test set and adversarial test set compared to other baselines.
3. We assess the transferability of *I-Guard* in strengthening model resilience against various character-level perturbations and achieve good defense performance.

2 Related Work

2.1 Adversarial Robustness

Several studies have shown that high-performing transformer-based models are susceptible to adversarial attacks and minor input perturbations (Lin et al., 2021; Neerudu et al., 2023; Gupta et al., 2024). Adversarial attacks have been explored in a variety of NLP tasks, including sentiment analysis (Jin et al., 2020a; Yuan et al., 2023; Mamta et al., 2023), machine translation (Wang et al., 2021b; Sai et al., 2021; Morris et al., 2020b), argument mining (Mayer et al., 2020; Sofi et al., 2022b), toxic content detection (Yuan et al., 2023), question answering (Goel et al., 2021; Moradi and Samwald, 2021; Kiela et al., 2021; Yuan et al., 2023; Gupta et al., 2024), and natural language inference (Wu et al., 2021; Morris et al., 2020b; Li et al., 2021; Yuan et al., 2023). Whilst the majority of character-based attacks rely on researchers defining text manipulation strategies, Le et al. (2022) introduced a realistic phonetic perturbation attack by collecting 600K human-written text variations from real-world data and utilizing them for adversarial attacks.

Adversarial defense strategies in NLP can be broadly classified as adversarial training-based defense, data augmentation, and regularization-based defense (Wang and Lin, 2025). Most works employ an adversarial training approach which retrains the model from scratch by adding adversarial examples to the training data (Li et al., 2019; Wang et al., 2021a, 2020; Jin et al., 2020b; Si et al., 2020). Other approaches incorporate adversarial training as a regularization technique; for example, Flooding-X (Liu et al., 2022), adversarial label smoothing (Yang et al., 2023), and temperature scaling (Raina et al., 2024) have proven effective in improving adversarial robustness. Some studies utilize adversarial training based on Generative Adversarial Networks (Ren et al., 2020) or Virtual Adversarial Training (Li and Qiu, 2021), where perturbations are introduced in the model's embedding space. Denoising-based methods have been proposed to improve adversarial robustness by applying changes to the embedding space of text (Yuan et al., 2024; Ji et al., 2024). However, these approaches often lack semantic correctness and can lead to incoherent modifications (Chen et al., 2020). In addition, there is a lack of explainability and transparency in the regularization-based defense methods (Goyal et al., 2023b). Another line of work focuses on ensembling-based methods, where multiple input text variants are generated at inference time, and predictions are aggregated across these variants. However, these approaches can be inefficient, as they require running the model on each variant, leading to increased inference time proportional to the number of ensembles (Li et al., 2023; Zeng et al., 2023).

2.2 Detection-based Defense

There are several methods for detecting adversarial perturbations in text (Goyal et al., 2023c). Some approaches focus solely on identifying and filtering these adversarial inputs, while others use spell checkers or rule-based techniques to correct the perturbations. However, these corrections often rely on predefined rules or dictionaries, which may be ineffective when the spelling of a word devi-

ates from those present in the corpus or dictionary. Approaches for identifying adversarial perturbations include the Synonym Encoding Method (Wang et al., 2021c), frequency-aware randomization frameworks (Bao et al., 2021) for detecting word substitutions, and robust density estimation (Yoo et al., 2022), among others. Our objective is to improve the robustness of models against adversarial perturbations, in particular phonetic and LEET-based attacks, which are inspired by real-world noise in the data. Rather than filtering such perturbations, we aim to build models that are robust and capable of handling them effectively.

2.3 Transformers Interpretation

The remarkable success of pre-trained transformer-based models has driven researchers to explore their interpretability in-depth, aiming to explain their black-box nature (Petroni et al., 2019; Liu et al., 2019; Hewitt and Manning, 2019). Many works define neurons as dimensions within contextualized representations and investigate the linguistic information encoded by these representations (Durrani et al., 2020; Dalvi et al., 2019). Other studies focus on analyzing multi-head self-attention layers (Clark et al., 2019; Voita et al., 2019) and examine the roles of different attention heads across various tasks (Gould et al., 2024; Conmy et al., 2023; Hanna et al., 2024).

Several works (Geva et al., 2021; Dai et al., 2022a) have explored the neurons in the feed-forward neural networks within transformer models, revealing that neurons in these layers encode word patterns and conceptual knowledge. For instance, Dai et al. (2022b) examined knowledge neurons in feed-forward networks, demonstrating their function in encoding factual knowledge for fill-in-the-blank tasks in BERT. Similarly, Wang et al. (2022a) identified skill neurons within the feed-forward layers of pre-trained transformers after prompt tuning for a task, showing that these neurons capture task-specific abilities. More recently, Yu and Ananiadou (2024) proposed a method to study neuron-level knowledge attribution in large language models by identifying *query neurons* which activate *value neurons*, offering deeper insights into model predictions. Kulkarni and Weng (2024) proposed a test-time defense primarily for image classification. Their approach focuses on neuron-level interpretability, computing importance based on methods like Leave-One-Out (LO-IR) or CLIP-Dissect (CD-IR) to identify and mask unimportant neuron activations at test time.

Our work differs from the aforementioned studies as follows. Instead of identifying neuron contributions, we conduct a fine-grained analysis of model parameters, focusing on those responsible for misleading the model. Based on this analysis, we further fine-tune the model to enhance its robustness against adversarial manipulations.

3 Methodology

3.1 Problem Formulation

Attack Goal For a given input X consisting of n tokens $\{x_1, x_2, \ldots, x_n\}$ and an associated ground truth label y, the attack objective is to generate a perturbed version X' such that the target model M misclassifies it, i.e., $M(X') \neq y$ (untargeted attack). In the case of two-input tasks, where the input consists of X and E, the adversary applies perturbations only to X, resulting in a modified input X', while keeping E unchanged. The goal remains to mislead the model such that $M(X', E) \neq y$. The attack is carried out using phonetic and LEET-based perturbations, following prior work (Le et al., 2022; Das et al., 2022).[1]

Defense Goal The defense mechanism must meet the following requirements: *1)* Enhance robustness against phonetic and LEET-based perturbations; and *2)* Ensure that the performance on the actual test set remains consistent with the original performance.

3.2 I-Guard Framework

In this section, we present *I-Guard*, a defense framework that enhances the robustness of the model against phonetic and LEET-based perturbations. Figure 1 depicts the main steps in *I-Guard*. The adversarial generator applies perturbations to actual examples to mislead the trained model M. We then identify the influential parameters responsible for the misclassification of adversarial examples using a subset of the misclassified adversarial examples and their corresponding actual examples. To determine these influential parameters, *I-Guard* leverages a model interpretation technique based on Shapley values (Lundberg and Lee, 2017). In the case of LLMs, our approach involves identifying specific LoRA parameters that contribute most to adversarial misclassifications. These selected parameters are then optimized to improve the model's robustness.

[1] See Appendix B for adversarial examples.

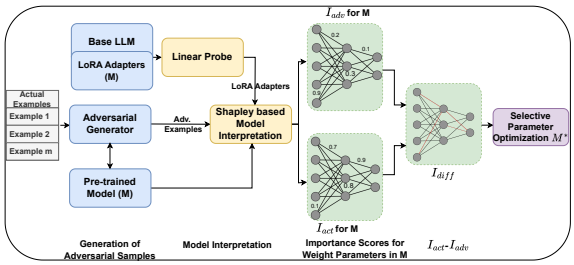

Figure 1: Key steps of *I-Guard*.

Adversarial Sample Generation To generate adversarial samples, we perform a black-box attack on BERT (Devlin et al., 2019) and Llama (Touvron et al., 2023) models. We follow Le et al. (2022) and Das et al. (2022) to apply perturbations to English and Hindi+English (Hinglish), respectively. To implement the phonetic and LEET-based perturbations for BERT, we follow Mamta and Ekbal (2022) to identify the important tokens. For Llama, we use feature ablation to identify the most important words for the model's predictions (Miglani et al., 2023). Next, we perturb the words in descending order of their importance scores until a successful attack is executed or all words have been perturbed.

Model Interpretation Shapley values originate from cooperative game theory, where players contribute to the overall outcome of a game (Lundberg and Lee, 2017). Here, we treat model parameters as players in a cooperative game, where their collective interaction determines the model's predictions. The Shapley value for each parameter represents its marginal contribution across all possible parameter combinations, providing a comprehensive measure of its importance. The mathematical formulation for the Shapley values is:

$$I_p(v) = \sum_{S \subseteq \Omega \setminus \{p\}} \frac{|S|!(|\Omega|-|S|-1)!}{|\Omega|!} [v(S \cup \{p\}) - v(S)] \quad (1)$$

where $\psi_p(v)$ represents the contribution score of parameter p, with $\Omega = \{1, 2, ..., n\}$ denoting the complete set of n model parameters. The term S encompasses all possible parameter subsets, and $v(S)$ represents the model's prediction using only the parameters in subset S, with other parameters left out. This formulation ensures a fair attribution of importance to each parameter by considering all possible combinations of parameter interactions.

Computing Shapley values (Equation 1) presents significant computational challenges, particularly for large neural networks. The computational complexity of $O(2^{\Omega})$ makes direct calculations inefficient for models with millions of parameters (Kang et al., 2023). Furthermore, in large-scale neural networks, the individual impact of single parameters on the overall prediction is typically minimal, making exhaustive computation inefficient.

To overcome this, we estimate the marginal contributions through random sampling by sampling k sized parameter subsets from Ω several times. This sampling-based approach significantly reduces computational overhead. As discussed in Kang et al. (2023), the marginal contribution calculation is approximated using the norm of weight-gradient products:

$$\begin{aligned} |v(S \cup p) - v(S)| &\approx |-L(\omega + \Delta\omega_i, x, y) + L(\omega, x, y)| \\ &\approx \left| \frac{\partial L}{\partial \omega_i} \Delta\omega_i \right| = \left| \frac{\partial L}{\partial \omega_i} \right| |\Delta\omega_i| = \left| \frac{\partial L}{\partial \omega_i} \right| |\omega_i|, \end{aligned} \quad (2)$$

This approximation leverages the first-order Taylor expansion of the loss function L. The contribution of parameter ω_i is approximated using the gradient of the loss with respect to the parameter and its magnitude.

Identifying Parameters for Adversarial Misclassifications To identify the key parameters responsible for adversarial misclassification, we follow a systematic approach involving adversarial sample generation, dataset partitioning, and Shapley value-based parameter importance estimation. The process is shown in Algorithm 1.

First, we generate adversarial samples for a specific perturbation type using the training set (line 2). We then pass the generated adversarial samples through the trained model M and analyze the predictions. Based on the classification outcomes, we make a subset of the dataset for misclassified Adversarial Samples. This includes original (D_{act}) and corresponding adversarial samples (D_{adv}) where examples are initially classified correctly by the model M, but their adversarial counterparts are misclassified (lines 3-13).

To understand the influence of individual model parameters in adversarial misclassifications, we apply Shapley value-based parameter importance estimation to both D_{act} and D_{adv} (lines 38-39). This method assigns importance scores (I_{act} and I_{adv}) to each parameter by evaluating its contribution to the model's predictions. The importance estimation process relies on repeated random sampling (K iterations), where, in each iteration m, a subset

Algorithm 1 Parameter Identification for Adversarial Misclassifications

Require: Model M, Training dataset $\mathcal{D}_{\text{train}}$, Perturbation type ϵ, Sampling iterations K, Sample size k, Importance threshold t, Fine-tuning epochs E
Ensure: Refined model M^*
1: // Generate adversarial samples
2: $\mathcal{D}_{\text{advtr}} \leftarrow \text{GenerateAdversarialSamples}(\mathcal{D}_{\text{train}}, M, \epsilon)$
3: // Initialize dataset partitions
4: $D_{\text{act}} \leftarrow \{\}$, $D_{\text{adv}} \leftarrow \{\}$
5: **for** (x, y) *in* $\mathcal{D}_{\text{train}}$ **do**
6: $x_{\text{adv}} \leftarrow \mathcal{D}_{\text{advtr}}[x]$
7: $y_{\text{pred}} \leftarrow M(x)$
8: $y_{\text{adv}} \leftarrow M(x_{\text{adv}})$
9: **if** $y_{\text{pred}} = y$ **and** $y_{\text{adv}} \neq y$ **then**
10: $D_{\text{act}} \leftarrow D_{\text{act}} \cup \{(x, y)\}$
11: $D_{\text{adv}} \leftarrow D_{\text{adv}} \cup \{(x_{\text{adv}}, y)\}$
12: **end if**
13: **end for**
14: // Initialize importance score vectors
15: $I_{\text{act}} \leftarrow \mathbf{0}_{|M|}$, $I_{\text{adv}} \leftarrow \mathbf{0}_{|M|}$
16: // Compute parameter importance
17: **function** CALCULATEIMPORTANCE(D, M)
18: **for** (x, y) *in* D **do**
19: **for** $m = 1$ *to* K **do**
20: $k \leftarrow k \times |M|$ ▷ Compute k% of elements
21: mask_indices $\leftarrow$ RandomSample(M, k) ▷ Select k random indices from complete model M or LoRA adapters
22: $M_{\text{masked}} \leftarrow M$
23: M_{masked}[mask_indices] $\leftarrow 0$
24: $\mathcal{L} \leftarrow \mathcal{L}(y, M_{\text{masked}}(x))$
25: Compute $\nabla_{M_{\text{masked}}} \mathcal{L}$
26: start_in, end_in $\leftarrow 0, 0$
27: **for** *par in* $M_{\text{masked}}.Params()$ **do**
28: start_in, end_in $\leftarrow$ end_in, end_in + $|par|$
29: **if** *par.gradient* $\neq$ None **then**
30: $w \leftarrow M_{\text{masked}}$[start_idx : end_idx]
31: I_{act}[start_idx : end_idx] $\leftarrow I_{act}$[start_idx : end_idx] + $||par.gradient.flatten() \cdot w||$
32: **end if**
33: **end for**
34: **end for**
35: **end for**
36: **return** I
37: **end function**
38: // Compute parameter importance for actual samples $I_{\text{act}} \leftarrow CalculateImportance(D_{\text{act}}, M)$
39: // Compute parameter importance for adversarial samples $I_{\text{adv}} \leftarrow CalculateImportance(D_{\text{adv}}, M)$
40: // Compute importance difference and identify crucial parameters
41: $I_{\text{diff}} \leftarrow I_{\text{adv}} - I_{\text{act}}$
42: $P_{\text{critical}} \leftarrow \text{SelectTopParameters}(I_{\text{diff}}, t)$
43: // Freezing selective parameters
44: $M^* \leftarrow M$
45: **for** *each layer* l *in* M^* **do**
46: **for** *each submodule* sm *in* l **do**
47: **if** $sm \in P_{\text{critical}}$ **then**
48: sm.requires_grad $\leftarrow$ True
49: **else**
50: sm.requires_grad $\leftarrow$ False
51: **end if**
52: **end for**
53: **end for**
54: Fine-tune M^* on D_{train} and D_{advtr}
55: **return** M^*

of parameters is randomly chosen to assess their marginal contribution. For subset selection, we use RandomSample(M, k), which selects k% parameter indices to be temporarily zeroed out, forming a zeroed out model M_{masked} (lines 18-23). Here, $|M|$ is the size of model, i.e., total number of parameters present in the model. A forward pass with M_{masked} model produces predictions $\hat{y} = M_{\text{masked}}(\mathbf{x})$, followed by computing the loss $\mathcal{L}(y, \hat{y})$. Backpropagation then yields gradients $\nabla_{M_{\text{masked}}} \mathcal{L}$, which, when multiplied by parameter values, approximate the marginal contributions (lines 24-31).

Finally, the importance scores for each parameter are accumulated across multiple iterations and batches (lines 27-34). Since there exists a one-to-one correspondence between the actual and adversarial subsets, some parameters which are crucial for correctly classifying D_{act} may not hold the same significance in D_{adv}. This discrepancy arises due to the perturbations altering the input space, causing certain parameters to play a more dominant role in misclassifications. Therefore, to pinpoint the parameters responsible for adversarial misclassification, we compute the difference between importance scores I_{adv} and I_{act} (line 41). This difference highlights parameters that significantly contribute to the misclassification of adversarial samples. *I-Guard* adapts parameter interpretation strategy for different types of language models, considering their architectural differences and computational requirements.

Pre-trained Language Models For BERT like models, *I-Guard* applies Shapley value-based model interpretation to the complete model as shown in Figure 1. This involves analyzing the contributions of parameters across all layers and components of the entire BERT model.

Large Language Models Due to the high computational cost associated with fine-tuning and interpreting LLMs, *I-Guard* applies Shapley-based interpretation to the LoRA adapters (Hu et al., 2022) as shown in Figure 1. *I-Guard* utilizes a model fine-tuned using the LoRA technique. We perform this initial fine-tuning with a Causal Language Modeling (CLM) objective, rather than a direct classification objective. This choice underscores the generic nature of our approach. After LoRA-based fine-tuning, we perform linear probing to adapt the LLM's learned representations for classification tasks. This is because the CLM loss, designed for next-token prediction, does not inherently provide a direct gradient signal which is required to

understand the influence of the parameters on a downstream classification task.

Probing We train a classifier (*probe*) on the top of hidden representation of the last layer to map this hidden representation to the task label. During fine-tuning, the weights of the LLM and LoRA adapters are frozen to ensure that the probe is using the existing task-specific information, rather than modifying the underlying representation. Therefore, only the parameters of the linear probe are updated during this training phase.

After training, we freeze parameters and use Shapley-based model interpretation (lines 17-37) to calculate the importance of LoRA parameters. As the trained probe transforms the hidden states into differentiable class logits, this transformation enables the calculation of a loss that is directly linked to the classification task.

Selective Parameter Optimization Based on the final importance scores, we identify the top t most influential parameters (line 42). In transformer-based models, each layer consists of Self-Attention Weights (Query, Key, and Values), Feed-Forward Network weights (intermediate or output), and Layer Norm Weights. Instead of fine-tuning individual weights, we selectively fine-tune the specific submodules (e.g., output Feed-Forward Network, submodules of LoRA) where these important parameters are located, while keeping the rest of the model frozen (lines 42-53). The model is then fine-tuned on a mix of actual and adversarial samples (line 54).

4 Experiments

4.1 Datasets

We conduct extensive evaluation of *I-Guard* on English and Hinglish datasets, covering two tasks.
Fact verification (English) We use CLIMATE-FEVER (Diggelmann et al., 2020) which contains claims related to climate change along with their evidence. Each claim is labeled with one of three classes: supports, refutes, and not-enough-info (NEI). The dataset includes 7,675 annotated claim-evidence pairs split into train (5,756), validation (767), and test set (1,152).
Sentiment analysis (Hindi + English) We use a Hinglish dataset comprising posts from various public Facebook pages (Joshi et al., 2016). Each Hinglish post is labeled with one of three sentiment categories: positive, negative, or neutral. In total, the dataset includes 3,879 instances split into train (2,482), validation (621), and test set (776).[2]

4.2 Baselines

We compare *I-Guard* with the following baselines:
BERT (Devlin et al., 2019): We fine-tune BERT-base and BERT-base-multilingual (mBERT) models on English and Hinglish datasets, respectively, by adding a dense layer on top of it.
Llama: We fine-tune Llama-3.2-1B model on English and Hinglish datasets using LoRA adapters with CLM objective.
Adversarial Training: We create adversarial samples using phonetic and LEET-based perturbations and incorporate them into the original training data, re-training the model from scratch on the combined dataset.
LoRA based Adversarial Training: We re-train the LoRA adapters of Llama from scratch on the combined dataset.
I-Guard-Adv: We fine-tune the selective parameters of the model on only adversarial samples.

5 Results and Discussion

Table 1 shows the accuracy and F_1 score on both the actual test set and adversarial test set (TS), as well as the percentage of fine-tuned parameters (PT) for both datasets. For Llama, we show the number of fine-tuned LoRA adapters.

English Dataset The accuracy and F_1 scores for BERT-base and Llama models drop significantly under phonetic and LEET-based perturbations. For instance, under phonetic perturbations for BERT (FV Phonetic), the F_1 score drops by 36.72%. The most widely used adversarial defense, adversarial training, significantly improves both accuracy and F_1 scores for these perturbations in BERT and Llama models. However, this approach is computationally expensive, as the model needs to be trained from scratch (100% PT and all LoRA adapters).

I-Guard-Adv also effectively handles phonetic and LEET-based perturbations by fine-tuning much fewer parameters than adversarial training. In phonetic perturbations, *I-Guard* achieves an F_1 score of 59.53% while fine-tuning only 21.79% of BERT's parameters, compared to adversarial training, which requires 100% fine-tuning. Similarly, for Llama, only 20 LoRA adapters are updated. For LEET-based perturbations also, *I-Guard* enhances the robustness while fine-tuning fewer parameters.

[2]See Appendix A for detailed experimental details.

		Phonetic					LEET				
		Original TS		Adv TS			Original TS		Adv TS		
		Acc	F1	Acc	F1	PT	Acc	F1	Acc	F1	PT
FV	BERT-base	71.61	62.70	51.21	25.98	-	71.61	62.70	49.91	22.52	-
	Adv training	71.44	61.02	75.17	**65.42**	100%	71.61	59.96	76.90	**68.24**	100%
	I-Guard-adv	60.58	58.03	73.87	64.38	**21.79%**	59.80	57.21	73.43	64.98	**26.11%**
	I-Guard	69.27	**62.40**	69.79	59.53	**21.79%**	72.13	**63.74**	74.82	66.82	**26.11%**
SA	mBERT	67.26	62.58	40.46	28.09	-	67.26	62.58	40.25	20.77	-
	Adv training	67.13	64.80	60.43	**56.93**	100%	65.72	60.83	72.16	**67.60**	100%
	I-Guard-adv	68.68	65.00	56.31	50.38	**28.98%**	59.77	58.17	71.77	66.39	**27.67%**
	I-Guard	67.96	**64.94**	56.31	51.36	**28.98%**	67.78	**64.72**	67.13	61.20	**27.67%**

Table 1: Results on Fact Verification (FV) and Sentiment Analysis (SA) for BERT. Here, TS: test set, Adv TS: Adversarial test set, PT: percentage of fine-tuned parameters.

		Phonetic					LEET				
		Original TS		Adv TS			Original TS		Adv TS		
		Acc	F1	Acc	F1	LM	Acc	F1	Acc	F1	LM
FV	Llama	57.20	35.84	15.00	8.98	224	57.20	35.84	18.76	10.81	224
	LoRA Adv	43.23	33.70	31.93	18.03	224	51.11	29.71	31.93	18.03	224
	I-Guard-adv	59.45	**35.76**	42.75	**22.30**	**20**	57.60	32.16	39.26	20.92	**20**
	I-Guard	58.77	35.58	39.09	20.83	**20**	56.27	**33.24**	40.13	**21.02**	**20**
SA	Llama	45.74	38.6	19.29	14.91	224	45.74	38.60	22.55	13.56	224
	LoRA Adv	51.80	39.04	35.28	25.19	224	50.23	**42.73**	39.94	29.11	224
	I-Guard-adv	51.15	**39.73**	36.98	26.05	**25**	51.59	41.35	38.53	28.49	**24**
	I-Guard	52.06	39.59	38.65	**27.97**	**25**	49.87	38.68	40.46	**30.01**	**24**

Table 2: Results on Fact Verification (FV) and Sentiment Analysis (SA) for LLama. Here, TS: test set, Adv TS: Adversarial test set, LM: fine-tuned LoRA modules.

		BERT		*I-Guard*	
		Acc	F1	Acc	F1
FV	Char Delete	49.82	22.52	68.05	55.92
	Char Insert	49.56	22.47	66.49	52.99
	Char Repetitiion	50.00	22.93	68.66	58.05
SA	Char Delete	37.75	26.20	52.83	48.15
	Char Insert	35.69	26.63	48.84	44.65
	Char Repetitiion	46.64	38.06	54.76	50.04

Table 3: Transferability of *I-Guard* to other attacks.

Moreover, adversarial training and *I-Guard-adv* reduce the F_1 score on the original test set (BERT). In contrast, *I-Guard* not only improves robustness against adversarial perturbations, but also enhances performance on the original test set, thereby maintaining a good balance between performance on both the original and adversarial test sets.

Hinglish Dataset We observe a similar phenomenon on the sentiment analysis task, where *I-Guard* achieves better generalization compared to adversarial training-based defense for the BERT and Llama models. This demonstrates that *I-Guard* can enhance robustness against adversarial text perturbations by fine-tuning fewer parameters compared to adversarial training-based defense. We also conducted experiments on a larger dataset (see Appendix D).

5.1 Transferability to other Perturbations

In addition, we assess the transferability of model trained using *I-Guard* to other types of adversarial attacks. Specifically, we generate adversarial test sets using character repetition, character deletion, and character insertion perturbations (Moradi and Samwald, 2021) as discussed in Section 3.2.

The results are presented in Table 3. We observe a significant drop in accuracy and F_1 score for BERT (base and multilingual) across all types of perturbations in both English and Hinglish. However, our proposed method demonstrates robustness against these perturbations, leading to improved accuracy and F_1 scores in both languages. This indicates that *I-Guard* not only defends against the phonetic perturbations considered, but also offers enhanced generalization capabilities to handle other related perturbations effectively.

5.2 Ablation Study

Affect of t We experiment with different values of t (top t parameters from I_{diff}) to understand their impact on the robustness and actual performance of the model. Table 4 presents BERT's behaviour for various parameter values. For the English language, we observe that fine-tuning only 0.016% (for t=50) of the model parameters significantly improves its robustness against phonetic perturbations. Similarly, for LEET-based perturba-

		Phonetic					LEET				
		Original TS		Adv TS			Original TS		Adv TS		
	t	Acc	F1	Acc	F1	PT	Acc	F1	Acc	F1	PT
FV	50	67.88	61.58	68.48	57.07	0.016	67.27	60.91	67.36	56.50	0.018
	100	68.14	61.60	68.48	57.57	0.020	67.53	60.82	68.66	58.28	0.023
	150	69.18	62.34	69.77	59.39	21.791	67.62	60.95	68.83	58.51	0.024
	200	69.35	62.54	69.79	59.39	21.793	**72.13**	**63.74**	**74.82**	**66.82**	**26.105**
	250	**69.27**	**62.40**	**69.79**	**59.53**	**21.794**	72.04	63.85	73.78	64.99	26.106
	300	69.27	62.40	69.79	59.53	21.798	72.04	63.85	73.78	64.99	26.106
SA	50	68.42	64.08	48.32	39.97	0.019	68.29	63.82	47.42	34.55	0.017
	100	68.29	63.81	48.06	39.55	0.020	68.55	65.67	64.17	57.89	0.021
	150	68.42	64.28	51.03	44.64	0.022	67.65	64.51	65.07	58.53	0.382
	200	68.42	64.28	51.03	44.64	0.023	**67.78**	**64.72**	**67.13**	**61.20**	**27.67**
	250	67.91	63.94	56.31	51.36	24.65	67.78	64.71	67.13	61.20	30.34
	300	**67.96**	**64.94**	**56.31**	**51.36**	**28.98**	67.43	64.21	67.13	61.20	32.21

Table 4: Affect of choosing different parameter values (top t).

tions, fine-tuning just 0.018% of the model parameters greatly enhances its robustness.

Increasing the value of t results in improved performance on both the actual and adversarial test sets. However, further increasing the number of fine-tuned parameters can negatively impact the model's performance. For example, increasing t from 200 to 250 leads to performance decrease on adversarial test set (FV LEET). Further increasing the number of parameters from 250 to 300 results in the same performance as when t is set to 250. In case of the Hinglish language also, fine-tuning only 0.019% (phonetic) and 0.017% (LEET) of the parameters improves performance on both the actual and adversarial test sets. Increasing the number of parameters from 250 to 300 (SA phonetic) does not affect the robustness, but it does lead to an increase in performance on actual test set.

We also observe that for the top 50, 100, and 150 indices (FV LEET), the percentage (PT) remains low (less than 1%). This indicates that these indices correspond primarily to a small subset of parameters within a few submodules. However, when the number of top t parameters increases from 150 to 200, there is a substantial rise in PT, suggesting that the additional indices span across multiple submodules, significantly increasing the number of parameters being fine-tuned. Beyond 200 indices, the PT stabilizes at around 26.11%, indicating that most of the newly added weight parameters belong to already unfrozen submodules rather than introducing entirely new ones. This pattern is consistent for both languages and perturbations.

Random parameter selection We randomly select 30% model parameters and fine-tune them on (i) adversarial data; (ii) mixture of adversarial and original data to observe the impact of model interpretability on targeted parameter optimization. We compare this random selection with two variants of our approach, *I-Guard* and *I-Guard-adv*. The results for BERT are reported in Table 5.

	Data	Original TS		Adv TS	
		Acc	F1	Acc	F1
Random	Mix	67.55	64.67	62.98	57.76
I-Guard		67.78	64.72	67.13	61.2
Random	Adv	44.58	41.05	74.8	70.09
I-Guard-adv		59.77	58.17	71.77	66.39

Table 5: Results on LEET perturbations for sentiment analysis by ablating model interpretability component.

I-Guard outperforms random parameter fine-tuning on both the original and adversarial test sets. Thus, leveraging interpretability techniques to identify and optimize the most relevant parameters leads to more robust performance. When training the model only on adversarial test data, random selection achieves higher accuracy and F_1 score, but at the cost of a significant reduction in accuracy on the original test set. However, *I-Guard-adv* performs better on the original test set, highlighting that a targeted optimization approach focused on adversarial data can enhance model robustness without overfitting to adversarial examples. We also fine-tuned the last layer of the model (see results in Appendix C.2).

Affect of *k* We conducted an ablation study by varying the value of k in Shapley calculation and subsequently evaluating the fine-tuned model's performance (Llama). k represents the sample-size of parameters considered within each repeat of Shapley approximation. The results for different k values on both clean and adversarial performance, are presented in Table 7. We observe a slight improvement in adversarial test set accuracy and F_1 score as k increases from 10 to 50. This means a larger sample-size k leads to more fair and stable comparisons in Shapley approximation, poten-

	Original Claim	Adversarial Claim	Original Label	BERT-base	Llama	BERT Adv Training	LoRA Adv Training	*I-Guard*
1	There are many lines of evidence which clearly show that the atmospheric co2 increase is caused by humans.	There are many lines of evidence which clearly show that the atmospheric co2 **increse** is caused by humans.	Support	NEI	Support	Support	Support	Support
2	Climate change will also reduce the number of cold days and cold spells.	Climate **changee wil** also **reduece** the number of **cooooold** days and cold spells.	Support	NEI	NEI	Support	Support	Support
3	In the past, warming has never been a threat to life on earth.	In the past, warming has **neveeeeer** been a threat to life on earth.	Refute	NEI	Support	Refute	Refute	Refute
4	Temperatures in the arctic have soared recently, and scientists are struggling to explain exactly why.	-	Support	Support	Support	NEI	NEI	Support
5	New study confirms evs considerably worse for climate than diesel cars.	-	Refute	Refute	Refute	NEI	NEI	Refute

Table 6: Behaviour of different models on adversarial (1-3) and original examples (4-5).

	Original TS		Adv TS		
	Acc	F1	Acc	F1	k
Llama	45.74	38.60	22.55	13.56	
I-Guard	50.90	38.52	39.43	28.00	k=10
I-Guard	49.87	38.68	**40.46**	**30.01**	k=25
I-Guard	50.51	38.50	40.59	29.76	k=50

Table 7: Affect of k values on LEET perturbations for sentiment analysis.

tially yielding more accurate and robust importance scores for parameters critical to adversarial defense. The model's performance on the original test set remains stable across the different k values, demonstrating that the parameter selection method does not significantly compromise the model's clean accuracy, regardless of the k chosen. These results indicate that an even smaller value of k is effective. The reason for this stability lies in the combined power of k and the number of repeats. The multiple repeats ensure a comprehensive exploration of different parameter combinations. In each subsequent repeat, another set of parameters is randomly chosen for the k coalition. This extensive averaging across diverse combinations effectively mitigates the variance that might arise from smaller individual k values, leading to stable and robust final Shapley importance scores. We also observe the impact of sample size on model interpretation (Appendix C.1).

5.3 Qualitative Analysis

We analyze the behavior of different models on adversarial and original samples for the English language in Table 6, where we apply phonetic perturbations to the original claims (evidence remains unchanged), leading to misclassification by the BERT-base model (examples 1–3) and Llama (2-3). It can be seen that when perturbations are applied to claims from the *Support* (1–2) or *Refute* (3) classes, the BERT-base model misclassifies them to *NEI* class and Llama model misclassifies them to other two classes. However, both adversarial training and our proposed *I-Guard* correctly classify these adversarial samples, indicating that these models understand phonetic perturbations.

Examples 4 and 5 illustrate the behavior of the models on actual examples. In these examples, the adversarial training-based defense misclassifies claims from the *Support* and *Refute* classes as *NEI*, whereas *I-Guard* correctly classifies them.

6 Conclusion

We proposed *I-Guard*, a defense mechanism designed to handle adversarial perturbations in transformer-based models. *I-Guard* employs model interpretability to identify the parameters responsible for adversarial misclassifications and applies selective fine-tuning. While the widely used adversarial training-based defense improves robustness, it requires re-training from scratch and can lead to a drop in performance on the actual test sets. Our experiments on English and Hinglish datasets demonstrate that *I-Guard* enhances both adversarial and actual test performance, ensuring better generalization while fine-tuning fewer parameters. Additionally, we showed the transferability of our approach in handling other related character perturbations. In future work, we aim to extend *I-Guard* to other models to assess its efficiency and robustness in more complex architectures.

Limitations

This study, like most others, has limitations that could be addressed in future research. Currently, we focus only on English and Hinglish datasets and have not evaluated the performance of *I-Guard* on other low-resource language pairs. Expanding our approach to additional languages, particularly those

with different scripts, would be a valuable direction for future work. Additionally, our evaluation primarily considers text-based adversarial attacks, such as phonetic and LEET perturbations. However, adversarial robustness in multimodal settings (e.g., text with images) remains unexplored. Future work could investigate I-Guard's robustness in multimodal scenarios. We also plan to integrate the selection of t into an automated hyperparameter optimization framework (e.g., Bayesian optimization) to systematically identify the value that maximizes robustness or generalization.

Ethics Statement

We use publicly accessible datasets for our experiments, strictly for academic purposes and in full accordance with their licensing terms.

Acknowledgements

This research was supported by EPSRC (grant number EP/X04162X/1).

References

Mohamad Ballout, Anne Dedert, Nohayr Muhammad Abdelmoneim, Ulf Krumnack, Gunther Heidemann, and Kai-Uwe Kühnberger. 2024. FOOL ME IF YOU CAN! an adversarial dataset to investigate the robustness of LMs in word sense disambiguation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5042–5059, Miami, Florida, USA. Association for Computational Linguistics.

Rongzhou Bao, Jiayi Wang, and Hai Zhao. 2021. Defending pre-trained language models from adversarial word substitutions without performance sacrifice. *arXiv preprint arXiv:2105.14553*.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352.

David Crystal. 2018. *The Cambridge encyclopedia of the English language*. Cambridge university press.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022a. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022b. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. 2019. What is one grain of sand in the desert? analyzing individual neurons in deep nlp models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6309–6317.

Sourya Dipta Das, Ayan Basak, Soumil Mandal, and Dipankar Das. 2022. Advcodemix: Adversarial attack on code-mixed data. In *Proceedings of the 5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD)*, CODS-COMAD '22, page 125–129, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Thomas Diggelmann, Jordan L. Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. 2020. CLIMATE-FEVER: A dataset for verification of real-world climate claims. *CoRR*, abs/2012.00614.

Nadir Durrani, Hassan Sajjad, Fahim Dalvi, and Yonatan Belinkov. 2020. Analyzing individual neurons in pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4865–4880, Online. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495.

Karan Goel, Nazneen Fatema Rajani, Jesse Vig, Zachary Taschdjian, Mohit Bansal, and Christopher Ré. 2021. Robustness gym: Unifying the NLP evaluation landscape. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 42–55, Online. Association for Computational Linguistics.

Rhys Gould, Euan Ong, George Ogden, and Arthur Conmy. 2024. Successor heads: Recurring, interpretable attention heads in the wild. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Shreya Goyal, Sumanth Doddapaneni, Mitesh M. Khapra, and Balaraman Ravindran. 2023a. A survey of adversarial defenses and robustness in NLP. *ACM Comput. Surv.*, 55(14s):332:1–332:39.

Shreya Goyal, Sumanth Doddapaneni, Mitesh M. Khapra, and Balaraman Ravindran. 2023b. A survey of adversarial defenses and robustness in nlp. *ACM Comput. Surv.*, 55(14s).

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

Ashim Gupta, Rishanth Rajendhran, Nathan Stringham, Vivek Srikumar, and Ana Marasovic. 2024. Whispers of doubt amidst echoes of triumph in NLP robustness. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5533–5590, Mexico City, Mexico. Association for Computational Linguistics.

Michael Hanna, Ollie Liu, and Alexandre Variengien. 2024. How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. *Advances in Neural Information Processing Systems*, 36.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Jiabao Ji, Bairu Hou, Zhen Zhang, Guanhua Zhang, Wenqi Fan, Qing Li, Yang Zhang, Gaowen Liu, Sijia Liu, and Shiyu Chang. 2024. Advancing the robustness of large language models through self-denoised smoothing. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 246–257, Mexico City, Mexico. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020a. 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.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020b. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8018–8025. AAAI Press.

Aditya Joshi, Ameya Prabhu, Manish Shrivastava, and Vasudeva Varma. 2016. Towards sub-word level compositions for sentiment analysis of Hindi-English code mixed text. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2482–2491, Osaka, Japan. The COLING 2016 Organizing Committee.

Mintong Kang, Linyi Li, and Bo Li. 2023. Fashapley: Fast and approximated shapley based model pruning towards certifiably robust dnns. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 575–592.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics.

Akshay Kulkarni and Tsui-Wei Weng. 2024. Interpretability-guided test-time adversarial defense. In *European Conference on Computer Vision*, pages 466–483. Springer.

Thai Le, Jooyoung Lee, Kevin Yen, Yifan Hu, and Dongwon Lee. 2022. Perturbations in the wild: Leveraging human-written text perturbations for realistic adversarial attack and defense. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2953–2965, Dublin, Ireland. Association for Computational Linguistics.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2021. Contextualized perturbation for textual adversarial attack. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5053–5069, Online. Association for Computational Linguistics.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society.

Linyang Li and Xipeng Qiu. 2021. Token-aware virtual adversarial training in natural language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8410–8418.

Linyang Li, Demin Song, and Xipeng Qiu. 2023. Text adversarial purification as defense against adversarial attacks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 338–350, Toronto, Canada. Association for Computational Linguistics.

Bill Yuchen Lin, Wenyang Gao, Jun Yan, Ryan Moreno, and Xiang Ren. 2021. RockNER: A simple method to create adversarial examples for evaluating the robustness of named entity recognition models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3728–3737, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Qin Liu, Rui Zheng, Bao Rong, Jingyi Liu, Zhihua Liu, Zhanzhan Cheng, Liang Qiao, Tao Gui, Qi Zhang, and Xuan-Jing Huang. 2022. Flooding-x: Improving bert's resistance to adversarial attacks via loss-restricted fine-tuning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5634–5644.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pages 4768–4777.

Mamta and Asif Ekbal. 2022. Adversarial sample generation for aspect based sentiment classification. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 478–492, Online only. Association for Computational Linguistics.

Mamta Mamta, Zishan Ahmad, and Asif Ekbal. 2023. Elevating code-mixed text handling through auditory information of words. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15918–15932, Singapore. Association for Computational Linguistics.

Mamta Mamta and Oana Cocarascu. 2025. FactEval: Evaluating the robustness of fact verification systems in the era of large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 10647–10660, Albuquerque, New Mexico. Association for Computational Linguistics.

Tobias Mayer, Santiago Marro, Elena Cabrio, and Serena Villata. 2020. Generating adversarial examples for topic-dependent argument classification 1. In *Computational Models of Argument*, pages 33–44. IOS Press.

Vivek Miglani, Aobo Yang, Aram Markosyan, Diego Garcia-Olano, and Narine Kokhlikyan. 2023. Using captum to explain generative language models. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, pages 165–173, Singapore. Association for Computational Linguistics.

Milad Moradi and Matthias Samwald. 2021. Evaluating the robustness of neural language models to input perturbations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1558–1570, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020a. TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, Online. Association for Computational Linguistics.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020b. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126.

Pavan Kalyan Reddy Neerudu, Subba Oota, Mounika Marreddy, Venkateswara Kagita, and Manish Gupta. 2023. On robustness of finetuned transformer-based NLP models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7180–7195, Singapore. Association for Computational Linguistics.

Tuc Van Nguyen and Thai Le. 2024. Adapters mixup: Mixing parameter-efficient adapters to enhance the

adversarial robustness of fine-tuned pre-trained text classifiers. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21183–21203, Miami, Florida, USA. Association for Computational Linguistics.

Dang Nguyen Minh and Anh Tuan Luu. 2022. Textual manifold-based defense against natural language adversarial examples. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6612–6625, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Yao Qiang, Subhrangshu Nandi, Ninareh Mehrabi, Greg Ver Steeg, Anoop Kumar, Anna Rumshisky, and Aram Galstyan. 2024. Prompt perturbation consistency learning for robust language models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1357–1370, St. Julian's, Malta. Association for Computational Linguistics.

Vyas Raina, Samson Tan, Volkan Cevher, Aditya Rawal, Sheng Zha, and George Karypis. 2024. Extreme miscalibration and the illusion of adversarial robustness. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 2500–2525. Association for Computational Linguistics.

Yankun Ren, Jianbin Lin, Siliang Tang, Jun Zhou, Shuang Yang, Yuan Qi, and Xiang Ren. 2020. Generating natural language adversarial examples on a large scale with generative models. In *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2156–2163. IOS Press.

Ananya B Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M Khapra. 2021. Perturbation checklists for evaluating nlg evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234.

Chenglei Si, Zhengyan Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2020. Better robustness by more coverage: Adversarial training with mixup augmentation for robust fine-tuning. *CoRR*, abs/2012.15699.

Mehmet Sofi, Matteo Fortier, and Oana Cocarascu. 2022a. A robustness evaluation framework for argument mining. In *Proceedings of the 9th Workshop on Argument Mining*, pages 171–180, Online and in Gyeongju, Republic of Korea. International Conference on Computational Linguistics.

Mehmet Sofi, Matteo Fortier, and Oana Cocarascu. 2022b. A robustness evaluation framework for argument mining. In *Proceedings of the 9th Workshop on Argument Mining*, pages 171–180.

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

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

Tianlu Wang, Xuezhi Wang, Yao Qin, Ben Packer, Kang Li, Jilin Chen, Alex Beutel, and Ed Chi. 2020. CATgen: Improving robustness in NLP models via controlled adversarial text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5141–5146, Online. Association for Computational Linguistics.

Wenqi Wang, Run Wang, Lina Wang, Zhibo Wang, and Aoshuang Ye. 2021a. Towards a robust deep neural network against adversarial texts: A survey. *ieee transactions on knowledge and data engineering*, 35(3):3159–3179.

Xiao Wang, Qin Liu, Tao Gui, Qi Zhang, Yicheng Zou, Xin Zhou, Jiacheng Ye, Yongxin Zhang, Rui Zheng, Zexiong Pang, et al. 2021b. Textflint: Unified multilingual robustness evaluation toolkit for natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 347–355.

Xiaosen Wang, Jin Hao, Yichen Yang, and Kun He. 2021c. Natural language adversarial defense through synonym encoding. In *Uncertainty in Artificial Intelligence*, pages 823–833. PMLR.

Xiaozhi Wang, Kaiyue Wen, Zhengyan Zhang, Lei Hou, Zhiyuan Liu, and Juanzi Li. 2022a. Finding skill neurons in pre-trained transformer-based language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11132–11152, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xuezhi Wang, Haohan Wang, and Diyi Yang. 2022b. Measure and improve robustness in NLP models: A survey. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4569–4586, Seattle, United States. Association for Computational Linguistics.

Yang Wang and Chenghua Lin. 2025. Tougher text, smarter models: Raising the bar for adversarial defence benchmarks. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 6475–6491, Abu Dhabi, UAE. Association for Computational Linguistics.

A Waswani, N Shazeer, N Parmar, J Uszkoreit, L Jones, A Gomez, L Kaiser, and I Polosukhin. 2017. Attention is all you need. In *NIPS*.

T Wolf. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel S Weld. 2021. Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. 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 6707–6723.

Yahan Yang, Soham Dan, Dan Roth, and Insup Lee. 2023. In and out-of-domain text adversarial robustness via label smoothing. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 657–669.

KiYoon Yoo, Jangho Kim, Jiho Jang, and Nojun Kwak. 2022. Detection of adversarial examples in text classification: Benchmark and baseline via robust density estimation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3656–3672.

Zeping Yu and Sophia Ananiadou. 2024. Neuron-level knowledge attribution in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3267–3280, Miami, Florida, USA. Association for Computational Linguistics.

Lifan Yuan, Yangyi Chen, Ganqu Cui, Hongcheng Gao, Fangyuan Zou, Xingyi Cheng, Heng Ji, Zhiyuan Liu, and Maosong Sun. 2023. Revisiting out-of-distribution robustness in NLP: benchmarks, analysis, and llms evaluations. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Shilong Yuan, Wei Yuan, Hongzhi Yin, and Tieke He. 2024. Roic-dm: Robust text inference and classification via diffusion model. *arXiv preprint arXiv:2401.03514*.

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.

A Experimental Setup

We use PyTorch and HuggingFace (Wolf, 2019) to implement all models. We focus on transformer-based fine-tuned classification models due to their success in NLP tasks. The BERT-base model consists of 12 transformer layers, a hidden dimension of 768, and 12 self-attention heads, totaling 110 million trainable parameters.

I-Guard has access to 200 misclassified samples and their corresponding actual examples to calculate the contribution of each weight parameter. We set k (weights in each iteration) to 25% and the number of iterations K to 10. The model was optimized using Adam, with weight updates guided by categorical cross-entropy loss. All experiments were conducted on an NVIDIA A100-SXM4 GPU with 40 GB of memory.

B Adversarial Samples

Table presents a few adversarial examples for phonetic and LEET based perturbations.

C Ablation Study

C.1 Sample size for model interpretation

To observe the impact of sample size on model interpretation, we consider all misclassified samples and their corresponding actual samples for Shapley value calculations instead of randomly choosing 200 misclassified samples. Results are presented in Table 9. We observe that both variants exhibit similar performance with minor variations, demonstrating that *I-Guard* can effectively determine the importance of parameters using only a subset of the data rather than all misclassified samples.

C.2 Fine-tuning Last Layer

We fine-tune the last layer of the model on (i) adversarial samples and (ii) a mixture of adversarial and original data and compare against *I-Guard* and *I-Guard-adv*. Results for the sentiment analysis task under LEET-based perturbations are reported in Table 10.

We observe similar phenomena in this case. The results show that fine-tuning the last layer on adversarial data yields better performance on the adversarial test set compared to *I-Guard-adv*, but this

	Original Claim	Phonetic Perturbations	LEET Perturbations
1	Human activities (mainly greenhouse-gas emissions) are the dominant cause of the rapid warming since the middle 1900s (ipcc, 2013).	**Humaaaan** activities (mainly greenhouse-gas emissions) **areeee** the dominant cause of the rapid warming since the middle 1900s (ipcc, 2013).	**Hvm4n act1vit1es** (mainly greenhouse-gas emissions) **ar3** the dominant cause of the rapid warming since the middle 1900s (IPCC, 2013).
2	In the past, warming has never been a threat to life on earth.	In the past, warming has **neveeeeer** been a threat to life on earth.	In the past, **w4rming** has **n3v3r** been a threat to life on earth.
3	Sea level rise due to global warming is exaggerated.	**Seaa** level **risee** due to global warming is exaggerated.	Sea level **r1se** due to **gl0bal** warm1ng is **exagg3rated**
4	Clouds provide negative feedback.	Clouds **provde negtive** feedback.	Clouds **pr0vide n3gat1ve** feedback.

Table 8: Examples of adversarial inputs generated using phonetic and LEET perturbations

		Phonetic				LEET			
	S	Original TS		Adv TS		Original TS		Adv TS	
		Acc	F1	Acc	F1	Acc	F1	Acc	F1
FV	200	69.27	62.40	69.79	59.53	72.13	63.74	74.82	66.82
	All	70.35	62.87	69.27	59.13	69.79	62.28	73.01	65.98
SA	200	67.96	64.94	56.31	51.36	67.78	64.72	67.13	61.20
	All	67.64	64.87	56.70	51.50	67.19	64.69	67.55	61.86

Table 9: Affect of choosing different sample size for model interpretation. Here S is number of samples.

	Original TS		Adv TS	
	Acc	F1	Acc	F1
Last layer	48.45	46.15	73.58	68.65
I-Guard-adv	59.77	58.17	71.77	66.39
Last layer	63.56	60.66	63.33	57.16
I-Guard	67.78	64.72	67.13	61.2

Table 10: Results on LEET perturbations for sentiment analysis by fine-tuning only last layer.

	Original TS		Adv TS		
	Acc	F1	Acc	F1	PT
mBERT	90.31	73.32	71.62	46.28	-
Adv training	89.45	71.78	90.77	73.59	100%
I-Guard-adv	67.73	59.77	90.17	78.46	7.20%
I-Guard	**90.2**	**73.08**	**90.82**	**75.96**	**7.20%**

Table 11: Results on HSOL dataset for phonetic perturbations.

comes at the cost of large drop in accuracy on the original test set. In contrast, *I-Guard-adv* maintains a better balance between performance on both the original and adversarial test sets.

Additionally, when fine-tuning the last layer on a mix of original and adversarial data, we observe that *I-Guard* consistently outperforms this last layer fine-tuning approach on both actual and adversarial test set.

D Results on Large Dataset

To demonstrate the effectiveness of our proposed approach, we conducted experiments on the Hate Speech and Offensive Language (HSOL) dataset. The HSOL dataset consists of tweets categorized into three classes: hate speech, offensive but not hate speech, and neither offensive nor hate speech.

It contains 24,783 tweets, which are split into training (18,587), validation (2,478), and test (3,718) sets. Results for the phonetic perturbation-based adversarial attack are presented in Table 11. We observe that both accuracy and F_1 scores drop significantly under this attack. Further, adversarial training-based defense improves performance on the adversarial test set by fine-tuning 100% of the parameters on a mixture of actual and adversarial data. Our proposed approach effectively defends against this attack by fine-tuning only 7.20% parameters. It outperforms the adversarial training based defense on both the actual and adversarial test set. In addition, we compute Shapley values using a subset of the data (i.e., 200 misclassified

samples and their corresponding actual examples). This illustrates that *I-Guard* can enhance adversarial robustness by fine-tuning fewer parameters compared to adversarial training-based defenses, even in the case of larger datasets.