IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 13 December 2023, accepted 10 January 2024, date of publication 22 January 2024, date of current version 19 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3356568

RESEARCH ARTICLE

ZDDR: A Zero-Shot Defender for Adversarial Samples Detection and Restoration

MUSHENG CHEN, GUOWEI HE, AND JUNHUA WU
School of Software Engineering, Jiangxi University of Science and Technology, Nanchang 330000, China

Corresponding author: Junhua Wu (271045802@qq.com)

This work was supported in part by the Doctoral Startup Fund of Jiangxi University of Science and Technology under Grant 205200100402, and in part by the Scientific Research Project of the Jiangxi Provincial Department of Education under Grant GJJ200839.

ABSTRACT Natural language processing (NLP) models find extensive applications but face vulnerabilities against adversarial inputs. Traditional defenses lean heavily on supervised detection techniques, which makes them vulnerable to issues arising from training data quality, inherent biases, noise, or adversarial inputs. This study observed common compromises in sentence fluency during aggression. On this basis, the Zero Sample Defender (ZDDR) is introduced for adversarial sample detection and recovery without relying on prior knowledge. ZDDR combines the log probability calculated by the model and the syntactic normative score of a large language model (LLM) to detect adversarial examples. Furthermore, using strategic prompts, ZDDR guides LLM in rephrasing adversarial content, maintaining clarity, structure, and meaning, thereby restoring the sentence from the attack. Benchmarking reveals a 9% improvement in area under receiver operating characteristic curve (AUROC) for adversarial detection over existing techniques. Post-restoration, model classification efficacy surges by 45% compared to the offensive inputs, setting new performance standards against other restoration techniques.

INDEX TERMS Adversarial defense, large language model, natural language processing, model security, prompt engineering.

I. INTRODUCTION

Recent advancements in natural language processing (NLP) have revolutionized human-machine textual interactions, finding applications in domains such as reading comprehension [1], [2], machine translation [3], [4], [5], question answering [6], [7], text classification [8], [9], sentiment analysis [10], [11] and dependency parsing [12]. However, these NLP models grapple with the threats of adversarial text attacks. Attackers craft adversarial samples by subtly altering input texts, intending to deceive the victim models [13]. These manipulations, often imperceptible to human observers, can distort information flow, compromise data security, and erode public trust, thereby posing significant societal and technological challenges [14]. Despite their resemblance to genuine samples, models tend to misinterpret these adversarial samples with misplaced confidence [15], resulting in compromised efficacy and augmented computational burdens.

The research community has been vigorously exploring new algorithms for generating adversarial texts [16], [17], [18]. Adversarial attacks in NLP can be categorized into three types: character-level, word-level, and sentence-level attacks [9], [19]. Character-level offensives tweak individual characters within words, although such maneuvers are typically thwarted by spell checkers. Word-level strategies, on the other hand, substitute words with semantic equivalents, aiming to deceive without semantic shift, and present a particularly intricate defense challenge, as discussed herein. Sentence-level attacks reframe an entire sentence while retaining its core meaning, effectively serving as an extension of word-level strategies by modifying constituent words. As adversarial strategies evolve, they can be operationalized in both transparent (white-box) and opaque (black-box) environments. Their heightened efficiency and preserved semantic integrity render defenses ever more demanding.

The associate editor coordinating the review of this manuscript and approving it for publication was S. K. Hafizul Islam.

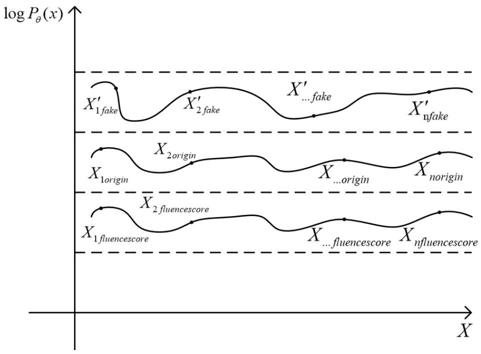


FIGURE 1. Illustration of fluency score calculation: given a victim model $F: X \rightarrow Y$, $X = (x_1, x_2, x_3, \ldots x_n)$ input data domain, $Y = (y_1, y_2, y_3, \ldots y_n)$ is the output data domain. Suppose $x \in X$ is a test input and the model correctly predicts its label as $y = F(x) \in Y$. An adversarial example of x is $x' \in X'$, such that $F(x') \notin Y$. First we use the victim model F to calculate the negative log-likelihood of all texts. Original texts $x_{1origin}, x_{2origin}, x_{3origin}, \ldots, x_{norigin}$ usually exhibit lower negative log-likelihoods. When weighted according to the grammar score, the fluency score of the original text $x_{1fluence}, x_{2fluence}, x_{3fluence}, \ldots, x_{nfluence}$ dropped further.

In NLP's adversarial defense landscape, three primary strategies have been proposed to develop supervised methods [20], [21]: creating similar environments during neural network training, demonstrating the robustness of the input region of the network, and identifying malicious inputs and correcting them with specialized techniques during training. Mimicking potential attack environments during the neural network's training phase, termed as "adversarial training" [22]. This approach enhances model robustness against perturbations by introducing adversarial disturbances into the training data, simulating potential attack scenarios. However, its effectiveness hinges on having a vast dataset. To consolidate model robustness, some researchers have aimed to verify the anti-interference ability of the network's input regions against adversarial attack. For instance, interval bound propagation (IBP) [23] offers a boundary strategy designed to train expansive and verifiable neural networks while ensuring robustness. Notwithstanding, such verifications aren't foolproof. Skilled attackers might still pinpoint unaccounted vulnerabilities, thereby circumscribing the defense's efficacy. Certain techniques actively scout for and address adversarial inputs during the training process. Notably, RDE [24] and UAPAD [25] stand out as supervised detection methodologies. By discerning the distinct traits of classifiers and gleaning relevant information, they have adeptly devised new classifier simulators attuned to adversarial text classification. However, as adversarial methodologies evolve, there's a pressing need to perennially update and recalibrate the identified adversarial dataset. In another approach, BERT-Defense [26] suggests a technique that assimilates context-dependent probabilities with embeddings of context-irrelevant hypotheses into a consolidated embedding. By masking tokens and leveraging masked language modeling (MLM) for predictions, they iteratively refine the approximation. While this heightened the restoration precision on adversarial sets, it somewhat diminished accuracy on the origin dataset, potentially impacting real-world applications.

This research introduces Zero-Shot Defender for Adversarial Sample Detection and Restoration (ZDDR), an innovative zero-shot, unsupervised framework tailored for the detection and restoration of adversarial samples in NLP. Uniquely, ZDDR stands resilient against both known and emergent attack vectors, sidestepping the need for labeled datasets. A pivotal observation underpinning this framework is the modus operandi of adversarial attacks in NLP. Notably, most adversarial strategies tamper with semantic constituents, modify structural elements, or intersperse extraneous characters, invariably resulting in compromised text fluency. In many instances, such alterations involve unwarranted word substitutions or structural tweaks that deteriorate sentence cohesiveness. The intent of this study is to harness these distributional discrepancies engendered by diminished fluency. To realize this, the study begins by quantifying the negative log-likelihood of sentences using the victim model. Subsequently, large language model (LLM) [27] is deployed to discern and score segments that deviate from grammatical conventions. These dual metrics, when weighted appropriately, facilitate the determination of a threshold for flagging adversarial samples, as depicted in Figure 1. For restoration, the research leans into the prompt engineering of LLM [28]. By crafting a universal prompt, the framework instructs LLM to rephrase identified adversarial constructs. Essentially, sentences tainted with adversarial elements are inputted into LLM. Guided by the prompt, the model is prompted to generate a new expression that's semantically congruent, yet distinct in phrasing, aiming to replace it with a more accurate rendition.

The contributions of this study can be summarized as follows:

1. This research introduces ZDDR. ZDDR represents a novel zero-shot, unsupervised framework that defends against adversarial samples, comprising two primary modules: DetectAttack (DA) and Restoration;
2. In its detection phase, the proposed DA method utilizes both the victim model to compute the negative log likelihood of the sentence, as well as a language model to comprehensively evaluate sentence fluency. Empirical assessments, spanning four attack algorithms and three representative datasets, underscore DA's superior detection performance relative to prevailing baselines;
3. This study champions the use of LLM for textual restoration post-attack. Through a meticulously designed generic prompt, adversarial constructs are rephrased to recover their original sentence mean. Comparative evaluations affirm the model's heightened defense efficacy when juxtaposed with compared attack scenarios.

II. RELATED WORK

A. ATTACK METHODS IN NLP

Within the domain of NLP, adversarial attacks predominantly manifest in three distinct forms: character-level, word-level, and sentence-level attacks [9], [29]. Character-level attacks focus on subtle manipulations within individual words. For instance, DeepWordBug [16] introduces an innovative character-level attack for black-box contexts. By determining word significance, it discreetly modifies selected words using tactics such as character swapping, flipping, insertion, or deletion. Another notable approach is HotFlip [30] which crafts adversarial samples via atomic character replacements, or flips, informed by the gradient of the one-hot input vector.

While word-level attacks pivot around the replacement of entire words, different strategies are often employed to find keyword and synonym replacements. A quintessential example is TextFooler [18], a black-box assault mechanism targeting BERT [31] for text classification. This method identifies key words from the victim model and cleverly replaces them with semantically consistent synonyms. PWWS [32] offers a similar strategy, but diverges in its methodology for synonym selection. Another salient study introduces TextBugger [17], adept at crafting adversarial samples in both black-box and white-box environments. In the latter, salient words are discerned via the Jacobian matrix, while in the former, pivotal sentences are earmarked first, followed by a scoring function pinpointing key word.

Sentence-level attacks typically involve sentences at various positions, ensuring linguistic fluency and semantic integrity. For instance, SCPLAN [33] crafts adversarial narratives that sustain semantic linearity but bewilder models by meticulously manipulating the syntactic parse tree. In another work adversarial perturbations are applied to the word embedding layer of CNN for text classification tasks, making the classification model robust against the worst perturbations.

B. DEFENSE

Defensive strategies against adversarial attacks in NLP can be delineated into three primary strategies: adversarial training, robustness certification, and adversarial identification and restoration.

Adversarial Training: Initially proposed by Goodfellow [34], adversarial training fortifies models against adversarial intrusions by embedding adversarial examples into the training dataset. The potency of these adversarial examples directly influences the model's robustness and its generalization capabilities. Follow-up research, such as LexicalAT [35], harnessed adversarial attacks in tandem with reinforcement learning to spawn resistant adversarial samples. Another intriguing direction involved leveraging adversarial training for cross-lingual text categorization, where a model trained on English data [36] was then employed to predict labels for non-English data, subsequently using these predictions as adversarial samples to bolster robustness. However, this adversarial training approach has limitations. It requires a large number of labeled clean samples and adversarial examples for supervised training, and cannot be easily generalized to real-world scenarios with imbalanced data and missing sample labels.

Robustness Certification: Primarily acknowledged for its efficacy in image processing, IBP ensures neural networks are sculpted to mitigate the extreme disparities between classification delineations and perturbed input zones. Within the task of textual classification, a study [37] proffered a rigorously vetted robust model capable of countering maximal perturbations. Through this approach, the peak disturbance's boundary is optimized using IBP, providing an upper limit for the discrete perturbation set in the word vector space. When encountering adversarial disturbances, IBP systematically computes an upper threshold for model losses. A noteworthy caveat of IBP is its computational heft, as it necessitates the demarcation of input-output frontiers at each network layer. Successive research introduced a perturbation space estimation technique anchored in model interpretation [38], which curtails computational demands while preserving estimation precision. Nonetheless, given constraints in segmentation techniques, it remains computationally intensive. Although these methods have achieved some success in addressing adversarial attacks, they have some limitations in adapting to different data distributions due to the constraints of supervised learning, especially insufficient adaptability to novel attacks.

Adversarial Identification and Restoration provide insights into various approaches. FGWS [39], a novel approach, identifies and replaces rare words in an input with prevalent synonyms. While efficacious for word-level attacks (where typical words are substituted with obscure synonyms), FGWS is less adept at countering character-level assaults where the adversarial entities are not recognized words, and thus, appropriate synonyms can't be ascertained. RDE [24] promotes robustness using disturbance detection via feature density estimation. Contrasting the traditional frequency-based likelihood estimation, RDE harnesses probability density models derived from features of pretrained architectures, such as BERT, emphasizing sentence density characteristics. Semi-character level recursive neural network (ScRNN) [40] model, functioning analogously to a spell-checker, predicts the appropriate word in the presence of disturbances. Its architecture mirrors traditional RNNs, ingesting semi-character vectors to anticipate the correct word in each interval, whilst contending with noise manifestations like jumbles, deletions, and insertions. Building upon ScRNN, ScRNN with Fallbacks [41] offers mechanisms to handle 'unknown' words by either leaving them as is, substituting with a neutral term, or turning to an extensive word recognition model. TREATED [42] stands out by defending against universal disturbances without assumptions, relying on multiple reference models to predict on both clean

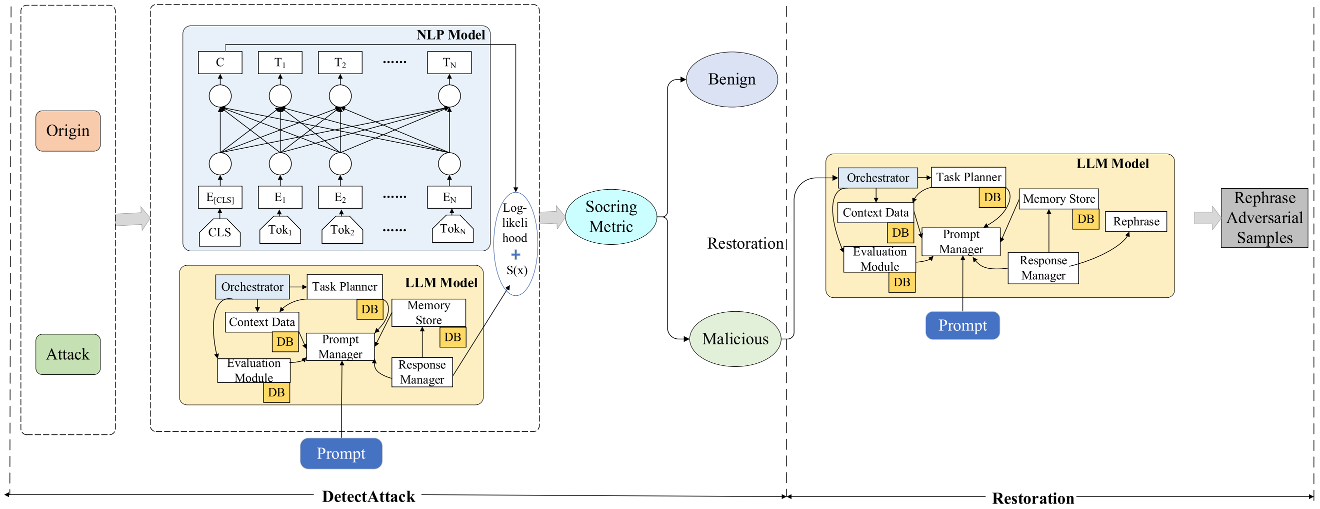


FIGURE 2. Defense framework structure diagram.

and adversarial samples. The consistency of these models across datasets is its key strength. Lastly, BERT-Defense [26] addresses the constraints of traditional spell-checkers in identifying and rectifying perturbations, this methodology harnesses BERT for tokenizing perturbed sentences. It subsequently computes context-irrelevant probability distributions via an adapted Levenshtein distance. By amalgamating contextually relevant probabilities and embeddings from all irrelevant hypotheses into a unified weighted embedding, it strives for restoration. The objective of recuperating tampered samples is realized through iterative token masking and subsequent prediction using MLM to achieve a proximate approximation. These methods heavily rely on resources such as dictionaries and corpora. This dependence causes the models to potentially fail when facing unknown language resources. It limits the model's comprehensive understanding of language, and the model's robustness is severely challenged.

III. METHODS

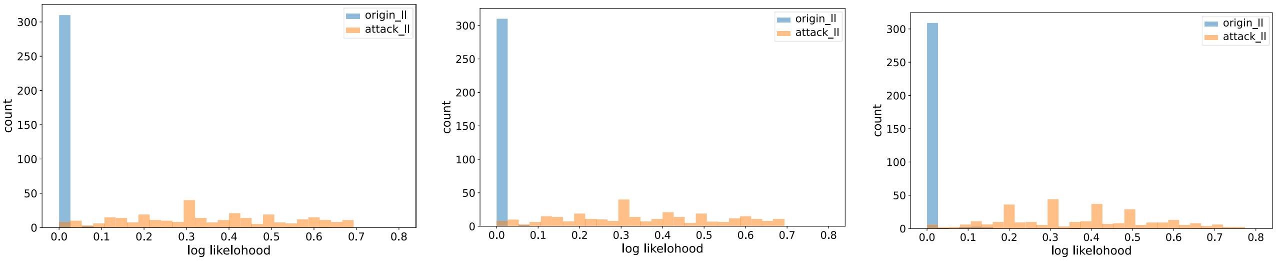
A. ZERO-SHOT DEFENDER FOR ADVERSARIAL SAMPLES DETECTION AND RESTORATION

A novel zero-shot defense framework has been introduced, tailored for addressing various text-based adversarial attacks.

At its core, the framework hinges on two pivotal strategies. First, it employs Adversarial Sample Detection, leveraging zero-shot capabilities to identify adversarial text samples. This method discerns adversarial text samples by exploiting the differential sensitivities of victim models to the negative log-likelihood of original and adversarial texts, thereby providing a scoring mechanism for various texts. LLM is employed to detect grammatical inconsistencies. By weighting the scores from both adversarial detection and grammatical analysis, a comprehensive text score distribution is derived. A threshold is then determined that maximizing AUROC for the identification of adversarial samples. The integration of grammatical scores with negative log-likelihoods results in comprehensive fluency scores for sentences. With the quantified score distribution, we can effectively distinguish adversarial samples with the help of some pre-calculated local sample thresholds. In the subsequent restoration phase, a rephrasing technique is advocated, leveraging LLM's grammatical analysis to comprehend and rectify adversarial content contextually. Through strategic text paraphrase, the framework enhances the model's comprehension of original content, effectively sidestepping adversarial manipulations. This holistic approach is depicted in Figure 2.

B. DETECTATTACK

Before delving into the proposed method, it is pertinent to understand the nuances of adversarial sample generation, which serves as the foundational inspiration for DA initiative. An examination of recent attack algorithms reveals a shared pattern. It begins by identifying vulnerable words and assessing them based on importance scores. The algorithms then iteratively modify words using character-level or word-level attacks, prioritizing words according to their importance rankings. As the iterative process continues, model confidence declines and sentence fluency deteriorates, ultimately leading to an altered prediction. Delving deeper into attack types, character-level attacks predominantly substitute task-relevant characters with aberrant or non-existent ones, word-level attacks often replace common words with complex or neutral variants, and sentence-level attacks modify a model's interpretative framework by adjusting a sentence's core structure and content, using tactics like restructuring, resequencing, or inserting/deleting key information, thus altering the model's understanding of the sentence's core message. In an ideal scenario, textual content, encompassing its



(a.1) IMDB

(a.1) SST-2

(a.3) AGNews

FIGURE 3. Negative log-likelihood score distributions for original vs. PWWS-attacked samples across datasets (IMDB, SST-2, AGNews): Attacked sentences exhibit a dispersed and elevated score range, whereas original samples cluster closely around zero.

Algorithm 1 DetectAttack detection adversarial samples

```
Input: text x of length n, victim model F, LLM,
 decision threshold $\varepsilon$
$-\log P(x) = -\log P(x_1) + \log P(x_2 \mid x_1) + \ldots$
  $+P(x_i \mid x_1 x_2 \ldots x_{n-1})$        //calculate negative log
                          probability with victim model.
$S(x) = LLM(x)$                       // LLM Inference score.
$g(x) = -\log P(x) + S(x)$       //fluence score calculation.
if $g(x) > \varepsilon$ then:
        return true              // probably adversarial sample
else:
        return false         // probably not adversarial sample
```

vocabulary, grammatical structure, and semantics, should exhibit inherent consistency. However, adversarial interventions fracture this harmony, obfuscating model interpretation and yielding text that strays from its natural fluency and coherence.

1) CALCULATE NEGATIVE LOG PROBABILITY WITH VICTIM MODEL

To elucidate the fluency disparity between original and adversarial samples, the negative log-likelihood scores of both sample types were computed and their respective distributions are illustrated in Figure 3. A pronounced distinction is evident between the negative log-likelihood distributions of the original and adversarial texts. Scores for the original samples gravitate towards 0, whereas the adversarial samples exhibit greater variability and magnitude. This observation aligns with the premise that more coherent text yields lower scores. These discernible contrasts provide a foundation for the subsequent detection method, which will be expounded upon in the ensuing sections.

Given a victim model $F : X \rightarrow Y$ as input data domain of X and output data domain of Y. Assume $x \in X$ as a test input, the model correctly predicts its label as $y = (x) \in Y$. The joint probability predicted by the victim model on a text x of length n is denoted as follows:

$$\begin{aligned} P(x) &= P(x)P(y|x) \\ &= P(x_1)P(x_1|x_2)\ldots P(x_i|x_1x_2\ldots x_{n-1}) \end{aligned} \quad (1)$$

where x_j denotes the j-th word of text x and the conditional probability of the j-th word given the preceding words is as follows:

$$P(x_j|x_1x_2\ldots x_{j-1}) \quad (2)$$

The negative log of the joint probability for text x can be expressed as follows:

$$\begin{aligned} -\log P(x) = -(&\log P(x_1) + \log P(x_1|x_2) + \ldots \\ &+ P(x_i|x_1x_2\ldots x_{n-1}) \end{aligned} \quad (3)$$

Transformer-based architectures, including BERT, XLNet, and GPT [43], employ tokenization techniques such as WordPiece [35] to shape their vocabulary. This stands in contrast to RNN-derived models like RNN, GRU, and LSTM, which lean more towards using the comprehensive English vocabulary. The tokenization strategy adopted by transformers truncates intricate words into shorter tokens, markedly diminishing vocabulary size. In the context of adversarial attacks, assailants deploy less frequently used token combinations, inducing segments of text to deviate in terms of grammar and semantics. Following tokenization, there might be deliberate distortions in sentence construction, diverging from standard linguistic patterns. This deviation amplifies the model's decoding uncertainty, culminating in heightened negative log-likelihood values for these tokens. A heightened negative log-likelihood can suggest that a text contains excessive redundancy not typically found in standard text. On the other hand, a diminished negative log-likelihood can hint at a higher concentration of information within the text. Higher negative log-likelihood indicates that the text contains too much redundant information that normal text does not possess. Conversely, lower negative log-likelihood suggests that the information in the text is concentrated.

2) LLM INFERENCE ERRORS

Tokenization techniques provide a fine-grained way to assess sentence fluency. It is pivotal to understand that these methods don't directly compute the negative log-likelihood; they instead shape the model's text decoding process and its probability distribution estimation. Elevated negative log-likelihood values in adversarial text don't unequivocally

TABLE 1. An examination of sentence fluency from the perspective of adherence to English grammar norms. Sentences are evaluated based on the following linguistic components.

constraint	Description
Lexicology	Includes the spelling, meaning, part of speech, etc. of words, as well as the correct use of compound words, derived words, etc.
Tense	Use various tenses correctly, such as the general Present tense tenses, Past tense tenses, Future tense tenses, etc., to express actions or states at different times.
Voice	Use Active voice and Passive voice correctly to express the performer and receiver of the action.
Articles and Determiner	Correctly use articles (a, an, the) and Determiner (this, that, these, those, some, any, etc.) to determine the specific or general reference of nouns.
Pronouns	Use pronouns correctly, avoid repetition and ambiguity, and maintain consistency with the noun they refer to.
Prepositions	Use prepositions correctly to determine the position or relationship of nouns or pronouns in a sentence.
Conjunctions	Use conjunctions correctly to connect phrases, sentences, or sentence components to maintain the logical relationship of the sentence.
Parallelism and Subordination	Use coordinate conjunctions (and, or, but, for, so, etc.) and subordinate conjunctions (because, although, if, less, etc.) correctly to construct coordinate and compound sentences.
Sentence structure	Ensure clear and concise sentences, avoid ambiguity and vague expressions.
Punctuation	Use Punctuation correctly (full stop, comma, colon, semicolon, question mark, exclamation mark, space, line break, etc.) to meet the needs of grammar rules and sentence meaning.
Subject predicate consistency	Ensure that the subject and predicate are consistent in person and number.
Adjectives and Adverbs	Use adjectives and adverbs correctly to describe the nature and state of nouns and verbs.
Antonyms and Synonyms	Use antonyms and synonyms correctly to avoid misunderstandings or ambiguity.
Inverted sentence	Use the inverted sentence structure correctly to make the sentence expression more emphasized or unique.
And then summarize, will these nongrammatical aspects cause ambiguity in understanding.	

TABLE 2. Examples of adversarial sentence: rephrased and restored. While TextFooler and TextBugger deploy word-level attacks, HotFlip combines character and word-level attacks. Note: OpenAttack toolkit standardizes text to lowercase, a factor that remains inconsequential to model classification.

Attacker	Origin Sentence	Adversarial Sentence	Restoration Sentence
TextFooler	Hopes for Eid Solution to Afghan Hostage Crisis KABUL (Reuters) - Three U.N. workers held by militants in Afghanistan were in their third week of captivity on Friday after calls from both sides for the crisis to be resolved ahead of this weekend's Muslim festival of Eid al-Fitr.	hopes for eid solvent to afghan surety crisis capital (reuters) - three uracil . newton. worker reserve by activist in afghanistan were in their third week of incarceration on friday after margin from both sides for the crisis to be resolved ahead of this weekend ' s muslim festival of eid al- fitr.	According to a Reuters report, three United Nations workers who were detained by militants in Afghanistan had been imprisoned for three weeks as of Friday. There were calls from all parties involved for the crisis to be resolved before the Muslim festival of Eid al-Fitr this weekend.
TextBugger	Cink Leads NEC Invitational by One Shot (AP) AP - Free from the burden of trying to make the Ryder Cup team, Stewart Cink looked at ease Thursday on a marathon day at the NEC Invitational that ended with his name atop the leaderboard.	cink chair nec invitational by one blastoff (ap) ap - liberal from the gist of render to pass the ryder loving squad, stewart cink reckon at relaxation th on a endurance day at the nec invitational that ended with his name atop the leaderboard.	Stewart Cink was relaxed on the final day of the NEC Invitational golf tournament, where he held the lead after shooting a low score in one round, putting him in position to make the Ryder Cup team.
HotFilp	Olympics: Hansen still strong enough to take bronze Every ounce of his energy was expended, leaving an empty fuel tank. But, even in a depleted state, Brendan Hansen found a way to bolster his ever-growing swimming legacy.	olympiad: hansen stock-still firm enough to direct bronze every ounce of his energy was expended, leaving an empty fuel tank. but,even in a depleted state , brendan hansen found a way to bolster his evergrowing swimming legacy .	Brendan Hansen remained completely focused and determined to give his all in the race. He used up every bit of energy he had, leaving him exhausted. However, even though he was drained, Brendan Hansen still managed to add to his impressive swimming career.

label the text as adversarial. At times, attackers craft perplexing structures intentionally, leading the model to register high negative log-likelihood even for standard sentences.

Therefore, a new detection approach is put forth. This approach harnesses the prompt engineering strategy of the pretrained LLM [44] to scrutinize the comprehensive fluency

TABLE 3. Summary of benchmark datasets. For SST-2, to ensure that some attack algorithms can generate 500 adversarial samples, we randomly selected 5000 data points from the training set while removing them from the training set.

Dataset	Topic	Task	Classes	Median Length
IMDB	movie reviews	sentiment classification	2	161
SST-2	movie reviews	sentiment classification	2	16
AGNews	news headline	topic classification	4	44

TABLE 4. Detection performance of DA compared to FGWS, RDE, UAPAD, and Log P.

Dataset	Attacker	FGWS		RDE		UAPAD		Log P		DA	
		AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1
IMDB	PWWS	83.6	64.3	87.5	67.5	89.2	72.3	93.5	88.2	**94.6**	**91.2**
	TextFooler	81.0	61.4	87.4	64.1	94.1	71.4	94.4	91.2	**95.5**	**93.6**
	TextBugger	75.2	58.3	83.4	59.1	93.6	68.5	95.5	92.1	**96.5**	**94.3**
	HotFilp	59.4	52.2	77.9	56.0	88.4	63.9	93.4	84.0	**94.1**	**88.9**
SST-2	PWWS	86.1	78.4	91.2	80.5	90.4	75.8	**93.1**	81.2	**93.1**	**83.2**
	TextFooler	77.2	66.7	89.0	79.2	**92.5**	78.1	92.1	84.7	91.8	**88.7**
	TextBugger	75.5	64.2	88.9	78.5	86.5	72.3	89.9	82.4	**92.0**	**85.4**
	HotFilp	67.1	65.6	87.2	71.1	85.3	69.3	**90.5**	81.3	**90.5**	**83.3**
AGNews	PWWS	82.9	63.1	86.4	64.0	90.4	71.4	94.2	83.7	**96.1**	**91.6**
	TextFooler	79.2	59.6	85.6	61.5	91.8	78.1	93.3	89.8	**95.6**	**93.7**
	TextBugger	74.4	53.9	81.2	55.1	89.6	67.1	94.8	71.3	**96.6**	**93.3**
	HotFilp	63.1	51.4	82.0	62.5	84.3	62.6	87.0	86.2	**88.2**	**91.6**

of sentences. The specific prompts formulated for this purpose are shown in Table 1.

The designed prompt is inputted into the original text $x \in X$ and the adversarial text $x' \in X'$ for fluency assessment. LLM model comprehensively includes fluency aspects of the sentences and provides an overall judgment score $S(x)$.

The prompt engineering technique capitalizes on LLM's profound linguistic comprehension and its capacity to manage vast contextual data. This method provides a holistic approach to gauging sentence fluency, offering advantages over mere negative log-likelihood comparisons by minimizing the risk of misinterpretation from isolated features.

3) SCORE CALCULATION

The process of DetectAttack is presented in Algorithm 1. After obtaining the negative log-likelihood scores for each input, the original text $x \in X$, and the attack text $x' \in X'$, these parameters are combined with the judgment score from LLM to calculate the final fluency score $g(x)$ for text x.

$$g(x) = -\log P(x) + S(x) \tag{4}$$

C. LLM RESTORE ADVERSARIAL SAMPLES

As attack algorithms advance, the text they produce increasingly mirrors genuine samples, posing challenges in model discernment and amplifying defense complexities. It is recommended to harness the capabilities of LLM [44] for pinpointing and rephrasing ungrammatical segments in sentences. LLMs, underpinned by the Transformer architecture, recognize long-range textual dependencies using the Self-Attention mechanism. With extensive parameterization and comprehensive training datasets, these models cultivate nuanced language representations, enabling proficient text understanding and generation. Throughout their pre-training phase, LLMs assimilate various linguistic dimensions—lexical, syntactic, and semantic—by engaging in tasks like tokenization and predictive modeling, while deploying optimization strategies for enhanced efficiency. Subsequently, during output formulation, techniques in NLP, such as grammar rectification and semantic coherence, are deployed to refine generated sentences, ensuring linguistic integrity aligns with task objectives. To guide this recovery process, the following prompt is provided: "Disregard the content, background, or inherent meaning of this text. The sentence may possess grammatical or punctuation inaccuracies. Grasp its essence and offer a rephrased version with improved fluency." This prompt strategy aids in effectively rephrasing and restoring adversarial attacked text, as demonstrated in Table 2.

The above statement is formalized with the concept of adversarial perturbation, where an adversarial samples for x is denoted as $(x + \xi)$, thereby $(x + \xi) \notin Y$, but the values of x

TABLE 5. Model classification accuracy after restoration using different LLM for different attack algorithms.

Dataset	LLM Model	PWWS	TextFooler	TextBugger	HotFilp
		ACC			
IMDB	Vicuna-13b	84.8	88.1	84.8	70.34
	GPT-3.5	91.5	93.5	90.5	74.5
	Claude 2	**91.8**	**94.1**	**91.8**	**89.4**
SST-2	Vicuna-13b	75.6	79.3	74.3	69.3
	GPT-3.5	91.1	86.0	87.7	74.6
	Claude 2	**92.4**	**92.1**	**93.5**	**85.2**
AGNews	Vicuna-13b	74.7	78.3	82.7	74.5
	GPT-3.5	86.8	89.5	86.4	77.8
	Claude 2	**91.9**	**92.5**	**96.6**	**88.9**

TABLE 6. Model classification success rates for rephrased adversarial samples compared to original samples by different attack algorithms.

Dataset	Attacker	Model ACC	ASR	Detect Attack Accuracy	Rephrase Text Accuracy	Attacked Model ACC	Restoration Model ACC
IMDB	PWWS	95.1	55.2	91.1	91.8	42.6	88.8
	TextFooler		68.4	92.3	94.1	30.1	89.5
	TextBugger		80.5	94.9	91.8	18.5	88.7
	HotFilp		61.2	85.9	89.4	36.9	83.9
SST-2	PWWS	92.2	52.1	82.1	92.4	44.2	83.7
	TextFooler		53.4	89.0	92.1	43.0	86.7
	TextBugger		81.6	85.6	93.5	17.0	82.3
	HotFilp		38.8	83.1	85.2	56.4	83.9
AGNews	PWWS	94.3	51.7	88.8	91.9	45.5	87.7
	TextFooler		54.6	92.8	92.5	42.8	89.7
	TextBugger		69.3	69.0	96.6	29.0	75.1
	HotFilp		38.3	74.9	88.9	58.2	83.7

and $(x + \xi)$ are very close. ξ is a perturbation generated by an attack algorithm targeting text x, introducing the concept of text perceptibility. In general, a normal human would not classify $(x+\xi)$ incorrectly, whereas the model would make a misclassification. This study performs rephrasing restore on the text to make ξ close to 0. The equation is expressed as follows:

$$F(T(x + \xi)) \approx (x) \in Y \tag{5}$$

Rephrasing of the attack text $T(x + \xi)$ is performed to ensure that it can be correctly classified by the proposed model.

IV. EXPERIMENTS

A. MODEL

In experiments addressing adversarial sample detection, this study employs a pre-trained RoBERTa-Base model from the HuggingFace Transformers library [45] as the victim classification model. This model has a 768-dimensional hidden layer size, 12 multi-headed self-attention heads, and 12 Transformer encoder layers. With an overall parameter size of 125M, vocabulary size of 50265, and support for position embeddings with a maximum of 512 positions. These parameters tuning and architectural designs result in significant performance improvements, including enhanced model capacity, better language representation learning, improved adaptability to textual data, ability to process longer sequences, and increased robustness. Using an NVIDIA 3090 GPU, after five epochs of fine-tuning with a batch size of 8 and sequence length of 512 by the Adam optimizer with a learning rate of 1e-5, this model demonstrates praiseworthy classification accuracy of over 92%.

For tasks related to attack text detection and restoration, models GPT-3.5 [46], Vicuna-13b [47], and Claude 2 [48], were deployed for text rephrasing. GPT-3.5, an enhancement by OPENAI from its predecessor GPT-3, retains its 17.5 billion parameters but outperforms due to refined techniques like knowledge distillation and model compression. While utilizing a Transformer decoder framework, GPT-3.5 has been modularized for enhanced iteration and upgrades.

TABLE 7. **Model classification success rates after restoration for ZDDR compared to BERT-Defense and TREATED.**

Dataset	Attacker	ZDDR	BERT-Defense	TREATED
		ACC		
IMDB	PWWS	**88.8**	78.5	85.6
	TextFooler	**89.5**	72.8	86.4
	TextBugger	**88.7**	74.6	77.4
	HotFilp	**83.9**	80.0	80.1
SST-2	PWWS	**83.7**	76.8	79.6
	TextFooler	**86.7**	77.1	82.2
	TextBugger	**82.3**	68.1	73.2
	HotFilp	83.9	83.6	**85.4**
AGNews	PWWS	**87.7**	82.8	83.1
	TextFooler	**89.7**	82.6	85.8
	TextBugger	**75.1**	67.9	71.0
	HotFilp	**83.7**	83.5	82.1

Anthropic's Claude 2, an evolved LLM, facilitates extended interactions up to 100k context length, prioritizing safe, controlled outputs that emulate natural, logical conversation. Lastly, Vicuna-13b, a collaborative creation by several esteemed institutions in 2023, including UC Berkeley and CMU, is fine-tuned on LLaMA proposal. With 13 billion parameters, it adopts a Sparse Transformer architecture, optimizing for computational and storage efficiency, making it apt for on-premises use. Vicuna uniquely balances efficiency and performance by integrating sparse attention mechanisms.

B. DATASET

For binary classification tasks, the defense efficacy of ZDDR was assessed using IMDB [49] movie review dataset and SST-2 [50] sentiment analysis dataset. AGNews [51] dataset served multiclass classification objectives. 5000 entries from each dataset were randomly sampled and adversarial samples were generated via specific attack algorithms.

C. METRICS

Building on prior research, four metrics were deployed to gauge the efficacy of defensive measures against adversarial samples:

Attack Success Rate (ASR): A prevalent metric that quantifies the ratio of successful adversarial samples to original ones. A higher ASR either signifies a potent attack algorithm or a less effective defense mechanism.

Area Under Receiver Operating Characteristic Curve (AUROC): This evaluates a model's prowess in detecting adversarial samples without being tethered to any fixed thresholds.

F1 Score: An amalgamation of precision and recall, it offers a holistic assessment of detection efficiency concerning adversarial samples.

Accuracy (ACC): Reflects a model's classification accuracy on the dataset, highlighting performance variations pre/post-attack and on rectified adversarial samples.

D. ATTACKER

Since sentence-level attacks do not affect the fluency of the sentence, they are outside the scope of our assumption. Therefore, we conduct experiments using character-level and word-level attacks as samples. Utilizing OpenAttack [43], a renowned open-source toolkit for textual adversarial strikes, attacks were orchestrated using prominent algorithms. These included:

PWWS [32]: A word-level attack algorithm leveraging WordNetfor synonym candidates.

TextFooler [18]: This word-level approach, akin to TextBugger, targets pivotal words in models, swapping them with synonyms until predictions shift.

HotFlip [30]: An innovative strategy for spawning adversarial samples via character substitutions, also accommodating insertions and deletions.

TextBugger [17]: A unique word-level attack methodology apt for both black-box and white-box contexts.

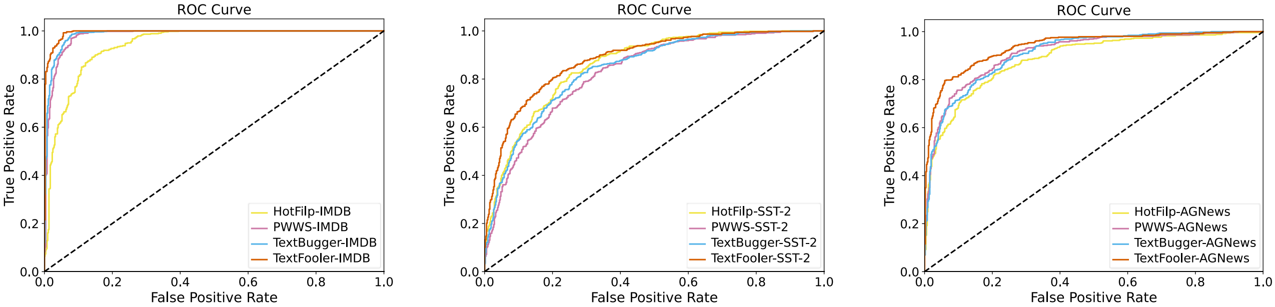
E. BASELINE DEFENSE METHODS

The proposed method was compared with four strong detection baselines:

RDE [25]: An approach grounded on feature density estimation to identify perturbations, pivoting from frequency to sentence probability density, utilizing models like BERT.

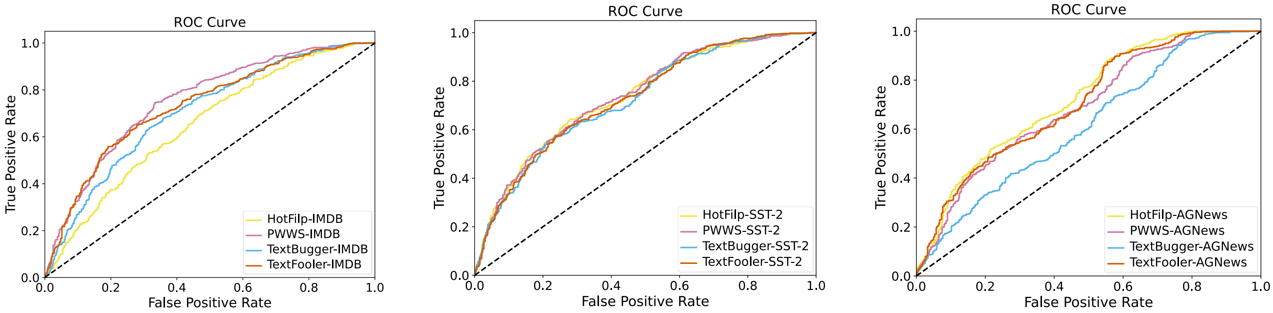
UAPAD [24]: This method gleans unique features from model outputs during the classification of both original and adversarial samples.

FGWS [39]: Recognizing word substitutions via frequency disparities between original words and substitutes, the



(b.1) IMDB **(b.2) SST 2** **(b.3) AGNews**

FIGURE 4. ROC curves for adversarial sample detection from ALBERT-Base-V2 attacks. Detection encompasses four attack algorithms across three datasets, with RoBERTa-Base as the detection model.



(c.1) IMDB **(c.2) SST 2** **(c.3) AGNews**

FIGURE 5. ROC curves for adversarial sample detection from ALBERT-Base-V2 attacks. Detection encompasses four attack algorithms across three datasets, with RoBERTa-Base as the detection mode.

method substitutes infrequent terms with more prevalent synonyms, marking a sample as adversarial if prediction shifts surpass set limits.

Furthermore, two restorations baselines were also considered:

TREATED [42]: Presents a universal defense strategy named TREATED, which leverages multiple reference models to differentiate predictions between original and adversarial data. Adversarial samples, when identified, are restricted from entering the classification model to enhance model robustness.

BERT-Defense [26]: Implements BERT tokenization to segment perturbed sentences. It employs an adapted Levenshtein distance to generate a context-independent probability distribution.

V. RESULTS

A. MAIN RESULTS

Detection Performance. Table 4 showcases detection outcomes for DA across three datasets under five attack scenarios. Optimal average metrics are emphasized in bold. Of the 12 dataset-attack pairings, DA excels in 11 regarding AUROC and all 12 in F1 score, underscoring its formidable detection prowess against diverse attack algorithms. Notably, consistent high-performance detection is observed for both IMDB and AGNews datasets, suggesting the method's insensitivity to data categories. A minor performance decrement is noted for SST-2 dataset. This can be attributed to its abbreviated average sentence length, engendering negligible fluency variances between adversarial samples produced by attack mechanisms and their native counterparts. For IMDB dataset, which boasts an average sentence span of 268, DA elevates detection precision by 3-11% relative to benchmark methods.

Adversarial Restoring. Table 5 displays the efficacy of prominent LLMs in restoring adversarial text given certain prompts. To account for LLM's inherent variability, each adversarial sample undergoes multiple restorations, with outcomes subsequently averaged. Vicuna-13b exhibits the least effective restoration, possibly due to its limited model size of 130 billion parameters. Conversely, Claude 2 stands out in restoration quality, attributed to its capacity to process and rephrase in expansive contexts up to 100k tokens. This suggests that the quality of restored text is influenced by the magnitude of LLM parameters. Thus, restorations rendered by Claude 2 are prioritized.

Table 6 presents restoration results of detected adversarial samples using LLM. "Model ACC" denotes the initial classification accuracy of the model on the dataset. "Attacked Model ACC" portrays the classification accuracy post-attack. "Detect Attack Accuracy" reveals the precision of the proposed detection method in identifying adversarial instances. "Rephrase Text Accuracy" measures the restoration method's efficacy. Adversarial instances undergo a restoration process before model classification; "Restoration Model ACC" illustrates classification accuracy post these combined operations.

Two leading restoration techniques were assessed for comparison shown in Table 7. "TREATED" a comprehensive defense strategy that utilizes various reference models to distinguish predictions between genuine and adversarial data. Once identified, adversarial samples are prevented from entering the classification model. It is worth noting that "BERT-Defense" is exclusively a restoration technique and lacks inherent adversarial detection capabilities. For this study, adversarial content detected by the proposed method was restored using "BERT-Defense".

B. CROSS-MODEL VALIDATION

DA on ALBERT. The detection efficacy of DA across various model architectures is examined. For cross-model validation, ALBERT is employed to gauge the adaptability of Detect Attack technique, discern disparities among diverse models, and ascertain its applicability across diverse scenarios.

ALBERT-Base-V2 model serves as the benchmark for Detect Attack detection strategy. Initially, ALBERT-Base-V2 Model is pretrained and subsequently fine-tuned over three cycles on an NVIDIA 3090, employing a batch size of 16 and a sequence length of 512. Adam optimizer with a learning rate of 1e-5 is utilized. Post-finetuning, a classification accuracy surpassing 92% is achieved. Consistent datasets and AUROC metric are applied. For TextFooler attack detection on IMDB, SST-2, and AGNews datasets, AUROC scores are 99.4%, 91.1%, and 94.8% respectively. These scores outstrip several algorithms outlined in Section V-A, marking a notable progression. Figure 4's AUROC curves underscore the consistent performance over varied Transformer structures, highlighting commendable adaptability. Moreover, with a capped false positive rate (FPR) at 5%, true positive rate (TPR) in detecting TextFooler attacks for IMDB, SST-2, and AGNews datasets are 96.5%, 51.4%, and 73.4% respectively. This accentuates the capability of this method to discern adversarial instances effectively, even with a 5% FPR constraint. The findings underscore the method's efficiency and robustness.

DA in Black-box Scenario. While zero-shot detection of adversarial samples experiments was executed in a white-box setting, practical application of detection methods often encounter challenges in accessing the internal weight information of the victim model, necessitating a shift to a black-box approach. This section delves into the performance in such black-box scenarios. Cross-validation was performed using RoBERTa-Base and ALBERT-Base-V2 models, wherein RoBERTa-Base model served to detect attack text generated by ALBERT-Base-V2 model. A comparison of results from Figure 5 and 4 reveals a significantly diminished area under ROC curve when deploying ALBERT-Base-V2 model for detection. For IMDB and SST-2 datasets, AUROC for HotFlip attacks witnessed a drop by 29.1% and 15.1%, respectively. Meanwhile, for AGNews dataset, AUROC for TextBugger plunged by 29.6%. To detect adversarial examples fashioned by TextFooler attacks on IMDB, SST-2, and AG-News datasets, with a stipulated 5% TPR, FPRs documented were 16.9%, 22.2%, and 15.7%, respectively. Such experimental outcomes suggest that DA predominantly aligns with white-box settings, positioning zero-shot black-box detection as a promising avenue for future research.

VI. DISCUSSION

Advancements in NLP technology, especially the rise of LLM, have made significant inroads in sectors like education, news, and arts. However, the arena faces a considerable threat from text adversarial attacks. Such attacks, where the input text is subtly altered to mislead victim models, can lead to the dissemination of false information, threaten security and privacy, undermine trust, and disrupt automated systems, thereby challenging societal trust, security, and reliability. In response, this study delves into zero-shot defenses against such adversarial threats. A notable observation was the stark contrast in fluency scores between original and adversarial text samples. The approach capitalizes on the negative log-likelihood to gauge the victim model's performance and leverages LLMs to assess sentence grammaticality. A threshold is identified within this distribution, maximizing AUROC score, to pinpoint adversarial instances. Additionally, a novel universal prompt steers LLM to restructure adversarial samples, modifying language nuances, structures, and rhetoric to maintain clarity, grammar, and original intent. Experimental evaluations underscore the method's superior efficacy against contemporary defense strategies.

A. LIMITATION

The defense strategy showcased efficacy against diverse attack methodologies across multiple datasets, however inherent constraints persist. One critical limitation of the zero-shot adversarial sample detection lies in its white-box premise, which predicates upon accessibility to the model's weight parameters. In real-world applications, this supposition might not consistently be met. As illustrated in Section V-B, utilizing RoBERTa-Base model to identify adversarial samples within ALBERT-Base-V2 considerably diminishes the potency of zero-shot detection. Future endeavors will pivot towards devising techniques to discern adversarial samples in a model-agnostic black-box milieu. Additionally, when restoring adversarial samples through rephrasing, the local deployment of the extensive language model, Vicuna-13b, faces challenges. Constrained computational resources at the local level lead to marginally suboptimal outcomes in comparison to established commercial API platforms, like GPT-3.5 or Claude 2, thereby inducing resource overheads.

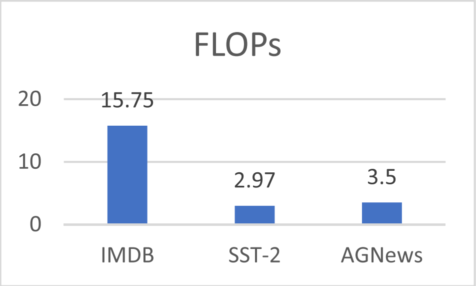


FIGURE 6. ZZDR uses victim model to calculate FLOPs of IMDB, SST-2, AGNews data sets. (Billion).

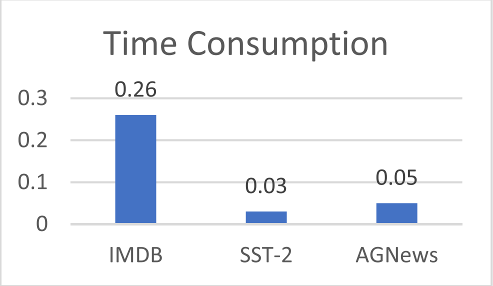


FIGURE 7. ZZDR uses the victim model to calculate the time consumption of IMDB, SST-2, and AGNews data sets. (Second).

B. RESOURCE CONSUMPTION

ZDDR uses victim model to calculate the negative log probability of text. Our ultimate goal is to transform the detection problem into a binary classification task, i.e., determine whether the text is an attack text. Therefore, different categories of data sets do not increase GPU resource consumption. As text length increases, the sequence length processed by the model also increases, which results in the model requiring more memory storage for intermediate representations and gradients. Processing long text may require more GPU memory to store intermediate activations and gradients, increasing computational complexity. As shown in the Figure 6 and 7, when using a 3090 GPU, taking the IMDB data set as an example, to process one piece of data, the number of floating point operations required is 15.75 billion, and the calculation time is 0.26 seconds. For the recovery and LLM detection steps of subsequent experiments, in order to achieve experimental results, we used the claude2 commercial API interface, which does not consume GPU resources.

C. POTENTIAL IMPACT

A potential drawback of this approach lies in its potential misuse by malicious entities. Such actors might exploit the methodology, manipulating sentence fluency to craft targeted attacks, which could proliferate misleading or harmful textual content, leading to security vulnerabilities. In view of this situation, we will implement a more stringent security and ethics review process, use generative detection technology and rule engines to achieve end-to-end abuse monitoring, comprehensively assess the risk of malicious use of the model before deployment, and introduce mechanisms Enhance the overall robustness of the system against potential threats.

VII. CONCLUSION AND FUTURE WORK

This study delves deep into adversarial attack algorithms in NLP, empirically establishing that such attacks frequently compromise sentence fluency. Stemming from this observation, the research introduces ZDDR, an innovative zero-shot unsupervised framework for the detection and restoration of adversarial samples. Unlike preceding methodologies, ZDDR operates without prior knowledge, employing negative log-likelihood to assess model capability and fluency scoring via LLM to evaluate sentence grammaticality. This culminates in determining a threshold that maximizing AUROC for adversarial sample detection. Should an input text's fluency score exceed this threshold, it is labeled as an adversarial sample. Further, the framework capitalizes on the prompt-driven capabilities of LLM. It harnesses LLM to reconstruct adversarial samples, altering expression and linguistic style, while ensuring syntactic consistency and semantic integrity post-reconstruction. Comprehensive experimental evaluations affirm the efficacy of ZDDR in countering adversarial samples.

As discussed in limitation, including the difficulty in converting detection performance under white-box premise to black-box scenarios, and the computational resource constraints of language model deployment, our follow-up research will focus on adversarial robust technology without parameters and low resources. Specifically, we plan to explore the effective fusion of large-scale language models with surrogate models or data to enable cross-model black-box adversarial example detection. In addition, we will design a parameter-free adversarial detection method that relies on the statistical characteristics of the input text, completely avoiding dependence on model parameters and structure. We hope that by combining the expressive capabilities of mature language models with the efficient implementation of other alternative technologies, the detection and recovery of adversarial samples can be transformed into practical applications, thereby improving the security and credibility of the model.

REFERENCES

[1] G. Prados Sánchez, R. Cózar-Gutiérrez, J. del Olmo-Muñoz, and J. A. González-Calero, "Impact of a gamified platform in the promotion of reading comprehension and attitudes towards reading in primary education," *Comput. Assist. Lang. Learn.*, vol. 36, no. 4, pp. 669–693, May 2023.

[2] R. Rejimoan, B. Gnanapriya, and J. Jayasudha, "A comprehensive review on deep learning approaches for question answering and machine reading comprehension in NLP," in *Proc. 2nd IEEE Delhi Sect. Flagship Conf. (DELCON)*, Feb. 2023, pp. 1–6.

[3] X. Deng and Z. Yu, "A systematic review of machine-translation-assisted language learning for sustainable education," *Sustainability*, vol. 14, no. 13, p. 7598, Jun. 2022.

[4] T. Brüggemann, U. Ludewig, R. Lorenz, and N. McElvany, "Effects of mode and medium in reading comprehension tests on cognitive load," *Comput. Educ.*, vol. 192, Jan. 2023, Art. no. 104649.

[5] B. Klimova, M. Pikhart, A. D. Benites, C. Lehr, and C. Sanchez-Stockhammer, ‘‘Neural machine translation in foreign language teaching and learning: A systematic review,’’ *Educ. Inf. Technol.*, vol. 28, no. 1, pp. 663–682, Jan. 2023.

[6] Q. Jin, Z. Yuan, G. Xiong, Q. Yu, H. Ying, C. Tan, M. Chen, S. Huang, X. Liu, and S. Yu, ‘‘Biomedical question answering: A survey of approaches and challenges,’’ *ACM Comput. Surveys*, vol. 55, no. 2, pp. 1–36, Feb. 2023.

[7] M. Zaib, W. E. Zhang, Q. Z. Sheng, A. Mahmood, and Y. Zhang, ‘‘Conversational question answering: A survey,’’ *Knowl. Inf. Syst.*, vol. 64, no. 12, pp. 3151–3195, Dec. 2022.

[8] A. Gasparetto, M. Marcuzzo, A. Zangari, and A. Albarelli, ‘‘A survey on text classification algorithms: From text to predictions,’’ *Information*, vol. 13, no. 2, p. 83, Feb. 2022.

[9] Y. Wang, C. Wang, J. Zhan, W. Ma, and Y. Jiang, ‘‘Text FCG: Fusing contextual information via graph learning for text classification,’’ *Exp. Syst. Appl.*, vol. 219, Jun. 2023, Art. no. 119658.

[10] E. León-Sandoval, M. Zareei, L. I. Barbosa-Santillán, L. E. F. Morales, A. P. Lora, and G. O. Ruiz, ‘‘Monitoring the emotional response to the COVID-19 pandemic using sentiment analysis: A case study in Mexico,’’ *Comput. Intell. Neurosci.*, vol. 2022, May 2022, Art. no. 4914665, doi: 10.1155/2022/4914665.

[11] E. León-Sandoval, M. Zareei, L. I. Barbosa-Santillán, and L. E. Falcón Morales, ‘‘Measuring the impact of language models in sentiment analysis for Mexico's COVID-19 pandemic,’’ *Electronics*, vol. 11, no. 16, p. 2483, Aug. 2022, doi: 10.3390/electronics11162483.

[12] X. Zheng, J. Zeng, Y. Zhou, C.-J. Hsieh, M. Cheng, and X. Huang, ‘‘Evaluating and enhancing the robustness of neural network-based dependency parsing models with adversarial examples,’’ in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 6600–6610.

[13] S. Qiu, Q. Liu, S. Zhou, and W. Huang, ‘‘Adversarial attack and defense technologies in natural language processing: A survey,’’ *Neurocomputing*, vol. 492, pp. 278–307, Jul. 2022.

[14] Z. Kong, J. Xue, Y. Wang, L. Huang, Z. Niu, and F. Li, ‘‘A survey on adversarial attack in the age of artificial intelligence,’’ *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–22, Jun. 2021.

[15] W. Luo, C. Wu, L. Ni, N. Zhou, and Z. Zhang, ‘‘Detecting adversarial examples by positive and negative representations,’’ *Appl. Soft Comput.*, vol. 117, Mar. 2022, Art. no. 108383.

[16] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, ‘‘Black-box generation of adversarial text sequences to evade deep learning classifiers,’’ in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2018, pp. 50–56.

[17] J. Li, S. Ji, T. Du, B. Li, and T. Wang, ‘‘TextBugger: Generating adversarial text against real-world applications,’’ 2018, *arXiv:1812.05271*.

[18] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits, ‘‘Is BERT really robust? A strong baseline for natural language attack on text classification and entailment,’’ in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 8018–8025.

[19] Z. Wang and H. Wang, ‘‘Defense of word-level adversarial attacks via random substitution encoding,’’ in *Proc. Int. Conf. Knowl. Sci., Eng. Manag.*, Hangzhou, China, Aug. 2020, pp. 312–324.

[20] Z.-H. Zhou, ‘‘A brief introduction to weakly supervised learning,’’ *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 44–53, Jan. 2018.

[21] S. Goyal, S. Doddapaneni, M. M. Khapra, and B. Ravindran, ‘‘A survey of adversarial defences and robustness in NLP,’’ 2022, *arXiv:2203.06414*.

[22] I. J. Goodfellow, J. Shlens, and C. Szegedy, ‘‘Explaining and harnessing adversarial examples,’’ 2014, *arXiv:1412.6572*.

[23] S. Gowal, K. Dvijotham, R. Stanforth, R. Bunel, C. Qin, J. Uesato, R. Arandjelovic, T. Mann, and P. Kohli, ‘‘On the effectiveness of interval bound propagation for training verifiably robust models,’’ 2018, *arXiv:1810.12715*.

[24] S. Gao, S. Dou, Q. Zhang, X. Huang, J. Ma, and Y. Shan, ‘‘On the universal adversarial perturbations for efficient data-free adversarial detection,’’ 2023, *arXiv:2306.15705*.

[25] K. Yoo, J. Kim, J. Jang, and N. Kwak, ‘‘Detection of word adversarial examples in text classification: Benchmark and baseline via robust density estimation,’’ 2022, *arXiv:2203.01677*.

[26] Y. Keller, J. Mackensen, and S. Eger, ‘‘BERT-defense: A probabilistic model based on BERT to combat cognitively inspired orthographic adversarial attacks,’’ 2021, *arXiv:2106.01452*.

[27] E. Kasneci, K. Seßler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, U. Gasser, G. Groh, S. Günnemann, E. Hüllermeier, and S. Krusche, ‘‘ChatGPT for good? On opportunities and challenges of large language models for education,’’ *Learn. Individual Differences*, vol. 103, Apr. 2023, Art. no. 102274, doi: 10.1016/j.lindif.2023.102274.

[28] J. D. Zamfirescu-Pereira, R. Y. Wong, B. Hartmann, and Q. Yang, ‘‘Why Johnny can't prompt: How non-AI experts try (and fail) to design LLM prompts,’’ in *Proc. CHI Conf. Human Factors Comput. Syst.*, Apr. 2023, pp. 1–21.

[29] Y. Chen, H. Gao, G. Cui, L. Yuan, D. Kong, H. Wu, N. Shi, B. Yuan, L. Huang, H. Xue, Z. Liu, M. Sun, and H. Ji, ‘‘From adversarial arms race to model-centric evaluation: Motivating a unified automatic robustness evaluation framework,’’ 2023, *arXiv:2305.18503*.

[30] J. Ebrahimi, A. Rao, D. Lowd, and D. Dou, ‘‘HotFlip: White-box adversarial examples for text classification,’’ 2017, *arXiv:1712.06751*.

[31] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, ‘‘BERT: Pre-training of deep bidirectional transformers for language understanding,’’ 2018, *arXiv:1810.04805*.

[32] S. Ren, Y. Deng, K. He, and W. Che, ‘‘Generating natural language adversarial examples through probability weighted word saliency,’’ in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1085–1097.

[33] A. Kumar, K. Ahuja, R. Vadapalli, and P. Talukdar, ‘‘Syntax-guided controlled generation of paraphrases,’’ *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 330–345, Dec. 2020, doi: 10.1162/tacl_a_00318.

[34] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, ‘‘Generative adversarial networks: An overview,’’ *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 53–65, Jan. 2018, doi: 10.1109/MSP.2017.2765202.

[35] J. Xu, L. Zhao, H. Yan, Q. Zeng, Y. Liang, and X. Sun, ‘‘LexicalAT: Lexical-based adversarial reinforcement training for robust sentiment classification,’’ in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 5518–5527.

[36] X. Dong, Y. Zhu, Y. Zhang, Z. Fu, and D. Xu, ‘‘Leveraging adversarial training in self-learning for cross-lingual text classification,’’ in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2020, pp. 1541–1544.

[37] R. Jia, A. Raghunathan, K. Göksel, and P. Liang, ‘‘Certified robustness to adversarial word substitutions,’’ 2019, *arXiv:1909.00986*.

[38] T. Du, S. Ji, L. Shen, J. Li, R. Beyah, and T. Wang, ‘‘Cert-RNN: Towards certifying the robustness of recurrent neural networks,’’ *CCS*, vol. 21, pp. 15–19, Nov. 2021.

[39] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, ‘‘Intriguing properties of neural networks,’’ 2013, *arXiv:1312.6199*.

[40] K. Sakaguchi, K. Duh, M. Post, and B. Van Durme, ‘‘Robsut wrod reocginiton via semi-character recurrent neural network,’’ in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 1–7.

[41] D. Pruthi, B. Dhingra, and Z. C. Lipton, ‘‘Combating adversarial misspellings with robust word recognition,’’ 2019, *arXiv:1905.11268*.

[42] B. Zhu, Z. Gu, L. Wang, and Z. Tian, ‘‘TREATED: Towards universal defense against textual adversarial attacks,’’ 2021, *arXiv:2109.06176*.

[43] S. Wu, M. Koo, L. Blum, A. Black, L. Kao, F. Scalzo, and I. Kurtz, ‘‘A comparative study of open-source large language models, GPT-4 and claude 2: Multiple-choice test taking in nephrology,’’ 2023, *arXiv:2308.04709*.

[44] I. L. Alberts, L. Mercolli, T. Pyka, K. Shi, A. Rominger, and A. Afshar-Oromieh, ‘‘Large language models (LLM) and ChatGPT: What will the impact on nuclear medicine be?’’ *Eur. J. Nucl. Med. Mol. Imag.*, vol. 50, no. 6, pp. 1549–1552, 2023.

[45] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, ‘‘RoBERTa: A robustly optimized BERT pretraining approach,’’ 2019, *arXiv:1907.11692*.

[46] A. Koubaa, ‘‘GPT-4 vs. GPT-3.5: A concise showdown,’’ Preprints.org, 2023, doi: 10.20944/preprints202303.0422.v1.

[47] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, ‘‘Judging LLM-as-a-judge with MT-bench and chatbot arena,’’ 2023, *arXiv:2306.05685*.

[48] M. Agarwal, A. Goswami, and P. Sharma, "Evaluating ChatGPT-3.5 and Claude-2 in answering and explaining conceptual medical physiology multiple-choice questions," *Cureus*, vol. 15, no. 9, 2023, doi: 10.7759/cureus.46222.

[49] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Human Lang. Technol.*, 2011, pp. 142–150.

[50] R. Socher, A. Perelygin, J. Wu, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1631–1642.

[51] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–9.

MUSHENG CHEN was born in Ganzhou, Jiangxi, China, in 1977. He received the Ph.D. degree in information system and information management from Nanchang University, Nanchang, Jiangxi, in 2018.

From 2002 to 2006, he was a Lecturer with the School of Software, Nanchang University. From 2007 to 2018, he was a Senior Engineer with the School of Software, Nanchang University. Since 2018, he has been a Lecturer and a Senior Engineer with the School of Software, Jiangxi University of Science and Technology. He is the author of two books, more than 30 articles, and more than two inventions. His research interests include natural language processing, large language models, web data mining, information security, information systems, and information management.

GUOWEI HE was born in Sanmenxia, Henan, in 1999. He received the bachelor's degree in the Internet of Things engineering from the School of Software Engineering, Jiangsu University, in 2022. He is currently pursuing the master's degree with the School of Software Engineering, Jiangxi University of Science and Technology, with a focus on artificial intelligence, natural language processing, large language models, text adversarial attacks, and information security.

JUNHUA WU was born in Fuzhou, Jiangxi, China, in 1985. She received the master's degree in software engineering from Nanchang University, Nanchang, Jiangxi, in 2010.

From 2005 to 2013, she was an Assistant Lecturer with the School of Software, Nanchang University. From 2013 to 2018, she was a Lecturer with the School of Software, Nanchang University. Since 2018, she has been a Lecturer with the School of Software, Jiangxi University of Science and Technology. She is the author of more than ten articles. Her research interests include natural language processing, large language models, web data mining, information security, text adversarial attacks, and software engineering.