Learning from Mistakes: Self-correct Adversarial Training for Chinese Unnatural Text Correction

Xuan Feng[12], **Tianlong Gu**[12*], **Xiaoli Liu**[123], **Liang Chang**[4]
[1]College of Cyber Security, Jinan University
[2]Engineering Research Center of Trustworthy AI (Ministry of Education)
[3]Graduate School of Engineering, Chiba University
[4]Guangxi Key Laboratory of Trusted Software
fenffef@163.com, {gutianlong, txlliu}@jnu.edu.cn, changl@guet.edu.cn

Abstract

Unnatural text correction aims to automatically detect and correct spelling errors or adversarial perturbation errors in sentences. Existing methods typically rely on fine-tuning or adversarial training to correct errors, which have achieved significant success. However, these methods exhibit poor generalization performance due to the difference in data distribution between training data and real-world scenarios, known as the exposure bias problem. In this paper, we propose a self-correct adversarial training framework for **L**earn**I**ng from **MI**s**T**akes (**LIMIT**), which is a task- and model-independent framework to correct unnatural errors or mistakes. Specifically, we fully utilize errors generated by the model that are actively exposed during the inference phase, i.e., predictions that are inconsistent with the target. This training method not only simulates potential errors in real application scenarios, but also mitigates the exposure bias of the traditional training process. Meanwhile, we design a novel decoding intervention strategy to maintain semantic consistency. Extensive experimental results on Chinese unnatural text error correction datasets show that our proposed method can correct multiple forms of errors and outperforms the state-of-the-art text correction methods. In addition, extensive results on Chinese and English datasets validate that LIMIT can serve as a plug-and-play defense module and can extend to new models and datasets without further training.

Introduction

Unnatural text correction (UTC) is a task that automatically corrects a variety of textual errors or mistakes in a given sentence, including spelling errors (such as visual and phonetic errors), and adversarial perturbation errors. It has attracted much attention in academia and industry due to the important role of UTC in improving text accuracy and readability (Liu, Wu, and Zhao 2024). With the widespread of unnatural texts and euphemisms on the Internet, it has become increasingly significant in various domains (Feng et al. 2024). For example, UTC can automatically fix errors in user-generated content and improve the quality of content moderation (Dai et al. 2023). Moreover, it is possible for UTC to detect and correct adversarial perturbations, enhancing the robustness and trustworthiness of the system (Liu et al. 2023). Therefore, correcting unnatural text errors or mistakes is crucial for content moderation and robust defense.

Figure 1: Examples of various unnatural text error types, the red characters are characters with errors, while the blue characters are correct characters. For easier understanding, pinyin errors in Chinese are represented by phonetic symbols in English (two → tu:).

Existing text correction methods typically rely on fine-tuning or adversarial training paradigms Liu et al. (2021); Li et al. (2022b,c,a); Wu et al. (2023b); Liu, Wu, and Zhao (2024). Although these methods have achieved significant success in common text correction tasks, they often exhibit poor generalization performance in real-world applications (Gupta et al. 2023). As shown in Figure 1, when encountering unnatural text errors, the model may be under-corrected or over-corrected. On the one hand, the under-correction is mainly due to the difference in data distribution between the training data and the real-world scenario, which is known as the exposure bias problem (Bengio et al. 2015). Specifically, training data are usually constructed or preprocessed manually with a relatively fixed and uniform distribution, while real-world data distributions are more complex and variable. In this case, the patterns and representations learned during training may not be sufficient to serve in the reasoning process, leading to unsatisfactory corrections. Thus,

*Corresponding author.

effectively solving the exposure bias problem and enhancing the model's generalization ability in real-world applications has become an important challenge. On the other hand, the model will also over-correct characters without errors in the text (Liang, Quan, and Wang 2023). In general, over-correction will distort the original meaning of the text and affect the reader's understanding. It can also affect the user's trust and experience with the error correction system. Therefore, maintaining semantic consistency while correcting text is also an important concern.

In these regards, we propose a self-correct adversarial training framework for **L**earn**I**ng from **MI**s**T**akes (**LIMIT**), which effectively copes with multi-type spelling errors and adversarial perturbations without external knowledge effectively. Specifically, we first implement a generative correction mechanism that enables models to correct multi-type errors or mistakes. As a unified mechanism, it corrects adversarial perturbations that are specific to different models and tasks. Second, we introduce self-correct adversarial training to fine-grain the contrasting examples according to the ranking loss, thereby obtaining robust representations. During the training process, incorrect examples generated based on the model's own predictions (e.g., samples inconsistent with the target generated by a beam search algorithm) are also incorporated into the learning process. This training process motivates the model to identify and correct its own biases by actively exposing its prediction errors in the inference phase. It not only mitigates the exposure bias in traditional training but also improves the robustness and reliability of the model against unnatural errors. In addition to the training phase, we also utilize semantic information in the inference process. Traditional decoding methods assign equal or probability-based weights to all candidate outputs, which leads to up-voting more erroneous answers with higher co-occurrence. To address this problem, we design a novel decoding intervention strategy to maintain semantic consistency. This helps the language model to maintain semantic consistency in decoding and thus reduces the over-correction problem.

Our main contributions are summarized as follows: (1) We implement a generative correction mechanism that enables models to correct multi-type errors. (2) We introduce self-correcting adversarial training that derives adversarial examples from the model's predictions, allowing the model to learn from its mistakes and effectively mitigate the exposure bias. (3) To address the over-correction problem of language models, we design a novel decoding intervention strategy to maintain semantic consistency. (4) Extensive experimental results on Chinese and English datasets show that our proposed method can correct multi-type errors or mistakes and can serve as a defense module in various natural language understanding and generation tasks.

Related Work

Unnatural Text Correction

Traditional Chinese text correction aims to address visual and phonetic errors caused by spelling errors. Early text correction methods adopted the process of recognizing and then correcting errors (Zhang et al. 2000). However, the effectiveness of these methods is limited by the varying accuracy of the identification and correction phases. To overcome these limitations, researchers have begun to explore end-to-end error correction methods. Wang, Tay, and Zhong (2019) utilized confusion sets and gating mechanisms, while Zhang et al. (2020a) optimized detection and correction using BERT (Jin et al. 2020). Liu et al. (2021) introduced PLOME, which leverages a pre-trained masked language model that incorporates misspelling knowledge. Li et al. (2022b,c,a) advanced text Correction techniques by learning heterogeneous knowledge from dictionaries, refining knowledge representations, and employing iterative correction strategies. Wu et al. (2023b) improved language model performance through random masking, and Liu, Wu, and Zhao (2024) rephrased sentences by filling slots.

Recently, Feng et al. (2024) extended this task to non-natural text correction to address additional challenges facing Chinese text correction, such as errors arising from perfect pinyin, abbreviation pinyin, and character split. However, perturbations in the form of insertions, deletions, inversions, and Unicode are still unexplored.

Adversarial Training

Adversarial Training (AT) (Goodfellow, Shlens, and Szegedy 2014) is a method used to improve a model's defense against adversarial perturbations by training the model with adversarial examples, thereby enhancing its robustness against deceptive inputs. Most AT methods tend to defend against suboptimal adversarial examples that deceive the decoder (Zhu et al. 2020; Jiang et al. 2020; Aghajanyan et al. 2020). More recently, Wu et al. (2023a) proposed contextualized representation adversarial training to deviate the contextual representation of the encoder from potential adversarial influences. Additionally, Gupta et al. (2023) designed a text rewriting module to eliminate perturbations in the input. Although these methods have made some progress in enhancing the model's generalization ability, they can lead to significant performance degradation on the original task. Therefore, it is crucial to mitigate the exposure bias and improve the model's adaptability and robustness to unseen scenarios.

Method

AT is usually applied to defend specific seen errors and perturbations. In contrast, real-world spelling errors and adversarial perturbations are subject to evolving model architectures and changing task contexts. To address the exposure bias problem associated with AT, as well as the inherent over-correction problem of language models, we propose a self-correct adversarial training framework for learning from mistakes (LIMIT), which consists of a generative correction mechanism, self-correct adversarial training, and a decoding intervention strategy. It corrects the text from unnatural errors through conditional generation. In this section, we illustrate the general design of the framework as well as the individual components. The overall process of our framework is shown in Figure 2.

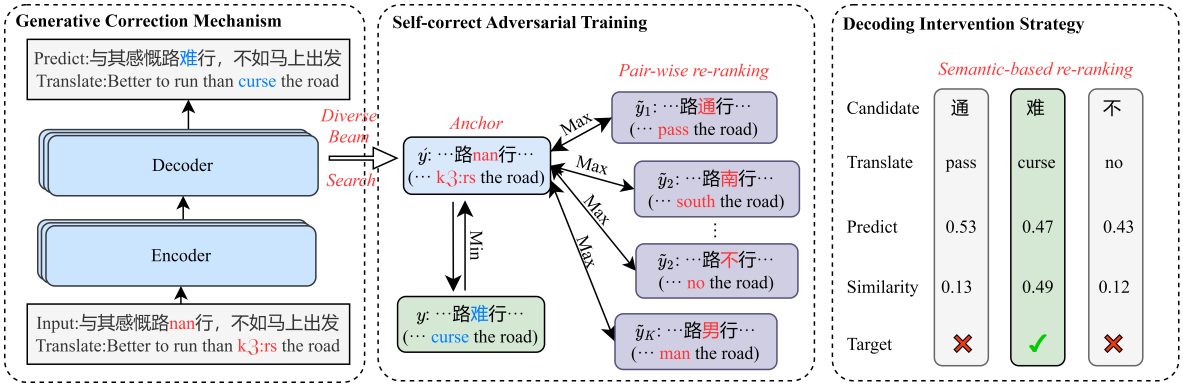

Figure 2: The overall correction process of LIMIT. For easier understanding, pinyin errors in Chinese are represented by phonetic symbols in English.

Generative Correction Mechanism

Mask-then-recovery is a commonly used correction mechanism for textual error correction. However, it fails to consider unseen multi-type errors and mistakes (such as pinyin, insertion, deletions, inversions, Unicode, character splitting, etc.) and unequal length errors (i.e., input and output lengths do not match) (Feng et al. 2024). In contrast to the mask-then-recovery process, we propose a generative correction mechanism explicitly trained for eliminating spelling errors and adversarial perturbations.

Formally, given a clean text $\mathcal{X} = \{x_1, x_2, ..., x_n\}$, a bounded, imperceptible perturbation δ is added to produce an adversarial example $\mathcal{X}' = \{x'_1, x'_2, ..., x'_m\}$. Notably, the length m of $\mathcal{X}'$ possibly differs from the original input sentence $\mathcal{X}$, i.e., the length of the input sentence $\mathcal{X}'$ is independent of the length of target sentence $\mathcal{Y}$. More specific perturbation processes can be found in the Technical Appendix. Traditional correction mechanisms can learn the optimal corrector by optimizing the following objectives:

$$\min_{f \in \mathcal{H}} \mathbb{E}_{(\mathcal{X},\mathcal{Y}) \in \mathcal{D}} \max_{|\delta| \leq \epsilon} \ell[f(\mathcal{X} + \delta), \mathcal{Y}] \tag{1}$$

Instead, our goal is to correct unnatural text errors and mistakes from the input and preserve the semantics of the original sentence. To this end, we implement a text-to-text generative mechanism, denoted by $\mathcal{P}_r$.

To effectively eliminate adversarial examples, the correct function $\mathcal{P}_r$ must recover the input $\mathcal{X}'$ to the target $\mathcal{Y} = \{y_1, y_2, ..., y_n\}$:

$$\mathcal{Y}_i = \mathcal{P}_r\left(\mathcal{X}'_i\right), \forall i \in \{1, n\} \tag{2}$$

where the perturbed characters in $\mathcal{X}'$ are replaced with the original ones to obtain $\mathcal{Y}$.

Compared to the mask-then-recovery process, the generative correction mechanism allows easier transfer to new error forms and tasks. This is a promising mechanism for correcting multi-type unnatural textual errors, in turn defending against potentially adversarial perturbations with word substitutions.

Self-correct Adversarial Training

The exposure bias problem in text correction occurs when a model is trained on a data distribution that does not accurately reflect the real-world scene. Unfortunately, real-world data distributions are more complex and variable, which may lead to poor generalization performance of the trained model. To address this problem, we introduce self-correct adversarial training. It constructs adversarial examples from its own predictions via a beam search algorithm and implements the ranking loss to help calibrate robust representations.

Following the contrastive learning framework (Chen et al. 2020), we train the model by comparing positive and negative sentence pairs to learn representations of ground truth sentences. By maximizing the similarity between source and target sequences while minimizing the similarity between negative sequences:

$$L^{\mathrm{NLL}} = -\log \frac{\exp\left(\mathrm{sim}\left(\boldsymbol{z}_{x'}, \boldsymbol{z}_y\right)/\tau\right)}{\sum \exp\left(\mathrm{sim}\left(\boldsymbol{z}_{x'}, \boldsymbol{z}_{y'}\right)/\tau\right)} \tag{3}$$

where $\boldsymbol{z}_{x'}$, $\boldsymbol{z}_y$, $\boldsymbol{z}_{y'}$ denote vector representations of input $\mathcal{X}'$, target $\mathcal{Y}$, and negative sample $\mathcal{Y}'$, respectively. τ is the temperature, and $\mathrm{sim}(\cdot,\cdot)$ defines the cosine similarity.

However, training models using naive contrastive learning frameworks typically yield error corrections. In light of this, we propose a principled method for automatically constructing adversarial negative and positive examples that allow the model to fully utilize mistakes. Specifically, we employ diverse beam search algorithms to dynamically create negative examples $\tilde{\mathcal{Y}} = \left\{\tilde{\mathcal{Y}}_1, \tilde{\mathcal{Y}}_2, \cdots, \tilde{\mathcal{Y}}_K\right\}$ from the top-K list of model predictions. These self-generated negative examples are intended to enrich the generalization capability of the model by providing more realistic test-time predictions.

We expect to fully utilize the model's mistakes, so we design a self-correcting loss function that realizes this property through pairwise comparisons. Specifically, we employ the sequence-level scores BLEU and similarity to quantify the generated examples. All examples were ranked according to their relative difference from the original sentence. Besides, the ranked example pairs are appended to the batch

to form pairwise examples $(\tilde{\mathcal{Y}}_1^+, \tilde{\mathcal{Y}}_2^-)$, where $+$ and $-$ are determined by their ranks. We optimize the model parameters using the weighted sum of the negative log-likelihood loss L^{NLL} and the self-correct ranking loss L^{RANK} as the training loss for each training pair $(\mathcal{X}', \mathcal{Y})$ as follows:

$$
\begin{aligned}
&= \\
&\sum_{a \notin \mathcal{Y}} \sum_{b \in \mathcal{Y}} \max\left(0, \gamma + \operatorname{sim}\left(z_{\mathcal{X}'}, z_{\tilde{\mathcal{Y}}_a^+}\right), \operatorname{sim}\left(z_{\mathcal{X}'}, z_{\tilde{\mathcal{Y}}_b^+}\right)\right) \\
&L = L^{\mathrm{NLL}} + L^{\mathrm{RANK}}
\end{aligned} \tag{4}
$$

During training, the L^{NLL} increases the similarity between the model output $\mathcal{X}'$ and the target sentence $\mathcal{Y}$. The L^{RANK} prevents the model from generating each counter-example containing an adversarial perturbation $\tilde{\mathcal{Y}}_k$, γ is the margin.

Decoding Intervention Strategy

The semantics of discrete text may be affected by even subtle errors and perturbations. Traditional decoding strategies may lead to a dramatic performance degradation under adversarial perturbations. Therefore, we design a decoding intervention strategy to address the over-correction problem and further improve the model's robustness. Specifically, we incorporate a similarity function into the decoding phase to dynamically evaluate the correctness of the next token predicted by the decoder. The decoding goal in LIMIT is to find the sequence $\mathcal{Y}$ that maximizes the likelihood of the learned similarity score and the regular language model:

$$
\begin{aligned}
s(\mathcal{X}', \mathcal{Y}) = \sum_{t=1}^{|\mathcal{Y}|} &\left(\log p_\theta\left(\mathcal{Y}_t \mid \mathcal{Y}_{<t}, \mathcal{X}'\right)\right. \\
&\left. + \alpha \times \operatorname{sim}\left(\mathcal{Y}_t, \mathcal{X}'\right)\right)
\end{aligned} \tag{5}
$$

where the first term is the original probability of the language model, and the second term is the similarity score between the given the input sentence $\mathcal{X}'$ and the generated $\mathcal{Y}_t$, and α is the hyper-parameter that balances the contribution of each term.

Experiments

In this section, we compare LIMIT with a range of text correction methods on unnatural text correction datasets. We also evaluate the adoption of LIMIT as a defense method against perturbations on natural language generation (NLG) tasks and natural language understanding (NLU) compared to adversarial training methods.

Datasets

Unnatural Text Correction Datasets: *PROTECT* (Feng et al. 2024) includes unnatural text errors that are possible in Chinese characters. There are four subdatasets *Perfect Pinyin*, *Abbreviation Pinyin*, *Character Split*, and *Hybrid*. It covers common spelling errors involving visually or phonetically similar characters, splitting characters into radicals, and converting characters to perfect or abbreviated pinyin forms. *Hybrid-v2* Based on the Hybrid perturbations, we construct insertion, deletion, inversion, and Unicode perturbations.

To verify that the proposed method can effectively serve as an adversarial defense method, following (Su et al. 2022) and (Feng et al. 2024), we perturb the NLU and NLG datasets. The specific perturbation process is detailed in the Technical Appendix.

NLU Datasets: For Chinese datasets, *TNEWS* (Xu et al. 2020) is a Chinese dataset for text classification. *AFQMC* (Xu et al. 2020) is a Chinese question-matching dataset designed to evaluate the performance of natural language processing models. *CMNLI* (Xu et al. 2020) is a Chinese multi-genre cross-domain natural language reasoning dataset that asses a model's ability to determine the relationships between premises and hypotheses. *IFLYTEK* (Xu et al. 2020) is a Chinese long-text classification dataset. *COLD* (Deng et al. 2022) is a Chinese offensive speech detection dataset.

For English datasets, we conduct our experiments on *advSST-2*, *advQQP*, *advMNLI*, and *advRTE* (Wang et al. 2021), which applies 14 state-of-the-art textual adversarial attack methods to GLUE tasks.

NLG Datasets: *ADGEN* (Shao et al. 2019) is an advertisement generation dataset. *CSL* (Zhang, Li, and Li 2021) is an academic domain text summarization dataset consisting of abstracts and titles of publications in the field of computer science. *LCSTS* (Hu, Chen, and Zhu 2015) is a large Chinese short text summarization dataset.

Baselines

Text Correct Baselines: *BERT* (Devlin et al. 2019) is a pre-trained language model that can be used for fine-tuning various natural language processing tasks. *SoftMasked* (Zhang et al. 2020b) used a pipeline structure of detection network and correction network to implement text error correction. *MDCSpell* (Zhu et al. 2022) employed a late fusion strategy to integrate the hidden states of the corrector with those of the detector, aiming to mitigate the adverse effects caused by misspelled characters. *PLOME* (Liu et al. 2021) designed a masking strategy based on a semantic confusion set when training pre-trained language models. *MFT* (Wu et al. 2023b) randomly masked 20% of the non-error tokens in the input sequence during the fine-tuning process, which is enough to learn a better language model without sacrificing the error model. *RobustGEC* (Zhang et al. 2023) proposed an effective post-training method Context Perturbation Robustness to enhance the stability and reliability of these systems in real-world applications. *ATINTER* (Gupta et al. 2023) is a module that intercepts and learns to rewrite adversarial inputs, making it non-adversarial for downstream text classifiers. *ReLM* (Liu, Wu, and Zhao 2024) trained the model to restate entire sentences by filling in extra slots instead of marking them word by word.

Adversarial Training Baselines: *FreeLB* (Zhu et al. 2020) is a fast adversarial training algorithm that integrates each intermediate example into a backward pass. *SMART* (Jiang et al. 2020) introduced smoothness-induced regularization in adversarial training for better generalization performance. *R3F* (Aghajanyan et al. 2020) replaced the previously used adversarial targets with parametric noise (sam-

Model	Perfect Pinyin		Abb. Pinyin		Char. Split		Hybrid		Hybrid-v2	
	Pre	F1	Pre	F1	Pre	F1	Pre	F1	Pre	F1
BERT	31.0	33.0	41.0	43.0	43.4	54.4	25.7	28.3	14.6	15.8
SoftMasked	32.0	34.5	45.7	46.8	43.4	50.6	22.0	25.8	15.3	16.7
MDCSpell	32.6	34.6	45.5	46.5	42.3	49.8	23.1	27.0	14.3	15.5
PLOME	59.5	59.7	33.5	35.1	2.9	3.2	48.9	48.8	-	-
MFT	30.8	32.8	30.2	31.7	39.6	46.8	48.0	56.2	-	-
ReLM	-	-	46.7	47.9	-	-	37.0	43.5	-	-
RobustGEC	58.6	53.2	46.9	30.2	67.7	65.5	51.7	38.5	16.4	15.4
ATINTER	70.0	61.2	58.0	35.6	82.0	78.7	59.1	41.3	54.1	23.5
PROTECT-Fewshot	<u>73.4</u>	67.0	68.7	45.4	81.4	78.7	66.8	47.4	77.1	50.3
PROTECT-Finetune	**90.2**	<u>82.1</u>	**84.8**	<u>57.7</u>	**94.4**	<u>91.8</u>	<u>90.4</u>	<u>71.2</u>	<u>83.1</u>	<u>59.6</u>
LIMIT(Ours)	**90.2**†	**84.6**†	<u>69.8</u>	**63.5**†	<u>91.9</u>	**93.2**†	**91.6**†	**81.2**†	**84.8**†	**66.8**†
GPT-3.5-Turbo-10shot	23.2	22.1	2.7	3.4	1.0	1.2	22.2	19.4	12.5	11.0

Table 1: Performance of the baseline model and our approach on five Chinese unnatural text correction datasets. The best and second-best results are highlighted in **bold** and <u>underline</u>. Where Abb. Pinyin and Char. Split represents the Abbreviation Pinyin and Character Split respectively. The superscript † indicates $p < 0.05$ for the t-test of the LIMIT vs. the PROTECT-Finetune.

pled from a normal or uniform distribution). *CreAT* (Wu et al. 2023a) presented a simple and effective contextual representation-adversarial training, where the attack is to explicitly optimize the contextual representation of the deviation encoder. *Match-Tuning* (Tong et al. 2022) added regularization between examples in the same batch.

Large Language Models: *Llama*[1], *Baichuan2* (Baichuan 2023), *OPT-66B* (Zhang et al. 2022), *BLOOM* (Le Scao et al. 2023) and *ChatGPT*. More experimental results for large language models are presented in the Technical Appendix.

Implementations

To obtain robust textual representations against unnatural textual errors. For the Chinese corpus, we constructed adversarial examples using 300k randomly extracted texts from Chinese Wikipedia and continued pre-training on the T5-Base-Chinese [2] model. Similarly, for the English corpus, we constructed adversarial examples using 300k randomly extracted texts from Comments2019. Likewise, we continued pre-training on the T5-Large[3] model.

To obtain robust representations, we pretrained the generation model after constructing adversarial examples. LIMIT has 12 layers/heads and 768 hidden neurons. It undergoes training on a scale of 60k with a batch size of 32, a learning rate of 1e-5, and a warm-up stage of 6k. The English version consists of 48 layers, 24 attention heads, and 1024 hidden neurons. It follows a learning rate of 1e-5, a warm-up stage of 6k, a batch size of 32, and a training stage of 60k.

LIMIT introduces three additional hyperparameters. The first one is the diversity of beam search size, denoted as K. The second one is the boundary strength, denoted as γ. The third one is the balancing factor, denoted as α. For all datasets, we set K to 12 and γ to 0.01. We tune α on the validation set using values from [0.3, 0.4, 0.5, 0.6, 0.7]. In practice, increasing the number of dynamic negative samples continually improves performance.

For the unnatural text correction task, we evaluate performance using precision (Pre) and the F1 score. In the NLU task, accuracy (Acc) serves as our primary metric. For NLG tasks, we employ Rouge-1 (R-1), Rouge-2 (R-2), and Rouge-L (R-L) to assess the quality of the generated text in comparison to the target text. These three metrics provide different perspectives on the quality of the generated text.

Results on Chinese Unnatural Text Correction Datasets

In the unnatural text correction task, we evaluate the performance of several baseline models and our proposed LIMIT in five different types of unnatural text correction tasks: perfect pinyin, abbreviation pinyin, character split, hybrid, and hybrid-v2. The experimental results are shown in Table 1. To guarantee the reliability of the experiments, all results are averaged over five experiments.

We analyze the performance of traditional Bert-based text correction methods. PLOME performed best in the perfect pinyin task with an F1 score of 59.7%, but worst in the character split task with an F1 score of only 3.2%. ReLM performed well in the abbreviation pinyin task with an F1 score of 47.9% but failed to correct the errors in the other tasks. Overall, traditional text correction methods perform poorly in unnatural text correction tasks with generally low F1 scores. For the harder Hybrid-v2 dataset, PLOME, MFT, and ReLM cannot handle such errors.

Furthermore, we analyze the performance of the generative correction methods. RobustGEC is the first to consider the robustness of a text error correction task against perturbations, however, the method performs poorly on the unnatural text correction tasks. The text rewriting strategy adopted by ATINTER performed well in the perfect pinyin and Character split tasks, with F1 scores of 61.2% and 78.7%, re-

[1]https://github.com/LlamaFamily/Llama-Chinese

[2]https://huggingface.co/uer/t5-base-chinese-cluecorpussmall

[3]https://huggingface.co/google-t5/t5-large

Model	TNEWS		AFQMC		CMNLI		IFLYTEK		COLD	
	Clean (Acc)	Adv (Acc)	Clean (Acc)	Adv (Acc)	Clean (Acc)	Adv (Acc)	Clean (Acc)	Adv (Acc)	Clean (Acc)	Adv (Acc)
BERT	66.6	65.4	75.1	72.4	80.8	77.5	58.4	56.2	93.1	80.5
FreeLB	67.1	65.5	74.2	70.9	80.1	77.4	59.3	57.6	93.1	80.5
SMART	66.6	64.7	73.1	70.9	79.4	76.3	58.3	55.5	93.1	80.5
R3F	67.1	65.5	74.1	71.0	80.1	77.5	58.7	56.5	93.1	80.5
CreAT	66.8	65.4	73.4	70.5	79.0	76.0	58.9	57.2	93.1	80.5
BERT+LIMIT(Ours)	66.6	**66.0**	75.1	**72.4**	80.8	**79.1**	58.4	**59.7**	93.1	**82.4**
Llama-7B	13.0	10.7	43.5	49.1	34.9	34.9	47.6	48.7	50.0	43.8
Baichuan2-13B	33.2	27.9	69.0	69.0	34.4	33.5	44.8	44.0	48.2	47.5
GPT-3.5-Turbo-10shot	49.9	47.9	69.0	68.9	52.9	51.0	49.8	37.1	51.9	50.0

Table 2: Performance of the adversarial training baseline models and our method on the Chinese NLU dataset. The best results are labeled with **bold**.

Model	Adv SST-2 (Acc)	Adv QQP (Acc)	Adv MNLI-m (Acc)	Adv RTE (Acc)
BERT ♮	32.3	50.8	32.6	37.0
FreeLB ♮	31.6	51.0	33.5	42.0
R3F ♮	38.5	40.6	35.8	50.1
CreAT ♮	35.3	51.5	36.0	45.2
Match-Tuning ♮	51.4	41.5	35.5	47.5
BERT+LIMIT(Ours)	**66.2**	**78.8**	**69.4**	**84.0**
OPT-66B ♭	52.4	46.1	39.7	42.0
BLOOM-176B ♭	51.3	41.0	26.4	43.2
GPT-3.5-Turbo-10shot	60.1	72.0	67.8	75.3

Table 3: Performance of the adversarial training baseline models and our method on the English AdvGLUE dataset. Partial experimental results are from (Wu et al. 2023a)♮ and (Wang et al. 2023)♭, with the best performing scores shown in **bold**. More results for the large language model are presented in the Technical Appendix.

spectively, but did not perform well in the more complex tasks. The PROTECT-Fewshot model performed well on the perfect pinyin, abbreviation pinyin, and character split tasks with F1 scores of 67.0%, 45.4%, and 78.7% respectively, but performed slightly poorer on the hybrid task with F1 scores of 61.2% and 78.7% respectively. This is because the method excels at predicting error accuracy but is less effective at correcting errors.

It is noteworthy that our proposed LIMIT model demonstrates exceptional performance across all tasks, while the large language model is ineffective at correcting errors in unnatural text. This suggests that the LIMIT has significant advantages and potential in the Chinese unnatural text correction task.

Results on Chinese NLU Datasets

Table 2 shows the experimental results on five Chinese NLU datasets. The experimental demonstrates show that the adversarial robustness of our proposed LIMIT achieves consistent improvement on the NLU task. On the perturbed TNEWS, AFQMC, CMNLI, IFLYTEK, and COLD datasets, LIMIT obtains an improvement of 0.6%, 1.9%, 3.1%, 2.5%, and 1.9%, respectively. We find that all the adversarial training methods suffer a loss of performance in Chinese tasks. The trade-off between performance and robustness is consistent with previous findings. For LIMIT, however, it is only responsible for removing adversarial perturbations from the input text. This preserves the performance of the language model to some extent. For example, the FreeLB and R3F outperform vanilla BERT on clean TNEWS and IFLYTEK datasets, while the SMART and CreAT sacrifice prediction performance on all tasks. On the relatively easier classification adversarial datasets TNEWS and IFLYTEK, most of the methods provide performance gains. However, for the inference tasks AFQMC and CMNLI, both lead to performance loss when trading off performance and robustness.

Results on English NLU Datasets

Table 3 shows the experimental results on four English NLU adversarial datasets (AdvGLUE). Experimental results illustrate that LIMIT outperforms state-of-the-art methods and achieves the best performance on four randomly selected datasets. For the pre-training and fine-tuning methods, Match-Tuning with BERT-large achieves competitive results by adding regularization. For the large language models, ChatGPT exhibits better performance than the specifically designed model, achieving accuracy scores of 60.1%, 72.0%, 67.8%, and 65.5% on the four datasets, respectively. However, models with the same parameter sizes show considerable variation in performance, with an average accuracy of only 42.2% for BLOOM. For adversarial training methods, FreeLB, R3F, and CreAT perform poorly, which has validated their struggles to cope with a multitype of errors and perturbations.

Results on Chinese NLG Datasets

The experimental results on the Chinese NLG dataset are demonstrated in Table 4. The results show that LIMIT exhibits the best adversarial robustness. This reflects its transferability to new tasks and models with competitive per-

Model	Clean (R-1)	Clean (R-2)	Clean (R-L)	Adv (R-1)	Adv (R-2)	Adv (R-L)
ADGEN						
RobustGEC	43.9	18.9	26.8	40.6	15.6	23.9
PROTECT	42.7	18.9	27.3	39.0	15.4	24.1
ATINTER	43.9	18.9	26.8	37.8	14.1	23.6
LIMIT(Ours)	43.9	18.9	26.8	**41.6**	**16.7**	**24.9**
CSL						
RobustGEC	64.6	52.6	61.4	52.9	38.2	49.9
PROTECT	63.6	52.0	60.7	52.8	37.7	49.0
ATINTER	64.6	52.6	61.4	48.8	35.3	46.1
LIMIT(Ours)	64.6	52.6	61.4	**58.9**	**45.3**	**55.5**
LCSTS						
RobustGEC	44.0	29.3	40.7	35.0	21.0	32.4
PROTECT	42.0	27.4	38.9	35.1	21.1	32.6
ATINTER	44.0	29.3	40.7	39.3	24.5	36.0
LIMIT(Ours)	44.0	29.3	40.7	**39.4**	**24.6**	**36.1**

Table 4: Performance of the generative correction baseline models and our method on the NLG dataset. The best results are labeled with **bold**.

Dataset / Model	Hybrid (F1)	CMNLI (Acc)	CSL (R-1)	CSL (R-2)	CSL (R-L)
Fine-tuning	75.2	77.5	52.8	38.2	49.9
+SC	75.6	78.8	58.0	45.1	55.4
+DI	**81.2**	**79.1**	**58.9**	**45.3**	**55.5**

Table 5: Ablation results in different components of LIMIT. The best-performing scores are in bold. Results for additional datasets are provided in the Technical Appendix.

formance. For the experimental results with BobustGEC as the backbone, there is an average improvement of 1.0%, 6.2%, and 3.9% for ADGEN, CSL, and LCSTS adversarial datasets at Rouge-1, Rouge-2, and Rouge-L, respectively. Although PROTECT is designed to correct unnatural errors, it performs poorly on all three adversarial perturbed datasets. While the ATINTER, which employs rewriting to mitigate adversarial perturbations, incurs a performance loss on the ADGEN and CSL adversarial datasets. Specifically, on the ADGEN adversarial dataset, it degrades by 2.8%, 1.5%, and 0.3% on Rouge-1, Rouge-2, and Rouge-L, respectively. Likewise, it drops by 4.1%, 2.9%, and 3.8% in the CSL dataset, respectively.

Ablation Study

Table 5 shows the ablation studies of the different components of LIMIT on the Chinese and English adversarial datasets. It indicates that the components of self-correct adversarial training (SC) and decoding intervention (DI) both play key roles in enhancing adversarial robustness. Specifically, with the addition of SC, the average accuracy of the NLU dataset is improved by an average of 2.2%, and the Rouge-L of the NLG dataset is improved by an average of 3.1%. Similarly, with the addition of DI, the average accuracy of the NLU dataset is improved by an average of 0.4%, and an average of 0.3% improves the Rouge-L of the NLG dataset.

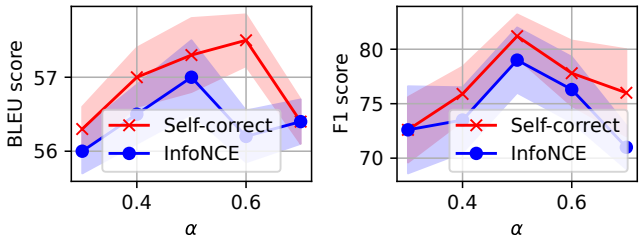

Figure 3: Relationship between α and BLEU under different training losses on the Chinese unnatural text correction dataset (Hybrid).

Task	Pinyin	Abb.	Char.	Hybrid
	#Overcorrections / #Undercorrections			
Vanilla	151/636	107/428	131/286	124/522
PROTECT	92/423	**22**/319	26/72	48/310
LIMIT(Ours)	**43/149**	31/**310**	**24/67**	**33/134**

Table 6: The empirical analysis results of the Vanilla finetune, PROTECT, and the proposed method in this paper. Where Abb. and Char. are Abbreviation Pinyin and Character Split respectively. Following Feng et al. (2024), we statistically counted the quantity of overcorrected samples.

Empirical Analysis on Hyper-parameter

Figure 3 shows the impact of the parameter α on BLEU scores and accuracy under different training losses. The proposed self-correct ranking loss achieves the best performance at $\alpha = 0.5$, with a BLEU score of 0.57 and an F1 score of 81.2%. In comparison, the traditional adversarial training loss, InfoNCE, reaches a BLEU score of 0.56 and an accuracy of 79.3% at $\alpha = 0.5$. It demonstrated the effectiveness of the self-correct ranking loss in Chinese unnatural text correction.

Empirical Analysis on Over-correction

LIMIT achieves the best performance in perfect pinyin, character split, and hybrid, as shown in Table 6. Nevertheless, improving correction accuracy for abbreviated pinyin remains a necessary direction that requires further effort.

Conclusion

In this paper, we propose a self-correct adversarial training framework for learning from mistakes, LIMIT, that enhances model robustness and adaptability to evolving spelling errors and adversarial perturbations. LIMIT offers a model- and task-agnostic solution for correcting unnatural text errors, ensuring robustness in error correction. Furthermore, it showcases transferability to various natural language understanding and natural language generation tasks, effectively resisting multi-type errors and perturbations.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. U22A2099 and Grant No. 62336003).

References

Aghajanyan, A.; Shrivastava, A.; Gupta, A.; Goyal, N.; Zettlemoyer, L.; and Gupta, S. 2020. Better Fine-Tuning by Reducing Representational Collapse. In *International Conference on Learning Representations*.

Baichuan. 2023. Baichuan 2: Open Large-scale Language Models. *arXiv preprint arXiv:2309.10305*.

Bengio, S.; Vinyals, O.; Jaitly, N.; and Shazeer, N. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A Simple Framework for Contrastive Learning of Visual Representations. *arXiv preprint arXiv:2002.05709*.

Dai, S.; Mahloujifar, S.; Xiang, C.; Sehwag, V.; Chen, P.-Y.; and Mittal, P. 2023. MultiRobustBench: Benchmarking Robustness Against Multiple Attacks. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 6760–6785. PMLR.

Deng, J.; Zhou, J.; Sun, H.; Zheng, C.; Mi, F.; Meng, H.; and Huang, M. 2022. COLD: A Benchmark for Chinese Offensive Language Detection. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 11580–11599. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *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)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Feng, X.; Gu, T.; Chang, L.; and Liu, X. 2024. PROTECT: Parameter-Efficient Tuning for Few-Shot Robust Chinese Text Correction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32: 3270–3282.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Gupta, A.; Blum, C.; Choji, T.; Fei, Y.; Shah, S.; Vempala, A.; and Srikumar, V. 2023. Don't Retrain, Just Rewrite: Countering Adversarial Perturbations by Rewriting Text. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 13981–13998. Toronto, Canada: Association for Computational Linguistics.

Hu, B.; Chen, Q.; and Zhu, F. 2015. LCSTS: A Large Scale Chinese Short Text Summarization Dataset. In Màrquez, L.; Callison-Burch, C.; and Su, J., eds., *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1967–1972. Lisbon, Portugal: Association for Computational Linguistics.

Jiang, H.; He, P.; Chen, W.; Liu, X.; Gao, J.; and Zhao, T. 2020. SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2177–2190.

Jin, D.; Jin, Z.; Zhou, J. T.; and Szolovits, P. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 8018–8025.

Le Scao, T.; Fan, A.; Akiki, C.; Pavlick, E.; Ilić, S.; Hesslow, D.; Castagné, R.; Luccioni, A. S.; Yvon, F.; Gallé, M.; et al. 2023. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Li, J.; Wang, Q.; Mao, Z.; Guo, J.; Yang, Y.; and Zhang, Y. 2022a. Improving Chinese Spelling Check by Character Pronunciation Prediction: The Effects of Adaptivity and Granularity. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 4275–4286. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Li, Y.; Ma, S.; Zhou, Q.; Li, Z.; Yangning, L.; Huang, S.; Liu, R.; Li, C.; Cao, Y.; and Zheng, H. 2022b. Learning from the Dictionary: Heterogeneous Knowledge Guided Fine-tuning for Chinese Spell Checking. In Goldberg, Y.; Kozareva, Z.; and Zhang, Y., eds., *Findings of the Association for Computational Linguistics: EMNLP 2022*, 238–249. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics.

Li, Y.; Zhou, Q.; Li, Y.; Li, Z.; Liu, R.; Sun, R.; Wang, Z.; Li, C.; Cao, Y.; and Zheng, H.-T. 2022c. The Past Mistake is the Future Wisdom: Error-driven Contrastive Probability Optimization for Chinese Spell Checking. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Findings of the Association for Computational Linguistics: ACL 2022*, 3202–3213. Dublin, Ireland: Association for Computational Linguistics.

Liang, Z.; Quan, X.; and Wang, Q. 2023. Disentangled Phonetic Representation for Chinese Spelling Correction. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 13509–13521. Toronto, Canada: Association for Computational Linguistics.

Liu, L.; Wu, H.; and Zhao, H. 2024. Chinese Spelling Correction as Rephrasing Language Model. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17): 18662–18670.

Liu, S.; Yang, T.; Yue, T.; Zhang, F.; and Wang, D. 2021. PLOME: Pre-training with misspelled knowledge for Chinese spelling correction. 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)*, 2991–3000.

Liu, Z.; Xu, Y.; Ji, X.; and Chan, A. B. 2023. TWINS: A Fine-Tuning Framework for Improved Transferability of Adversarial Robustness and Generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16436–16446.

Shao, Z.; Huang, M.; Wen, J.; Xu, W.; and Zhu, X. 2019. Long and Diverse Text Generation with Planning-based Hierarchical Variational Model. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3257–3268. Hong Kong, China: Association for Computational Linguistics.

Su, H.; Shi, W.; Shen, X.; Xiao, Z.; Ji, T.; Fang, J.; and Zhou, J. 2022. RoCBert: Robust Chinese Bert with Multimodal Contrastive Pretraining. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 921–931.

Tong, S.; Dong, Q.; Dai, D.; Song, Y.; Liu, T.; Chang, B.; and Sui, Z. 2022. Robust Fine-tuning via Perturbation and Interpolation from In-batch Instances. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 4397–4403. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Wang, B.; Xu, C.; Wang, S.; Gan, Z.; Cheng, Y.; Gao, J.; Awadallah, A. H.; and Li, B. 2021. Adversarial GLUE: A Multi-Task Benchmark for Robustness Evaluation of Language Models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Wang, D.; Tay, Y.; and Zhong, L. 2019. Confusionset-guided pointer networks for Chinese spelling check. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5780–5785.

Wang, J.; Xixu, H.; Hou, W.; Chen, H.; Zheng, R.; Wang, Y.; Yang, L.; Ye, W.; Huang, H.; Geng, X.; et al. 2023. On the Robustness of ChatGPT: An Adversarial and Out-of-distribution Perspective. In *ICLR 2023 Workshop on Trustworthy and Reliable Large-Scale Machine Learning Models*.

Wu, H.; Liu, Y.; Shi, H.; hai zhao; and Zhang, M. 2023a. Toward Adversarial Training on Contextualized Language Representation. In *The Eleventh International Conference on Learning Representations*.

Wu, H.; Zhang, S.; Zhang, Y.; and Zhao, H. 2023b. Rethinking Masked Language Modeling for Chinese Spelling Correction. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 10743–10756. Toronto, Canada: Association for Computational Linguistics.

Xu, L.; Hu, H.; Zhang, X.; Li, L.; Cao, C.; Li, Y.; Xu, Y.; Sun, K.; Yu, D.; Yu, C.; Tian, Y.; Dong, Q.; Liu, W.; Shi, B.; Cui, Y.; Li, J.; Zeng, J.; Wang, R.; Xie, W.; Li, Y.; Patterson, Y.; Tian, Z.; Zhang, Y.; Zhou, H.; Liu, S.; Zhao, Z.; Zhao, Q.; Yue, C.; Zhang, X.; Yang, Z.; Richardson, K.; and Lan, Z. 2020. CLUE: A Chinese Language Understanding Evaluation Benchmark. In Scott, D.; Bel, N.; and Zong, C., eds., *Proceedings of the 28th International Conference on Computational Linguistics*, 4762–4772. Barcelona, Spain (Online): International Committee on Computational Linguistics.

Zhang, L.; Zhou, M.; Huang, C.; and Pan, H. 2000. Automatic detecting/correcting errors in Chinese text by an approximate word-matching algorithm. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, 248–254.

Zhang, S.; Huang, H.; Liu, J.; and Li, H. 2020a. Spelling Error Correction with Soft-Masked BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 882–890.

Zhang, S.; Huang, H.; Liu, J.; and Li, H. 2020b. Spelling Error Correction with Soft-Masked BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 882–890. Online: Association for Computational Linguistics.

Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X. V.; et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Zhang, X.; Li, P.; and Li, H. 2021. AMBERT: A Pre-trained Language Model with Multi-Grained Tokenization. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 421–435. Online: Association for Computational Linguistics.

Zhang, Y.; Cui, L.; Zhao, E.; Bi, W.; and Shi, S. 2023. RobustGEC: Robust Grammatical Error Correction Against Subtle Context Perturbation. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 16780–16793. Singapore: Association for Computational Linguistics.

Zhu, C.; Cheng, Y.; Gan, Z.; Sun, S.; Goldstein, T.; and Liu, J. 2020. FreeLB: Enhanced Adversarial Training for Natural Language Understanding. In *International Conference on Learning Representations*.

Zhu, C.; Ying, Z.; Zhang, B.; and Mao, F. 2022. MDCSpell: A Multi-task Detector-Corrector Framework for Chinese Spelling Correction. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Findings of the Association for Computational Linguistics: ACL 2022*, 1244–1253. Dublin, Ireland: Association for Computational Linguistics.