TaiChi: Improving the Robustness of NLP Models by Seeking Common Ground While Reserving Differences

Huimin Chen[†], **Chengyu Wang**[‡], **Yanhao Wang**[†], **Cen Chen**[†], **Yinggui Wang**[§]
[†]School of Data Science and Engineering, East China Normal University, Shanghai, China
[‡]Alibaba Group, Hangzhou, China
[§]Ant Group, Hangzhou, China
saichen@stu.ecnu.edu.cn, chengyu.wcy@alibaba-inc.com,
{yhwang,cenchen}@dase.ecnu.edu.cn, wyinggui@gmail.com

Abstract

Recent studies have shown that Pre-trained Language Models (PLMs) are vulnerable to adversarial examples, crafted by introducing human-imperceptible perturbations to clean examples to deceive the models. This vulnerability stems from the divergence in the data distributions of clean and adversarial examples. Therefore, addressing this issue involves teaching the model to diminish the differences between the two types of samples and to focus more on their similarities. To this end, we propose a novel approach named *TaiChi* that employs a Siamese network architecture. Specifically, it consists of two sub-networks sharing the same structure but trained on clean and adversarial samples, respectively, and uses a contrastive learning strategy to encourage the generation of similar language representations for both kinds of samples. Furthermore, it utilizes the Kullback-Leibler (KL) divergence loss to enhance the consistency in the predictive behavior of the two sub-networks. Extensive experiments across three widely used datasets demonstrate that *TaiChi* achieves superior trade-offs between robustness to adversarial attacks at token and character levels and accuracy on clean examples compared to previous defense methods. Our code and data are publicly available at https://github.com/sai4july/TaiChi.

Keywords: text classification, neural language representation learning, text analytics, model robustness

1. Introduction

Pre-trained Language Models (PLMs), such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), have achieved prominent performance in many natural language processing (NLP) applications due to their strong ability to learn text representations. However, they are well known to be vulnerable to adversarial examples (Gao et al., 2018; Li et al., 2019; Ren et al., 2019; Jin et al., 2020) that are delicately crafted from (original) clean examples with human-tolerable spelling, syntactic, or grammatical mistakes. For example, as illustrated in Table 1, a BERT-based sentiment classification model is easily misled by an adversarial sample that simply changes one token "*perfect*" in a clean sample to its synonym "*spotless*". Such attacks put the security of PLM-based text classification models at risk.

Sentence	Label	Predict
perfect performance by xxx	Positive	Positive
spotless performance by xxx	Positive	Negative

Table 1: Adversarial example generated by PWWS (Ren et al., 2019) for a BERT-based sentiment classification model.

As suggested by several existing studies (Gao et al., 2018; Li et al., 2019; Ren et al., 2019; Jin et al., 2020), the vulnerability of PLMs is due to the inconsistent distributions between adversarial and clean examples, such as differences in word choice and sentence structure. Therefore, an intuitive defense method, Adversarial Data Augmentation (ADA) (Morris et al., 2020; Si et al., 2021), reduces this inconsistency by incorporating adversarial samples into the training set. As such, the model can learn to recognize subtle differences in characteristics between both types of samples (*clean* vs. *adversarial*), improving its robustness against adversarial attacks. Furthermore, the core idea of ADA echoes the consistency regularization principle in semi-supervised learning (Lee, 2013; Sohn et al., 2020; Kim et al., 2022): A model should produce the same prediction when small perturbations are applied to the input. To make ADA effective, two conditions must be met: First, the vector representations of adversarial and clean examples produced by the model must be closely aligned. Second, the model should assign different predicted labels to adversarial and clean examples, necessitating that the vector representations of clean examples and their adversarial counterparts lie near the model's decision boundary.

However, as illustrated in Figure 1, we observe that the representations of adversarial samples generated by the popular attack method PWWS (Ren et al., 2019) actually diverge from those of clean samples, and most are positioned far from the model's decision boundary. This phenomenon

LREC-COLING 2024, pages 15542–15551

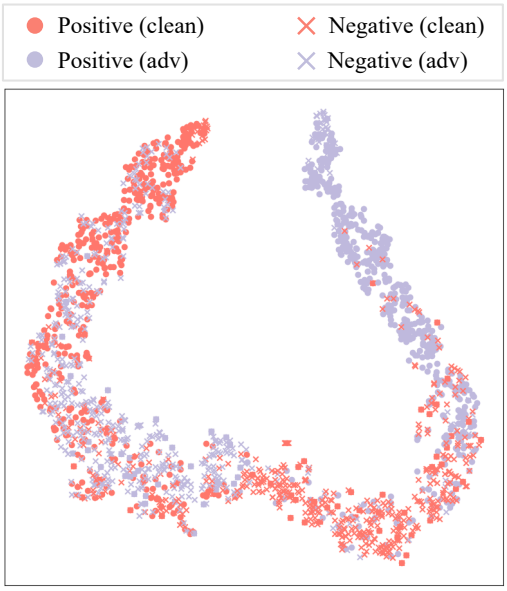


Figure 1: Illustration of BERT sentence representations of clean examples and their adversarial counterparts generated by PWWS on the SST-2 dataset.

calls into question the rationale behind using the same *one-hot label* for adversarial samples as their clean counterparts in ADA to enhance the robustness of the model (Si et al., 2021) since the underlying assumptions about vector representations do not hold. Our experimental results, which will be discussed in Tables 4 and 5, indirectly validate this by indicating that ADA can significantly decrease the model performance on clean samples. Therefore, although ADA may increase model robustness to some degree, it could also compromise the accuracy and generalizability of models due to the potential label-shift problem, thereby limiting the models' applicability in practical scenarios.

To address the issue mentioned above, we propose a novel method called *TaiChi*[1] that adopts a different approach from vanilla ADA to enhance the robustness of PLM-based text classification models. In the *TaiChi* framework, we aim to resolve the potential problem of label conflicts by guiding the model to generate similar representations for clean samples and their adversarial counterparts. Considering that label conflicts stem from a single model's need to process two types of samples with disparate characteristics, we introduce a Siamese network architecture in *TaiChi*. This architecture comprises two sub-networks, each trained on either clean or adversarial samples – referred to as the *clean* and *adversarial* models, respectively. Then, we apply a contrastive learning approach (Reimers and Gurevych, 2019; Chen et al., 2020; Clark et al., 2020; Conneau et al., 2020) to encourage the encoder of each model to produce similar vector representations for clean examples and their adversarial counterparts, while simultaneously distinguishing them from unrelated samples. Our intuition is that this will implicitly enable each model to make consistent predictions on both clean and adversarial samples. Furthermore, we employ the Kullback-Leibler (KL) divergence loss (Joyce, 2011) to promote information exchange between the clean and adversarial models, thus enhancing the consistency of their predictive behaviors. This facilitates a model-level integration and achieves a more favorable balance between generalization and robustness for both models.

Finally, we conduct extensive experiments on three widely used benchmark datasets to evaluate the effectiveness of *TaiChi*. The results[2] demonstrate that *TaiChi* achieves significantly better trade-offs between robustness to adversarial attacks at token and character levels and accuracy on clean examples compared to previous defense methods. Through ablation studies, we further verify the contributions of the adversarial model, contrastive loss, and Kullback-Leibler (KL) divergence loss components in the *TaiChi* framework separately.

2. Related Work

2.1. Adversarial Data Augmentation

Adversarial Data Augmentation (ADA) is a commonly used method to enhance the robustness of machine learning models. The fundamental principle of ADA involves generating adversarial samples with respect to clean ones, assigning each adversarial sample the same one-hot label as its clean counterpart, and incorporating them into the training set for model refinement. Traditionally, adversarial samples in ADA pertain exclusively to tangible texts produced by altering the original texts. Notable methods for this type of adversarial sample generation include DeepWordBug (Gao et al., 2018), TextBugger (Li et al., 2019), TextFooler (Jin et al., 2020), and Probability Weighted Word Saliency (PWWS) (Ren et al., 2019). In this study, we broaden the concept of adversarial examples to include "virtual" samples devised in the embedding space by perturbing word vectors, as in (Miyato et al., 2017; Madry et al., 2018; Jiang et al., 2020; Zhu et al., 2020). Owing to the generation method, virtual adversarial samples are often more distant from the originals, yet their vector representations tend to be closer to their real counterparts, potentially leading to enhanced model generalizability.

[1]"TaiChi" is an important concept in Chinese culture that symbolizes the interplay of two competitive and complementary forces, known as "Yin" and "Yang". This also signifies the philosophy of our work.

[2]We utilize the *clean model* for inference due to its higher prediction accuracy on clean samples.

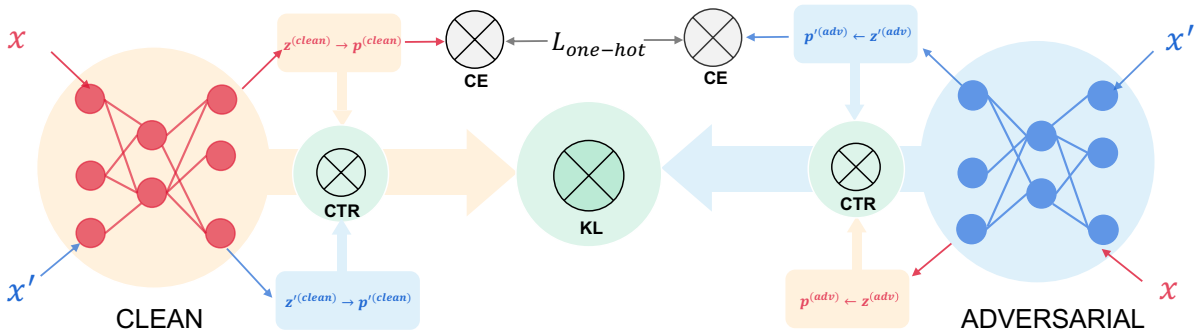


Figure 2: Framework of our method *TaiChi*. The neural network in red (CLEAN) signifies the model trained only on clean samples for classification, whereas the network in blue (ADVERSARIAL) denotes the model designed for adversarial examples. Here, x and x' refer to a clean sample and its adversarial counterpart, respectively. Additionally, z, p and z', p' represent the vector representations and the corresponding logits for x and x', respectively. Finally, $L_{\text{one-hot}}$, **CE**, **CTR**, and **KL** denote the one-hot labeling process, the text classification task, the contrastive data augmentation task, and the model-level fusion task, respectively.

Conversely, real adversarial samples, with vector representations that may be more divergent from the clean ones, can amplify the robustness of models more substantially than virtual samples, as they better approximate real-world adversarial instances. Furthermore, real texts offer superior explainability compared to abstract word vectors in NLP applications. Consequently, our research prioritizes the balance between robustness and generalizability in text classification models when employing real adversarial samples.

2.2. Contrastive Learning

Contrastive learning is a widespread representation learning technique with extensive applications across various fields, such as graph neural networks (You et al., 2020), natural language processing (Gao et al., 2021; Reimers and Gurevych, 2019; Chen et al., 2020; Conneau et al., 2020), computer vision (Dai and Lin, 2017), and speech recognition (Wang et al., 2022). Its core concept involves improving the representational capabilities of a model by focusing on and bringing together positive pairs while distancing negative pairs. In the context of text classification, a positive pair is defined as two samples that share semantic proximity. This work utilizes contrastive learning to diminish the representational gap between clean samples and their adversarial counterparts, striving to minimize discrepancies in their vector representations.

3. Our Method

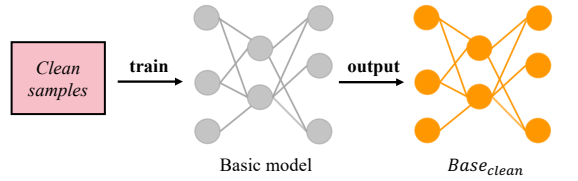
This section introduces the framework of our proposed method *TaiChi*. First, we provide a formal definition of the *adversarial samples* utilized in our approach. Subsequently, we explain the three training tasks that constitute our method.

3.1. The TaiChi Framework

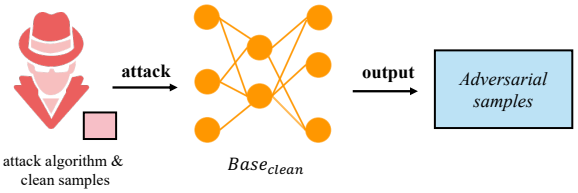
The framework of our method, *TaiChi*, is shown in Figure 2. It comprises two models with identical structures: the CLEAN model f_{clean} and the ADVERSARIAL model f_{adv}. In addition, there are three key training tasks: classification, data augmentation, and model-level fusion. The CLEAN and ADVERSARIAL models are distinguished primarily by the composition of their classification training sets. For classification, the CLEAN model only uses clean samples, while the ADVERSARIAL model exclusively uses adversarial samples. It is important to note that the ADVERSARIAL model is designed only to augment the CLEAN model and is not used for inference after the combined training process.

3.2. Basic Model Structure

The basic structure shared by the CLEAN and ADVERSARIAL models is described below. The model $f : \mathcal{X} \to \mathcal{C}$ consists of two main components: the encoder and the K-class classifier. The encoder, denoted $f_{\text{ENC}} : \mathcal{X} \to \mathcal{Z}$, where $\mathcal{X}$ is the input space and $\mathcal{Z}$ is the embedding space, transforms a sequence of m input tokens $x = [t_1, t_2, \ldots, t_m]$ into a sequence of vector representations $f_{\text{ENC}}(x) = [h_1, h_2, \ldots, h_m]$. The encoder then uses the average vector representation $z(x) = \frac{1}{m}\sum_{i=1}^{m} h_i$ as the aggregate sentence representation. The K-class classifier, represented as $f_{\text{CLS}} : \mathcal{Z} \to \mathcal{C}$, where $\mathcal{C} = \{1, 2, \ldots, K\}$ is the label space, maps the embedding space $\mathcal{Z}$ to label space $\mathcal{C}$. The classifier f_{CLS} can be decomposed into two functions: $u : \mathcal{Z} \to \mathbb{R}^K$, which projects the input embeddings onto *prediction logits* $p = u(z) = [p_1, p_2, \ldots, p_K]$, where $\sum_{i=1}^{K} p_i = 1$, and $v : \mathbb{R}^K \to \mathcal{C}$, which maps the logits to a specific category by choosing the label with the highest logit value, i.e., $c = \arg\max_{i \in [K]} p_i$.



Stage 1: train a basic model on clean samples



Stage 2: generate adversarial samples

Figure 3: Illustration of the adversarial sample generation procedure.

3.3. Generating Adversarial Samples

The core ideas of existing methods to generate adversarial examples are very similar and intuitive. Generally, an adversarial sample is generated by first identifying the words or characters in an (original) clean text with the largest impact on the prediction of the model and then modifying them accordingly. Therefore, in our method, we assume that an attacker's algorithm is already known, and the algorithm is directly used to simulate the attacker's activities on our fine-tuned model. This allows us to generate adversarial samples as an augmented training set to improve the robustness of the model.

Given a victim model f that has already been fine-tuned on a task-specific training set $D_{clean} = \{(x_i, y_i)\}_{i=1}^{n}$, where $y_i \in \mathbb{R}^K$ is the one-hot label and K represents the number of classes, we use a certain attack algorithm to construct an adversarial training set of the corresponding attack type $D_{adv} = \{(x'_i, y'_i)\}_{i=1}^{n}$, where each sample that is correctly classified in the original training set is wrongly classified after modifications, i.e., $f_\theta(x'_i) \neq f_\theta(x_i)$. In particular, when D_{clean} and D_{adv} are used together as the training set, x_i and x'_i with the same index are called a pair of clean and adversarial samples, and the adversarial sample x'_i shares the same one-hot label of its corresponding clean sample x_i, i.e., $y_i = y'_i$. An illustration of the adversarial sample generation procedure is presented in Figure 3.

3.4. Training Tasks

We detail the implementation of TaiChi as follows. Initially, we concurrently input a clean sample $x_i \in D_{\text{clean}}$ and its associated adversarial sample $x'_i \in D_{\text{adv}}$ into f_{clean} and f_{adv} to derive their average embeddings $z^{(\text{clean})}, z'^{(\text{clean})}, z^{(\text{adv})}, z'^{(\text{adv})}$, as well as the logit vectors $p^{(\text{clean})}, p'^{(\text{clean})}, p^{(\text{adv})}, p'^{(\text{adv})}$. Subsequently, we outline the three training tasks.

Classification. The primary training objective in our method is text classification. We distinguish the two models based on their respective classification tasks. Specifically, the CLEAN model f_{clean} is solely tasked with classifying clean samples from D_{clean}. The cross-entropy loss for f_{clean} is thus given as

$$\mathcal{L}_{\text{CE}}^{\text{clean}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{K} y_{i,c} \log p_{i,c}^{(\text{clean})},$$

where N denotes the batch size. Parallelly, the ADVERSARIAL model f_{adv} exclusively processes adversarial samples from D_{adv} for classification. Thus, the cross-entropy loss for f_{adv} is

$$\mathcal{L}_{\text{CE}}^{\text{adv}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{K} y'_{i,c} \log p'^{(\text{adv})}_{i,c}.$$

Data Augmentation. Ideally, a model's resilience against noise is bolstered when the representations of clean and adversarial samples produced by PLMs are highly similar. Meanwhile, we aim for the model to discern true adversarial examples from irrelevant samples, rather than indiscriminately reducing the distance between representations of arbitrary sample pairs. To achieve both objectives, we incorporate contrastive learning as an additional regularization mechanism during fine-tuning. Specifically, we treat the sentence representations $z_i^{(\text{clean})}$ of a clean sample $x_i^{(\text{clean})}$ and $z_i'^{(\text{clean})}$ of its corresponding adversarial sample $x_i'^{(\text{clean})}$ as a positive example pair. For a batch of N clean and corresponding adversarial samples, we consider $z_j'^{(\text{clean})}$, for all $j \neq i$, as negative samples relative to a specific $z_i^{(\text{clean})}$. Consequently, the contrastive loss objective based on clean samples is formulated as follows:

$$\mathcal{L}_{\text{CTR}}^{(\text{clean})} = -\sum_{i=1}^{N} \log \frac{\exp(s(z_i^{(\text{clean})}, z_i'^{(\text{clean})})/\tau)}{\sum_{j=1}^{N} \mathbb{1}_{[j\neq i]} \exp(s(z_i^{(\text{clean})}, z_j'^{(\text{clean})})/\tau)},$$

where $s(u, v) = \|u - v\|_2$ denotes the l_2-distance between two vectors u and v, τ is the temperature hyperparameter, and $\mathbb{1}_{[j\neq i]}$ is an indicator function for $j \neq i$. Analogously, the contrastive objective for the adversarial samples is

$$\mathcal{L}_{\text{CTR}}^{(\text{adv})} = -\sum_{i=1}^{N} \log \frac{\exp(s(z_i'^{(\text{adv})}, z_i^{(\text{adv})})/\tau)}{\sum_{j=1}^{N} \mathbb{1}_{[j\neq i]} \exp(s(z_i'^{(\text{adv})}, z_j^{(\text{adv})})/\tau)}.$$

Model-level Fusion. Using the classification and data augmentation tasks, we have built two distinct models: CLEAN and ADVERSARIAL. We then turn to the central concept of *TaiChi*, which involves facilitating communication between these seemingly antagonistic models to find commonalities while preserving their unique strengths. In practical terms, if

Method	Data Augmentation	CLS	Training Tasks	Adversarial Model
Base_{clean}	n/a	clean	CE	×
Base_{adv}	n/a	adv	CE	×
FGSM (Goodfellow et al., 2015)	virtual	clean	CE	×
PGD (Madry et al., 2018)	virtual	clean	CE	×
FreeLB (Zhu et al., 2020)	virtual	clean	CE	×
SMART (Jiang et al., 2020)	virtual	clean	CE	×
ADA (Si et al., 2021)	real text	clean+adv	CE	×
CLEAN / $\text{TaiChi}_{w/o\ KL}$	real text	clean	CE+CTR	×
ADVERSARIAL	real text	adv	CE+CTR	×
Base_{KL} / $\text{TaiChi}_{w/o\ CTR}$	real text	clean	CE+KL	✓
$\text{TaiChi}_{w/o\ ADV}$	real text	clean	CE+CTR+KL	×
TaiChi (*)	real text	clean	CE+CTR+KL	✓

Table 2: Comparison of methods used in the experiments. "*Data augmentation*" describes the types of adversarial samples employed (*n/a* for none, *virtual* for samples generated in the embedding space by perturbing word vectors, and *real text* for samples created by direct perturbations of the original texts). "*CLS*" specifies the training data used for text classification (*clean* for clean samples only, *adv* for adversarial samples only, and *clean+adv* for a combination of both). "*Training Tasks*" indicates which of the three tasks – CE, CTR, and KL – are utilized in the training process. Finally, "*Adversarial Model*" denotes the presence or absence of an adversarial model for model-level fusion.

two models have been effectively fused, saying that they have assimilated the characteristics of each other's training data, they should yield consistent predictions for the same input samples. That is, for an input sample x, the logits $p^{(\text{clean})}$ and $p^{(\text{adv})}$ produced by the CLEAN and ADVERSARIAL models, respectively, should closely resemble each other. We employ the Kullback-Leibler (KL) divergence loss (Joyce, 2011) to implement this model-level fusion. The KL divergence loss is distinct for clean and adversarial samples. For clean samples, the KL divergence loss function is defined as

$$\mathcal{L}_{\text{KL}}^{(\text{clean})} = -\frac{1}{N}\sum_{i=1}^{N} p_i^{(\text{clean})} \log \frac{p_i^{(\text{adv})}}{p_i^{(\text{clean})}}.$$

The KL divergence loss function for adversarial samples is defined as

$$\mathcal{L}_{\text{KL}}^{(\text{adv})} = -\frac{1}{N}\sum_{i=1}^{N} p_i'^{(\text{adv})} \log \frac{p_i'^{(\text{clean})}}{p_i'^{(\text{adv})}}.$$

Overall Training Objective. The models are fine-tuned in a multi-task fashion with the combined training objective:

$$\mathcal{L} = \underbrace{(\mathcal{L}_{\text{CE}}^{(\text{clean})} + \mathcal{L}_{\text{CTR}}^{(\text{clean})})}_{\text{CLEAN}} + \underbrace{(\mathcal{L}_{\text{CE}}^{(\text{adv})} + \mathcal{L}_{\text{CTR}}^{(\text{adv})})}_{\text{ADVERSARIAL}} + \underbrace{(\mathcal{L}_{\text{KL}}^{(\text{clean})} + \mathcal{L}_{\text{KL}}^{(\text{adv})})}_{\text{KL DIVERGENCE}},$$

where the superscripts (clean) and (adv) denote the objectives specific to the CLEAN and ADVERSARIAL models and the subscripts CE, CTR, and KL correspond to the objectives of the classification, contrastive data augmentation, and model-level fusion tasks, respectively.

4. Experiments

4.1. Setup

Methods. We benchmark our method, TaiChi, against five well-established defense techniques for countering adversarial attacks: FGSM (Goodfellow et al., 2015), PGD (Madry et al., 2018), FreeLB (Zhu et al., 2020), SMART (Jiang et al., 2020), and ADA (Si et al., 2021). For baselines, we use two BERT-based text classification models (Devlin et al., 2019) fine-tuned on clean samples (Base_{clean}) and adversarial samples (Base_{adv}), respectively. Additionally, we conduct ablation studies to assess each component of TaiChi in isolation. We designate the models involved in these studies as follows: CLEAN (or $\text{TaiChi}_{w/o\ KL}$) and ADVERSARIAL are the models that only incorporate the CTR data augmentation objective and differ in their training sets for classification. $\text{TaiChi}_{w/o\ CTR}$ (or Base_{KL}) refers to the model trained exclusively on clean samples, with the CTR objective omitted. $\text{TaiChi}_{w/o\ ADV}$ represents the model trained on clean samples, modifying the original dual-model fusion training objective to include the KL divergence loss within a single model. The KL divergence loss for $\text{TaiChi}_{w/o\ ADV}$ is defined as follows:

$$\mathcal{L}_{KL} = -\frac{1}{N}\sum_{i=1}^{N} p_i^{(\text{clean})} \log \frac{p_i'^{(\text{clean})}}{p_i^{(\text{clean})}}.$$

Consequently, the overall training objective of $\text{TaiChi}_{w/o\ ADV}$ is given as follows:

$$\mathcal{L} = \mathcal{L}_{CE}^{(\text{clean})} + \mathcal{L}_{CTR}^{(\text{clean})} + \mathcal{L}_{KL}.$$

The distinctions among these methods are outlined in Table 2 for clarity.

Dataset	C	L	#train	#dev	#test
SST-2	2	19	6.9K	872	1.8K
AG	4	32	114K	6K	7.6K
TREC	6	10	5K	452	500

Table 3: Statistics of datasets used in the experiments, where C represents the number of classes, L is the average sentence length, and #train, #dev, and #test denote the numbers of samples in the training, validation, and test sets, respectively.

Datasets. We use three text classification datasets with different numbers of classes and tasks in the experiments, namely, SST-2 (Socher et al., 2013) for binary sentiment classification, AG (Zhang et al., 2015) for four-class news classification, and TREC (Li and Roth, 2006) for six-class question classification. The statistics of these datasets are reported in Table 3.

Performance Measures. The performance of a model is evaluated in terms of *generalizability* and *robustness*. Generalizability is assessed by the model's test accuracy on clean samples, denoted as *clean%*. To gauge the robustness of a model, we employ the following four prevalent attack methods:

- **DeepWordBug** (Gao et al., 2018) operates at the character level, scoring and perturbing the most influential tokens in a sentence by swapping, substituting, deleting, or inserting characters.
- **PWWS** (Ren et al., 2019) is a word-level attack that uses word saliency and classification probability to determine which word to substitute and in what sequence.
- **TextFooler** (Jin et al., 2020) also targets words, choosing replacements based on their significance – measured by the impact of their removal on class probability – and synonym similarity in the embedding space.
- **TextBugger** (Li et al., 2019) functions at both word and character levels by modifying selected target tokens using various techniques such as synonym replacement and character-level perturbations.

The robustness of the model is measured by the accuracy on adversarial samples generated by each attack method, denoted as *deepwordbug%*, *pwws%*, *textfooler%*, and *textbugger%*, respectively. We used the implementations of these attack algorithms provided in the TextAttack framework (Morris et al., 2020).

Implementation Details. Adversarial attacks were conducted using different sample sizes across various datasets. For the SST-2 and AG datasets, we randomly selected 1,000 samples from the test sets; and for the TREC dataset, we attacked all 500 test samples. To train the models, we set the batch size to 16, the maximum sequence length to 50, the number of training epochs to 3, and the learning rate to 3×10^{-5}. We used *bert-base-uncased*[3] as the pre-trained model, with other hyperparameters defaulting to those of the Transformer model (Wolf et al., 2020). The baseline implementations were sourced from the original authors, and their default parameter configurations were followed. All experiments were performed on a server with two Intel Xeon Silver 4210R 2.40 GHz CPUs and an NVIDIA Tesla V100 SXM2 32 GB GPU.

Measure	Base_{clean}	Base_{adv}	Base_{adv} (soft)
clean%	92.6	76.7	88.9

Table 4: Test accuracy on clean samples (clean%) of different models on the SST2 dataset. All models in this suite of experiments are trained from scratch.

4.2. Results

Validation of Label Conflict. To substantiate our assertion in Section 1 regarding the potential inappropriateness of ADA's practice of assigning identical one-hot labels to adversarial samples as their clean counterparts, we use the adversarial attack algorithm PWWS for illustration. In both Base_{clean} and Base_{adv}, clean and adversarial samples are labeled with their ground-truth labels. Conversely, Base_{adv} (soft) differs from Base_{adv} in that it utilizes the *predicted logits* from Base_{clean} as soft labels. For instance, the adversarial training sample "spotless performance by xxx" is labeled [0,1] (positive) in Base_{adv}, whereas in Base_{adv}(soft) it receives a soft label of [0.8,0.2] (leaning negative). The findings presented in Table 4 reveal that Base_{adv} records the lowest accuracy on clean test samples, which is in line with our expectations. Moreover, although Base_{adv} (soft) trails behind Base_{clean} for clean sample test accuracy, it shows a notable enhancement over Base_{adv}. These observations lend credence to the concerns highlighted in Section 1 and may catalyze further investigation into labeling strategies to bolster the robustness of the NLP model. In our work, we address label conflicts by incorporating an adversarial model, applying a contrastive learning approach, and leveraging the KL divergence loss for model optimization.

Effectiveness of TaiChi. To assess the efficacy of our approach *TaiChi*, we benchmark it against five established defense strategies for mitigating adversarial attacks in terms of generalizability and robustness. As detailed in Table 5, *TaiChi* significantly

[3] https://huggingface.co/google-bert/bert-base-uncased

Dataset	Method	*clean%*	*deepwordbug%*	*pwws%*	*textfooler%*	*textbugger%*
SST-2	Base$_{clean}$	92.6	21.1	16.0	7.1	37.3
	PGD	92.5	22.8	22.5	10.6	39.9
	FGSM	90.9	26.5	21.9	12.5	36.9
	FreeLB	91.7	21.7	19.0	8.3	35.1
	SMART	**92.6**	25.3	20.2	12.8	38.8
	ADA	90.8	22.5	28.9	17.3	38.1
	TaiChi	91.7 (↓ 0.9)	**34.1** (↑ 7.6)	**34.8** (↑ 5.9)	**20.3** (↑ 3.0)	**51.0** (↑ 11.1)
TREC	Base$_{clean}$	94.6	42.6	53.4	42.2	64.8
	PGD	**95.4**	51.0	59.0	46.4	66.6
	FGSM	94.8	48.4	53.2	40.0	62.4
	FreeLB	95.0	52.4	54.0	40.0	62.8
	SMART	94.8	48.6	53.0	40.8	62.8
	ADA	91.9	54.6	64.0	40.4	72.4
	TaiChi	93.9 (↓ 1.5)	**61.0** (↑ 6.4)	**68.0** (↑ 4.0)	**51.8** (↑ 5.4)	**75.2** (↑ 2.8)
AG	Base$_{clean}$	94.4	15.7	25.5	10.0	31.5
	PGD	**94.4**	41.3	19.6	44.8	19.3
	FGSM	93.4	41.3	58.0	35.6	42.7
	FreeLB	94.0	40.0	55.8	35.2	43.2
	SMART	93.0	18.7	22.7	6.8	29.5
	ADA	92.9	45.4	48.9	35.9	24.1
	TaiChi	93.8 (↓ 0.6)	**48.8** (↑ 3.4)	**65.7** (↑ 7.7)	**45.3** (↑ 0.5)	**50.7** (↑ 7.5)

Table 5: Results of different defense methods against four types of adversarial attacks. For ADA and TaiChi, we report their average $clean\%$ of four types of attacks. For TaiChi, we compare its score on each measure with the best among all remaining methods (in bracket).

PLM	Method	*clean%*	*deepwordbug%*	*pwws%*	*textfooler%*	*textbugger%*
BERT-Large	Base$_{clean}$	93.5	15.8	16.6	6.5	36.4
	ADA	90.9	26.8	21.3	11.7	43.5
	TaiChi	92.9 (↓ 0.6)	**34.9** (↑ 8.1)	**33.2** (↑ 11.9)	**15.4** (↑ 3.7)	**53.6** (↑ 10.1)
DeBERTa	Base$_{clean}$	93.0	22.9	20.9	8.8	40.7
	ADA	92.5	29.6	30.5	13.7	49.2
	TaiChi	92.6 (↓ 0.4)	**39.5** (↑ 9.9)	**37.1** (↑ 6.6)	**19.3** (↑ 5.6)	**52.7** (↑ 3.5)

Table 6: Additional results of Base$_{clean}$, ADA, and TaiChi against four types of adversarial attacks on the SST-2 dataset when BERT-Large and DeBERTa are used as base models.

enhances the robustness of text classification models against all four attack methods across three different datasets. The robustness metrics for *TaiChi* are consistently at least 3% higher than those of the best-performing existing methods. In certain cases, it even shows improvements of more than 10% compared to the second-best method. Meanwhile, the detrimental impact on accuracy in clean samples is marginal, not exceeding a reduction of 1.5%. In particular, *TaiChi* performs better on both generalizability and robustness than ADA in all datasets. This outcome underscores the efficacy of the contrastive learning-based data augmentation and the KL divergence loss-based model-level fusion techniques used by *TaiChi*, compared to enriching the training set by directly adding adversarial samples assigned with one-hot labels, as is done in ADA.

Finally, we conduct additional experiments on the SST-2 dataset using BERT-Large[4] and DeBERTa (He et al., 2021) as base models. The results are reported in Table 6, which further validates the applicability of *TaiChi* to other PLMs.

[4] https://huggingface.co/google-bert/bert-large-uncased

4.3. Ablation Studies

Overall Ablation Results. In our ablation study on the SST-2 dataset, we dissect the contributions of data augmentation and model-level fusion to the performance of TaiChi separately. The key insights obtained, as depicted in Figure 4, are as follows:

- First, TaiChi$_{w/o\ CTR}$, which incorporates only model-level fusion, exhibits marginally lower accuracy on clean samples than Base$_{clean}$ but significantly better robustness.
- Second, TaiChi$_{w/o\ KL}$, which employs only data augmentation, outperforms Base$_{clean}$ in accuracy for both clean and adversarial samples.
- Third, the accuracy of TaiChi on clean samples is slightly lower than that of TaiChi$_{w/o\ KL}$. But it shows significantly better robustness against all four types of attacks.

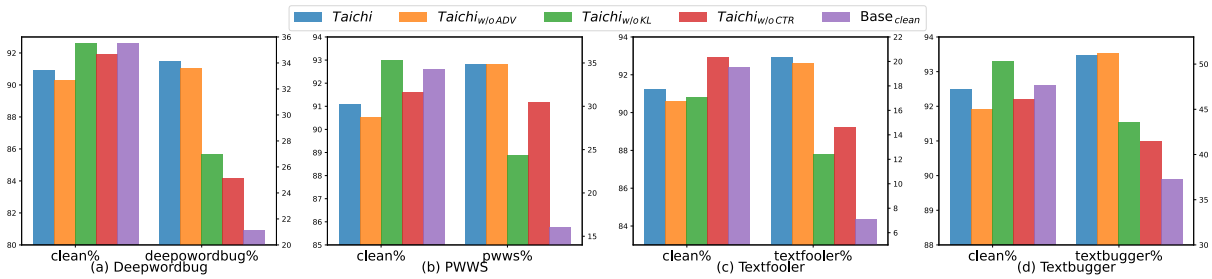


Figure 4: Ablation studies for TaiChi on the SST-2 dataset.

bert-base-uncased	Before Fine-Tune	Base$_{clean}$	TaiChi$_{w/o\ KL}$	TaiChi$_{w/o\ CTR}$	TaiChi$_{w/o\ ADV}$	TaiChi
MSE (pos, ↓)	0.0393	0.2830	0.0638	0.0966	0.0304	**0.0303**
MSE (neg, ↑)	0.1069	0.5935	**0.7563**	0.4713	0.5848	0.6275
KL (pos, ↓)	0.0030	1.3690	0.5222	0.2207	0.1771	**0.1206**
KL (neg, ↑)	0.0065	4.2273	**4.7785**	4.0947	2.1123	2.3727
CTR (↓)	6.6380	6.8853	6.0832	6.5214	6.0550	**6.0305**
clean% (↑)	49.6	92.6	**92.9**	92.4	90.4	91.6
textfooler% (↑)	1.0	7.1	12.4	14.6	19.8	**20.3**

Table 7: High-order statistical information exploring the abilities of different models to learn text representations on the SST-2 dataset under the TextFooler attack. Here, the average MSE and KL-divergence losses between *positive* and *negative* sample pairs and the average contrastive learning loss are computed from 1K clean and adversarial sample pairs. Each pair of clean and adversarial samples forms a *positive* sample pair. The *negative* pairs are constructed by assigning a fixed sentence with a negative label to all clean samples with positive labels, and vice versa.

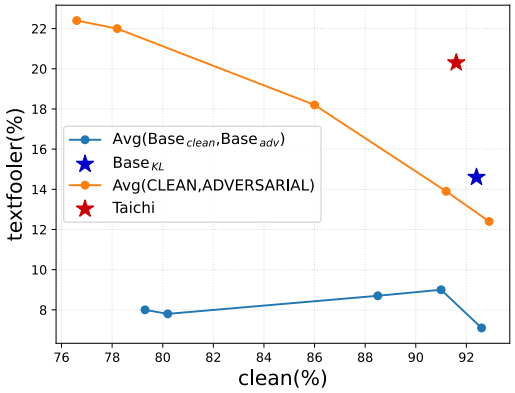


Figure 5: KL divergence loss vs. linear average for model-level fusion on the SST-2 dataset under the TextFooler attack.

- Fourth, although TaiChi$_{w/o\ ADV}$ underperforms TaiChi in terms of accuracy on clean samples, it achieves comparable robustness, highlighting the efficacy of dual-model strategies and reinforcing the value of the CTR and KL tasks in bolstering model robustness.

KL Divergence Loss vs. Linear Average for Model-level Fusion. We compare our KL divergence loss-based fusion technique with a straightforward linear average model aggregation method, where a new set of model parameters, f_{avg}, is computed as a weighted average: $f_{avg} = \lambda f_{clean} + (1-\lambda) f_{adv}$, for a hyperparameter $\lambda \in (0, 1)$. As illustrated in Figure 5, our method offers a more advantageous balance between robustness and generalizability than linear averaging due to the interactive aspect of model fusion during training.

High-order Statistics. The statistics presented in Table 7 indicate a strong correlation between the consistency of model predictions for clean and adversarial sample pairs (*KL (pos)*) and the overall model robustness, aligning with our hypothesis. The model-level fusion task directly improves *KL (pos)* by calibrating prediction outcomes, while the data augmentation task indirectly enhances it by reducing the distances between the two types of samples in the embedding space (*MSE (pos)*). Additionally, we posit that any loss in clean sample accuracy attributable to our method may result from the oversight of prediction inconsistencies between samples with different labels (*KL (neg)*).

5. Conclusion

In this paper, we investigate the problem of improving the robustness of text classification models by leveraging adversarial data augmentation. We identify the shortcomings of previous methods, that is, potentially unreasonable label assignments. To address the above issue, we propose a novel method, *TaiChi*, which introduces an additional adversarial model that shares the same structure as the original (clean) model, adopts a contrastive

learning approach to data augmentation, and utilizes the KL divergence loss for the fusion of clean and adversarial models. Extensive experiments on three widely used benchmark datasets confirm that TaiChi achieves better generalizability-robustness trade-offs than existing defense methods.

6. Acknowledgements

This work was supported by the National Natural Science Foundation of China under grant numbers 62202170 and 62202169, Alibaba Group through the Alibaba Innovation Research Program, and CCF-AFSG Research Fund.

7. Bibliographical References

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, pages 1597–1607.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *Proceedings of the 8th International Conference on Learning Representations*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Bo Dai and Dahua Lin. 2017. Contrastive learning for image captioning. In *Advances in Neural Information Processing Systems 30*, pages 898–907.

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

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

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *Conference Track Proceedings of the 3rd International Conference on Learning Representations*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *Proceedings of the 9th International Conference on Learning Representations*.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. 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*, pages 2177–2190.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT really robust? a strong baseline for natural language attack on text classification and entailment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8018–8025.

James M. Joyce. 2011. Kullback-Leibler divergence. In Miodrag Lovric, editor, *International Encyclopedia of Statistical Science*, pages 720–722. Springer.

Jiwon Kim, Youngjo Min, Daehwan Kim, Gyuseong Lee, Junyoung Seo, Kwangrok Ryoo, and Seungryong Kim. 2022. ConMatch: Semi-supervised learning with confidence-guided consistency regularization. In *Proceedings of the 17th European Conference on Computer Vision - Part XXX*, pages 674–690.

Dong-Hyun Lee. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on Challenges in Representation Learning*.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium*.

Xin Li and Dan Roth. 2006. Learning question classifiers: the role of semantic information. *Nat. Lang. Eng.*, 12(3):229–249.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *Conference Track Proceedings of the 6th International Conference on Learning Representations*.

Takeru Miyato, Andrew M. Dai, and Ian J. Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. In *Conference Track Proceedings of the 5th International Conference on Learning Representations*.

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

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

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

Chenglei Si, Zhengyan Zhang, Fanchao Qi, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2021. Better robustness by more coverage: Adversarial and mixup data augmentation for robust finetuning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1569–1576.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A. Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. 2020. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems 33*, pages 596–608.

Yiming Wang, Jinyu Li, Heming Wang, Yao Qian, Chengyi Wang, and Yu Wu. 2022. Wav2vec-switch: Contrastive learning from original-noisy speech pairs for robust speech recognition. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7097–7101.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. In *Advances in Neural Information Processing Systems 33*, pages 5812–5823.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28*, pages 649–657.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2020. FreeLB: Enhanced adversarial training for natural language understanding. In *Proceedings of the 8th International Conference on Learning Representations*.