Turn Up the Heat: Training with High Temperatures Boosts Robustness Against Unseen Adversarial Attacks

Anonymous ACL submission

Abstract

Deep learning models have achieved remarkable performance across various domains, but are vulnerable to adversarial attacks. Existing defences such as adversarial training face challenges when applied to NLP models due to the computational complexity, while others are form-specific. A prevalent practical strategy is augmentation-based adversarial training, where adversarial examples are included in the training set. While successful, this approach largely only improves robustness against the specific attack forms the model is trained on and its training time scales linearly with the augmentation factor. We propose a simple modification to the standard training algorithm which boosts absolute accuracy in the presence of adversarial examples up to 14 accuracy points, without increasing model training time. Our modification is the use of a high temperature parameter during training to scale down predicted logits from classification systems. We finally show that this high temperature training approach complements existing adversarial training techniques, further improving the adversarial robustness of augmentation-based, adversarially trained NLP systems against unseen adversarial attacks.[1]

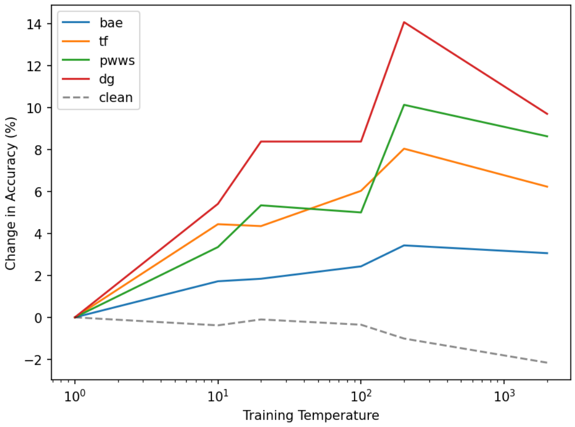


Figure 1: The addition of a training temperature is a simple modification to standard model training, where the temperature parameter, T is used to scale down the predicted model logits during training. We observe that a higher temperature during training increases model robustness against adversarial attacks unseen during model training (*bae, tf, pwws, dg* here). The increase in model robustness is demonstrated by the absolute change in adversarial accuracy (accuracy when test samples are adversarially attacked) relative to the baseline model ($T = 1$). An optimal training temperature can be identified as the temperature that yields substantial gains in robustness, with only a minor reduction in clean accuracy (accuracy in the absence of adversarial attacks).

1 Introduction

Due to their impressive performance, deep learning models have been deployed in a range of areas in natural language processing (NLP). However, these models are susceptible to adversarial attacks, where small perturbations to the input can result in large, undesired changes in the model's prediction. These perturbed inputs are termed adversarial examples (Szegedy et al., 2014). The presence of adversarial examples is ubiquitous and a real threat to NLP systems used in high stakes situations (Sun et al., 2018; Tan et al., 2021; Raina et al., 2022).

Although many adversarial attack algorithms demonstrate the ease with which adversarial examples can be discovered, there also exist approaches to defend against these attacks (Piktus et al., 2019; Tan et al., 2020b; Raina and Gales, 2022). The standard approach is to use adversarial training (Goodfellow et al., 2015; Bai et al., 2021), where the default training scheme of deep learning systems is modified to minimize the empirical risk associated with the worst-case adversarial example for each input. Implementation of adversarial training in the computer vision domain has shown some success in the development of more adversarially robust models. However for NLP models, practical implementation of the min-max formulation of the adversarial training paradigm is challenging: generating

[1] We include the code modification in Appendix E.

the worst-case adversarial example for each textual input in every training step is too computationally expensive (Yoo and Qi, 2021). Various strategies can be used to adapt the adversarial training algorithm for NLP models (Wang et al., 2019b), but the practical, de facto approach is equivalent to naive data augmentation. Here, adversarial examples are created from the training set of a NLP model trained in the standard manner and then the model is re-trained on a training set augmented with these adversarial examples (Zhang et al., 2020).

Augmentation-based adversarial training approaches have shown some success in developing more robust systems, but it has also been observed that these systems develop robustness to only a specific form of adversarial attack, i.e. the form of adversarial attack used to generate the adversarial examples for the augmentation (e.g., in Jin et al. 2019). Therefore, the development of more general adversarial robustness to unseen forms of adversarial attacks is an open research question.

In this work, we make significant progress in this research direction. We find that using a high temperature parameter to scale down the predicted logits of a NLP classification system during standard training boosts model robustness without added computational cost (Figure 1). This simple modification is independent of the form of the adversarial attack, yet is complementary with existing NLP adversarial training schemes (Madry et al., 2018; Zhu et al., 2020; Dong et al., 2021). Since the model is not exposed to any adversarial examples during training, this makes it robust to unseen adversarial attacks, by definition. We conduct experiments across multiple datasets, encoder models, and adversarial attacks to demonstrate its efficacy at improving robustness against unseen adversarial attacks. Our experiments show it is possible to choose a checkpoint that yields substantial gains in robustness with a relatively negligible (sometimes no) decrease in clean accuracy. Critically, the trade-off is often consistent across adversarial attacks for a model finetuned on a particular dataset. This implies that we can use *known* adversarial attacks to select a temperature that will be robust to a future adversarial attack of an *unknown* form. In summary, we demonstrate that high temperature training is an effective approach to boost the adversarial robustness of standard and adversarially-trained NLP encoder models to unseen adversarial attack forms.

2 Training Methodology

The adversarial robustness literature often discusses training-time improvements in terms of the training objective, i.e., standard training (ST) v.s. adversarial training (AT). In this work, we posit that the training objective is not the only dimension that can be manipulated to affect test-time robustness, and there exists yet another: training temperature. We specifically demonstrate that the use of a high temperature during training can improve model robustness. Table 1 summarises the naming convention for the combination of ST and AT systems with a high training temperature, T.

	standard	adversarial
$T = 1$	ST	AT
High T	ST $\oplus T$	AT $\oplus T$

Table 1: Naming convention for experiments with different training objectives and high temperature training.

2.1 Training Objective

Standard Training (ST) methods have the objective to find model parameters, θ that minimise the empirical risk (for a dataset of $(\mathbf{x}, y) \sim p(\mathbf{x}, y)$),

$$\hat{\theta} = \arg\min_{\theta} \underset{(\mathbf{x},y)\sim p(\mathbf{x},y)}{\mathbb{E}} [\mathcal{L}(\mathbf{x}, \theta)], \quad (1)$$

where a cross-entropy loss is used for classification tasks,

$$\mathcal{L}(\mathbf{x}, \theta) = \log p(y|\mathbf{x}; \theta). \quad (2)$$

The objective in Adversarial Training (AT) is to instead minimise the empirical risk associated with the *worst-case* adversarial example, $\tilde{\mathbf{x}}$,

$$\hat{\theta} = \arg\min_{\theta} \underset{(\mathbf{x},y)\sim p(\mathbf{x},y)}{\mathbb{E}} \left[\max_{\substack{\tilde{\mathbf{x}}: \\ \mathcal{G}(\mathbf{x},\tilde{\mathbf{x}})\leq\epsilon}} \mathcal{L}(\tilde{\mathbf{x}}, \theta) \right]. \quad (3)$$

Note that $\mathcal{G}(\mathbf{x}, \tilde{\mathbf{x}}) \leq \epsilon$ represents a constraint on the adversarial example $\tilde{\mathbf{x}}$ to ensure it is a ‘small’ distance from the clean sample $\mathbf{x}$. In the computer vision domain, this constraint is typically the l^p-norm on the perturbation size, whilst for NLP this constraint is more abstract as it limits the change in semantic content of a textual input as measured by a proxy function; e.g., embedding space distances or pre-transformation constraints such as only allowing replacements from the same part-of-speech category (Tan et al., 2020a).

Note that in practice, for NLP models, generating textual adversarial examples for each clean input $\mathbf{x}$ at every iteration step of training is too slow. Therefore the de facto solution is to generate adversarial examples once on a model trained with the standard objective (Equation 1). Then these adversarial examples are used to augment the training dataset. Standard training can then be followed with the augmented training dataset to simulate AT for NLP models. Nevertheless, there are embedding-based NLP AT approaches such as PGD (Madry et al., 2018) and FreeLB (Zhu et al., 2020), as well as region-based approaches such as ASCC (Dong et al., 2021) that aim to follow the AT training process of Equation 3 more directly - these approaches are included for comparison in the experiments (Section 3.2).

2.2 Training Temperature

The cross-entropy loss (Equation 2) is used to characterize the empirical risk during training. This loss function uses the model's predicted probability of the true class, y. For a model's predicted logits, $l_1, \ldots, l_C$ for classes 1 to C, the probability of the true class y is typically given by the softmax function,

$$p(y|\mathbf{x}) = \frac{\exp(l_y)}{\sum_i \exp(l_i)}. \tag{4}$$

The training algorithm can be adjusted to include a training temperature T, such that the probability distribution used to compute the loss function $\mathcal{L}$ is manipulated,

$$p(y|\mathbf{x}) = \frac{\exp(l_y/T)}{\sum_i \exp(l_i/T)}. \tag{5}$$

Intuitively, a larger temperature T encourages a flatter probability distribution over classes and may be viewed as making it more challenging for the model to minimize the empirical loss during training. In this work, we show that a high training temperature T boosts model robustness. Note that using a high training temperature makes no assumptions about the nature of the adversarial attack as no specific attack form is used during training. Therefore, this is an effective approach to build robustness against unseen adversarial attack forms since no adversarial examples are seen during training time.

3 Experiments

We now study the effect of high temperature training on robustness against four common adversarial attacks using five NLP classification datasets.

3.1 Experimental Setup

Data. We conduct experiments across five standard NLP classification datasets to ensure our findings are robust (statistics summarised in Table 2). Rotten Tomatoes (*rt*; Pang and Lee, 2005) is a binary sentiment classification task for movie reviews. The Emotion Dataset (*emotion*; Saravia et al., 2018) categorizes Twitter tweets into one of six emotions: love, joy, surprise, fear, sadness or anger. The remaining three datasets are sourced from the the General Language Understanding Evaluation (GLUE) benchmark (Wang et al., 2019a).[2] The Corpus of Linguistic Acceptability (*cola*) dataset comprises English acceptability judgments sourced from books and journal articles on linguistic theory. Each instance consists of a word sequence annotated to indicate if it is grammatically correct. The Question-answering NLI (*qnli*) dataset assesses the task of sentence pair classification, where one sentence is a question and the other a context. The goal is to ascertain whether the context sentence contains the answer to the question. The Microsoft Research Paraphrase Corpus (*mrpc*) consists of pairs of sentences automatically extracted from online news sources. Human annotations identify if the sentences in each pair are semantically equivalent.

Dataset	#classes	Train	Validation	Test
rt	2	8.53k	1.07k	1.07k
emotion	6	16k	2k	2k
cola	2	8.55k	1.04k	1.06k
qnli	2	105k	5.46k	5.46k
mrpc	2	3.67k	408	1.73k

Table 2: Dataset statistics

Models. We finetune state-of-the-art pretrained encoder-based Transformer (Vaswani et al., 2017) models on each task. We present results on the DeBERTa base (110M parameters) model (He et al., 2020) here, but note that we observe identical trends for BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019). Results for the BERT and RoBERTa models are given in Appendix B.

Adversarial Attacks. We explore four widely used off-the-shelf adversarial attack methods. Firstly, the BERT Adversarial Example (*bae*; Garg

[2] For datasets where the provided test set is not labeled, we used the validation set.

and Ramakrishnan, 2020) operates as a word-level blackbox attack, where the adversary only possesses access to the model inputs and predicted logits. Subsequently, we incorporate more effective approaches: Textfooler (*tf*; Jin et al., 2019) and Probability Weighted Word Saliency (*pwws*; Ren et al., 2019). These also operate as word-level, blackbox adversarial attacks. As the fourth attack, we consider the DeepWordBug (*dg*; Gao et al., 2018) attack, functioning as a whitebox approach at the *character* level. Each of these adversarial attacks adheres to default configurations as outlined in the TextAttack Library (Morris et al., 2020). In this work, we consider an adversarial attack as *seen* when the model developer has access to that specific attack during training; otherwise an attack is considered *unseen*. It is desirable to have robustness to both seen and unseen attack forms. To assess the impact of these diverse adversarial attacks, we measure the adversarial accuracy. This metric is the target model's accuracy when presented with adversarial examples as the input for each test sample.

Adversarial Training Methods. The experiments in this work explore the impact of combining the high temperature training (Section 2.2) approach with various standard NLP Adversarial Training (AT) baseline approaches. Specifically, we explore PGD-K (Madry et al., 2018) and FreeLB (Zhu et al., 2020) as embedding-space AT schemes, alongside ASCC (Dong et al., 2021) as a combined text-embedding AT approach. Additionally, we investigate the widely used NLP AT technique: augmenting the training set with adversarial examples. Here, adversarial examples are generated by training the target model conventionally (as per Equation 1) and then subjecting the trained model to the DeepWordBug adversarial attack. This process aims to find an adversarial example for each clean training sample. Subsequently, the target model architecture is re-trained on the augmented training set, which includes the generated adversarial examples. In this augmentation-based AT model, DeepWordBug is a *seen* attack, while the other attacks are still *unseen*. It is expected that this AT model has relatively higher robustness to the *seen* DeepwordBug attack.

Evaluation. anonymous (2023)[3] demonstrated that highly overconfident models display an *illusion of robustness* (IOR), where off-the-shelf adversarial attacks struggle to identify adversarial examples for these models, causing them to appear robust. However, they showed that an adversary can perform simple post-hoc calibration to remove this illusion and enable adversarial examples to be found. Therefore, following anonymous (2023), we apply temperature calibration in our robustness evaluations to ensure any observed robustness gains are genuine and not an illusion.

Hyperparameters. We train the Transformer-*base* models (training as defined by Equation 1), using standard hyper-parameter configurations from (He et al., 2020). These include an initial learning rate of $1e-5$, a batch size of 8, a total of 5 epochs, and no warm-up steps, following TextDefender (Li et al., 2021a). It is worth noting that, despite experimentation with warm-up steps at 50 and 100, the validation accuracy remained consistent. For optimization, we use the ADAMW optimizer with a weight decay of 0.01 and specific parameters: $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e-8$. Notably, our findings suggest that global gradient normalization enhances the robustness of the *rt*, *emotion*, and *cola* datasets. However, we experienced loss explosion when using global gradient normalization for the *qnli* and *mrpc* datasets, and switched to clipped gradient normalization for the experiments on *qnli* and *mrpc*.

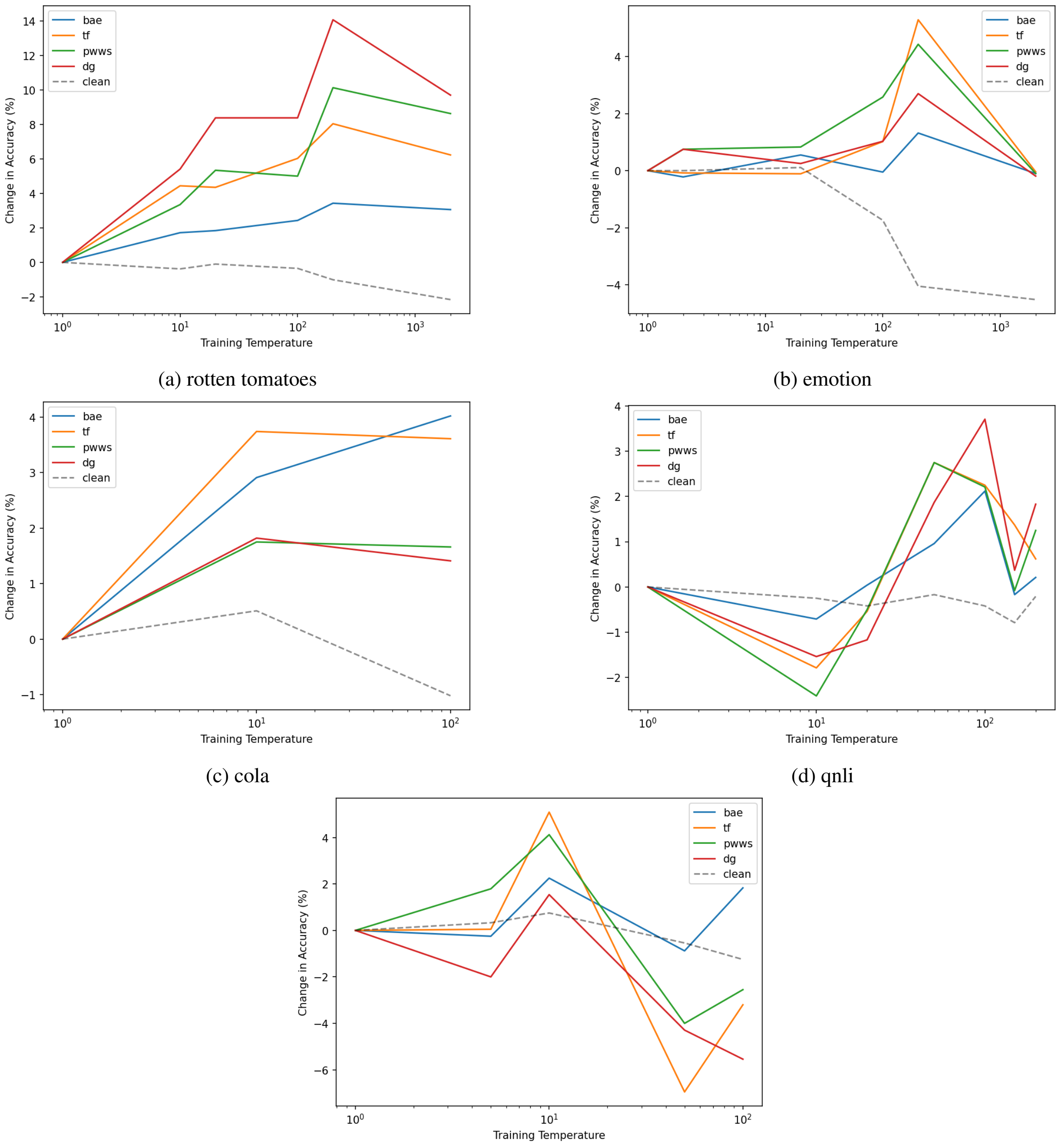
In the case of Adversarial Training (AT) methods (described by Equation 3), we maintain the same hyperparameters as the training of the ST model. Specific to AT, we adhere to hyperparameters outlined in Li et al. (2021b). The default settings for AT approaches PGD, ASCC, and FreeLB include 5 adversarial iterations, an adversarial learning rate of 0.03, an adversarial initialization magnitude of 0.05, an adversarial maximum norm of 1.0, an l2 adversarial norm type, $\alpha = 10.0$ for ASCC, and $\beta = 40.0$ for ASCC.

All experimental results in this work are reported as an average across 3 seeds.

3.2 Results

Standard Training. For each dataset, Figure 2 presents the change in clean and adversarial accuracy of a *ST* model trained as per the standard training objective (Equation 1), with different temperatures T used during training. We present the detailed breakdown of the clean and adversarial accuracies for each training temperature, for each

[3] Abstract in Appendix D.



(a) rotten tomatoes

(b) emotion

(c) cola

(d) qnli

(e) mrpc

Figure 2: The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. This increased robustness is quantified by the absolute change in adversarial accuracy compared to the baseline $T = 1$ ST model.

adversarial attack and each dataset, in Appendix A.

We first observe a general increase in adversarial accuracy (robustness) with the training temperature and then a decrease in the accuracy with extremely large training temperatures,[4] across all datasets. In some datasets (e.g., qnli and mrpc) there is a slight decrease in robustness before the sudden rise in robustness. Nevertheless, there exists a consistent robustness profile for each dataset, where robustness peaks at similar temperatures for all tested ad-

[4] The drop in robustness for extremely large training temperatures may be attributed to large temperatures excessively smoothing the predicted probability distribution during training, which makes it too challenging for the model to learn, as is reflected by the significant decrease in clean accuracy.

Method	Clean	bae	tf	pwws	dg
ST	$\underline{88.96}$ ±0.30	31.39 ±1.20	17.82 ±0.49	20.42 ±0.62	20.11 ±0.94
$\oplus T$	87.55 ±0.44	$\underline{35.83}$ ±0.84	$\underline{26.83}$ ±4.57	$\underline{31.49}$ ±3.07	$\underline{35.18}$ ±4.71
PGD	$\underline{88.59}$ ±0.64	33.71 ±0.20	17.73 ±0.86	25.20 ±1.80	25.74 ±1.46
$\oplus T$	87.77 ±0.43	$\underline{34.77}$ ±0.33	$\underline{24.55}$ ±1.76	$\underline{31.46}$ ±1.08	$\underline{31.77}$ ±2.64
FreeLB	$\underline{88.74}$ ±0.32	32.52 ±0.52	19.51 ±1.70	24.55 ±0.70	24.52 ±0.73
$\oplus T$	88.02 ±0.52	$\underline{35.15}$ ±0.80	$\underline{25.17}$ ±0.96	$\underline{29.96}$ ±0.68	$\underline{31.49}$ ±1.04
ASCC	87.77 ±0.36	33.61 ±0.64	15.13 ±2.17	23.50 ±0.77	26.80 ±2.11
$\oplus T$	86.36 ±0.80	$\underline{34.93}$ ±1.12	$\underline{27.36}$ ±0.72	$\underline{30.93}$ ±1.38	$\underline{33.46}$ ±1.65
dg-aug	$\underline{87.12}$ ±0.39	34.74 ±1.59	22.36 ±1.83	26.11 ±2.57	$\underline{37.43}$ ±0.75
$\oplus T$	87.09 ±0.22	$\underline{36.99}$ ±2.64	$\underline{26.92}$ ±2.86	$\underline{31.43}$ ±1.67	36.40 ±1.90

Table 3: Adversarial Training (AT) combined with a training temperature of $T = 200$ ($\oplus T$). For each adversarial attack, the higher adversarial accuracy between the AT model and the AT $\oplus T$ model is underlined. In almost all cases, the higher training temperature improves adversarial accuracy. Dataset: Rotten Tomatoes.

Method	Clean	bae	tf	pwws	dg
ST	$\underline{93.13}$ ±0.24	30.17 ±0.85	5.77 ±0.55	11.80 ±2.01	8.32 ±2.98
$\oplus T$	92.83 ±0.89	$\underline{32.10}$ ±0.95	$\underline{6.42}$ ±1.58	$\underline{12.68}$ ±1.20	$\underline{8.45}$ ±1.37
PGD	$\underline{93.48}$ ±0.03	28.83 ±0.43	4.88 ±1.24	9.95 ±0.69	5.45 ±1.08
$\oplus T$	93.40 ±0.10	$\underline{30.58}$ ±0.65	$\underline{5.43}$ ±0.25	$\underline{10.78}$ ±0.99	$\underline{6.33}$ ±1.51
FreeLB	93.67 ±0.23	29.15 ±1.00	4.93 ±1.25	10.15 ±0.30	$\underline{5.48}$ ±0.73
$\oplus T$	$\underline{93.72}$ ±0.10	$\underline{30.23}$ ±0.53	$\underline{5.58}$ ±0.03	$\underline{10.78}$ ±0.96	5.23 ±0.98
ASCC	91.15 ±0.57	34.65 ±0.23	4.60 ±1.05	12.15 ±0.22	11.28 ±1.40
$\oplus T$	$\underline{91.78}$ ±0.24	$\underline{34.78}$ ±0.03	$\underline{7.57}$ ±0.45	$\underline{14.08}$ ±0.64	$\underline{11.55}$ ±1.48
dg-aug	$\underline{92.58}$ ±0.11	31.52 ±2.82	4.68 ±0.25	9.33 ±0.11	$\underline{29.45}$ ±0.64
$\oplus T$	91.98 ±0.13	$\underline{31.88}$ ±0.85	$\underline{5.38}$ ±0.28	$\underline{9.40}$ ±0.87	23.63 ±1.26

Table 4: Adversarial Training (AT) combined with a training temperature of $T = 20$ ($\oplus T$). For each adversarial attack column the higher adversarial accuracy between the AT model and the AT $\oplus T$ model is underlined. In almost all cases, a higher training temperature improves adversarial accuracy. Dataset: Emotion.

versarial attack types (*bae, tf, pwws, and dg*). This is particularly useful, as a model developer, with access to only one form of adversarial attack, can tune the training temperature for optimal robustness on that specific attack form, yet be confident that the robustness gains will also transfer to the other unseen/unknown attack forms.

Temperature Selection. A further observation is that increasing temperature can lead to a small drop (between 1% and 4%) in clean accuracy. This is perhaps expected as the model can be viewed as being trained in a mode further from the optimal hyper-parameter setting. However, across all the datasets, the optimal temperature (aligned with the peak in adversarial accuracy) results in a maximal drop in clean accuracy of 1% (apart from for the emotion dataset). Given the gains in adversarial accuracy can be between 4% and 14%, this trade-off for clean accuracy can be acceptable. Further, a model developer can choose to operate at a different operating point, by selecting a training temperature that gives a smaller drop in clean accuracy (and settle for a less significant gain in the model robustness).

Adversarial Training Combination. As indicated in Table 1, beyond considering just the *ST* model's training modified with a high training temperature, we consider adversarially trained (AT) models: we explore the impact of combining the high temperature training approach with popular NLP AT methods. As outlined in Section 3.1, we consider four popular adversarial training approaches: dg-aug, which performs augmentation with deepwordbug adversarial examples; PGD and FreeLB, which augment iteratively following a gradient-based approach for the min-max formulation of AT (Equation 3) in the embedding space; and ASCC (AT approach with adversarial example augmentation in the embedding-space guided by text-space substitutions). Table 3 and Table 4 give the ST results and AT results combined with the temperature training approach on the *rt* and *emotion* datasets, respectively.

Although more significant for *rt* than *emotion*, for both datasets, combining with the high training temperature approach improves the adversarial accuracy for all adversarial attack forms (*bae*, *tf*, *pwws*, and *dg*) for the different adversarial training approaches PGD, FreeLB, and ASCC. This demonstrates that high temperature training is complementary with such adversarial training approaches and thus consistently encourages a gain in robustness. Interestingly, we observe that for *dg-aug*, high temperature training is able to consistently improve adversarial accuracy for *bae*, *tf*, and *pwws* adversarial attacks, but can cause a drop in adversarial accuracy for the *dg* attack. It should be emphasized that *dg* in this context behaves as a *seen* attack form, as the training uses augmentation with *dg* adversarial examples, whilst the other attacks (*bae*, *tf*, and *pwws*) can be considered *unseen* attack forms that the model developer has no knowledge of during training. This suggests that for augmentation-based NLP adversarial training approaches, a high training temperature does not necessarily increase robustness to *seen* attack forms, but is successful in boosting robustness to *unseen* attack forms. Overall, the high training temperature approach is effective in boosting adversarial robustness of both ST models (standard training objective of Equation 1) and AT models (adversarial training objective of Equation 3).

Transferability. It is shown that training with a high temperature leads to a consistent gain in adversarial robustness to unseen adversarial attack forms. However, an adversary may attempt to exploit attack *transferability* when looking to attack the target model trained with high temperature. To explore this notion of a transfer attack, with the *rt* dataset, Table 5 shows the impact of finding adversarial examples for the source ST model and assessing their efficacy on the target ST $\oplus$ T model. It is evident from the significant increase in the adversarial accuracy for all the attack forms (*bae*, *tf*, *pwws*, and *dg*), that a transfer attack is not able to degrade the observed robustness gains for models trained with high temperatures.

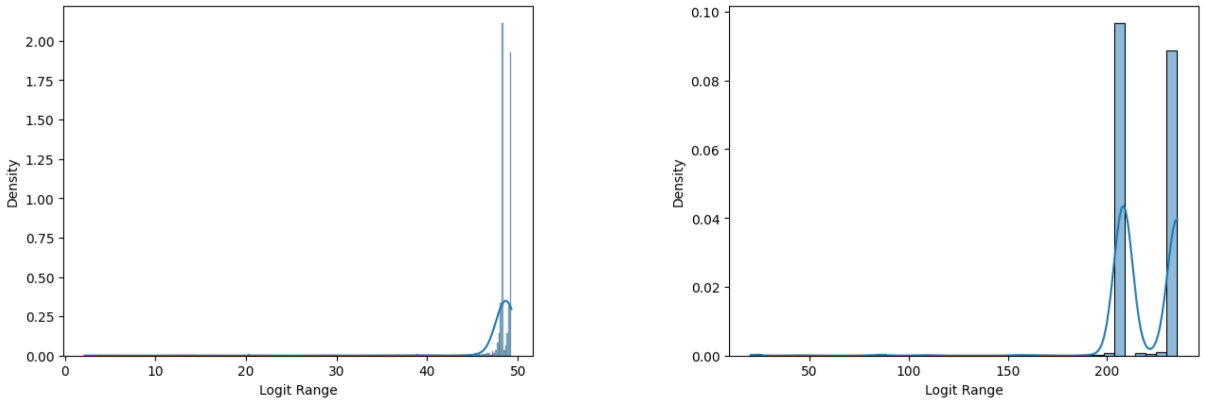
Source	Target	bae	tf	pwws	dg
ST	ST	31.39 ±1.20	17.82 ±0.49	20.42 ±0.62	20.11 ±0.94
ST $\oplus$ T	ST $\oplus$ T	35.83 ±0.84	26.83 ±4.57	31.49 ±3.07	35.18 ±4.71
ST	ST $\oplus$ T	50.13 ±0.30	46.90 ±0.38	47.53 ±1.22	46.09 ±1.13

Table 5: Transferability: adversarial examples are found for the *source* model and evaluated on the *target* model on the *rt* test set. The results here demonstrate that the standard trained, high temperature (ST $\oplus$ T) model's robustness gains relative to the ST model cannot be compromised by a transferability attack, i.e. the performance of the ST $\oplus$ T model are not degraded by adversarial examples generated from the ST model.

4 Related Work and Discussion

Adversarial Defence. In the last decade, a range of adversarial attack approaches (Alzantot et al., 2018; Garg and Ramakrishnan, 2020; Li et al., 2020; Gao et al., 2018; Wang et al., 2019c; Ren et al., 2019; Jin et al., 2019; Li et al., 2018; Tan and Joty, 2021; Tan et al., 2020a) and adversarial defence approaches have emerged for Natural Language Processing (NLP) systems. Adversarial defence approaches can be broadly classed as: detectors that identify adversarial inputs (Zhou et al., 2019; Raina and Gales, 2022; Mozes et al., 2021); form-specific *a priori* defences (Jones et al., 2020; Tan et al., 2020b); or adversarial training (Madry et al., 2018; Zhu et al., 2020; Li and Qiu, 2020; Wang et al., 2020; Dong et al., 2021; Zhou et al., 2020; Nguyen Minh and Luu, 2022), where a model is trained to explicitly encourage model robustness. Each new defence approach aims to protect models against the latest attacks, whilst new attack approaches aim to circumvent the latest defences. This has led to increasingly complex and computationally expensive defence approaches (e.g., Nguyen Minh and Luu (2022)). Further, many defences, such as the de-facto use of augmentation-based adversarial training (Jin et al., 2019; Wang and Bansal, 2018; Kang et al., 2018) or detectors tailored to specific attack forms (Zhou et al., 2019), are vulnerable to unknown/unseen attack approaches. To begin to address these challenges, we demonstrated that by simply training with a higher temperature, some model robustness can be developed against unknown attack forms, with no greater computational cost than that of standard training. To further develop this line of research, we encourage future research to also explore computationally efficient and attack-agnostic adversarial defence approaches.

Training Temperature. The use of temperature during training has not previously been discussed in literature in the context of adversarial training and model robustness. However, the temperature parameter is often exploited in other areas of research and understanding its success there can perhaps give some explanation for its success in boosting model robustness. In knowledge distillation (Hinton et al., 2015), temperature is a parameter that is often used in the softmax function during the training process (Jafari et al., 2021). Knowledge distillation is a model compression technique where a smaller model (student) is trained to replicate the behavior of a larger, more complex model (teacher). A high temperature is often used during the training of the student model. This softens the probability distribution, making the training signal more informative and providing the student model



(a) $T = 1$

(b) $T = 100$

Figure 3: Probability Density (histogram plot) of predicted class logits' range (smallest logit subtracted from largest logit value) on *rt* test set with and without a high training temperature for the ST DeBERTa model. The higher temperature training setting ($T = 100$) has a larger class logits' range, suggesting that an adversarial attack has to make a greater change in the logit space to be successful in changing the predicted class.

with additional information from the teacher model. The idea is that the teacher model's softened output helps the student model learn a smoother and more generalized decision boundary. Although the temperature parameter is not used to soften the teacher distribution in our context, Agarwala et al. (2020) find that model generalization depends strongly on the training temperature and so it is unsurprising that the temperature parameter also influences robustness. More specifically, the success of high temperature training for adversarial robustness can perhaps be explained by considering the size of the class margin (Robey et al., 2023). A high temperature smooths the probability distribution across classes, such that the probabilities of the different classes are closer together. To minimize the cross entropy loss (Equation 2), during the training, the model's parameters learn to compensate for this smoothing by pushing the logits of the different classes further apart (we see this in Figure 3, where the range of logits substantially increases with higher training temperatures). Intuitively, this can be viewed as increasing the distance to the class boundary in the logit space and thus making it more difficult for an adversarial attack to change the predicted class, giving rise to the observed increase in adversarial robustness. Future work will aim to rigorously understand and explain the observed robustness gains of training with a high temperature.

5 Conclusion

While adversarial training has proven effective in enhancing robustness, its practical implementation for NLP models poses challenges due to computational costs. Additionally, the effectiveness of the commonly employed augmentation-based adversarial training approach in NLP tends to be limited to the specific attack forms used for augmentation. To address these limitations, we propose a straightforward modification to the standard training algorithm, involving the use of a high temperature parameter to scale down predicted logits in NLP classification systems. Experiments across multiple datasets, models, and classification tasks show this simple approach boosts adversarial robustness to various unseen attack forms, and does so without incurring additional computational costs during training. Our experiments also demonstrate that our high temperature training approach is complementary with existing NLP adversarial training schemes, yielding a further increase in model robustness to unseen adversarial attacks.

6 Limitations

In our experiments, we used a constant temperature during training for simplicity. It would be interesting to consider the impact of other temperature schedules, e.g., temperature annealing (Cai, 2021) where the temperature is slowly decreased. Further, our experiments are limited to NLP classification systems and it would therefore be interesting to explore how a high training temperature influences the robustness of generative NLP models. Finally, in this work, we empirically demonstrated the usefulness of high temperature training but do not claim to formally investigate the theory behind its efficacy. A promising future direction would be to uncover the theory behind this phenomenon and potential connections to model generalization.

7 Risks & Ethics

This work presents findings on the topic of adversarial attack defence. We do not propose any new attack algorithms and instead propose a training modification to increase model robustness to adversarial attacks. Therefore, there are no perceived risks or ethical concerns associated with this work.

References

Atish Agarwala, Jeffrey Pennington, Yann N. Dauphin, and Samuel S. Schoenholz. 2020. Temperature check: theory and practice for training models with softmax-cross-entropy losses. *CoRR*, abs/2010.07344.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. pages 2890–2896.

anonymous. 2023. Extreme confidence and the illusion of robustness in adversarial training.

Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. 2021. Recent advances in adversarial training for adversarial robustness. *CoRR*, abs/2102.01356.

Zhicheng Cai. 2021. SA-GD: improved gradient descent learning strategy with simulated annealing. *CoRR*, abs/2107.07558.

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

Xinshuai Dong, Anh Tuan Luu, Rongrong Ji, and Hong Liu. 2021. Towards robustness against natural language word substitutions. *CoRR*, abs/2107.13541.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. *CoRR*, abs/1801.04354.

Siddhant Garg and Goutham Ramakrishnan. 2020. BAE: BERT-based adversarial examples for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6174–6181, Online. Association for Computational Linguistics.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced BERT with disentangled attention. *CoRR*, abs/2006.03654.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.

Aref Jafari, Mehdi Rezagholizadeh, Pranav Sharma, and Ali Ghodsi. 2021. Annealing knowledge distillation. *CoRR*, abs/2104.07163.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019. Is BERT really robust? natural language attack on text classification and entailment. *CoRR*, abs/1907.11932.

Erik Jones, Robin Jia, Aditi Raghunathan, and Percy Liang. 2020. Robust encodings: A framework for combating adversarial typos. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2752–2765, Online. Association for Computational Linguistics.

Dongyeop Kang, Tushar Khot, Ashish Sabharwal, and Eduard Hovy. 2018. AdvEntuRe: Adversarial training for textual entailment with knowledge-guided examples. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2418–2428, Melbourne, Australia. Association for Computational Linguistics.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2018. Textbugger: Generating adversarial text against real-world applications. *CoRR*, abs/1812.05271.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. pages 6193–6202.

Linyang Li and Xipeng Qiu. 2020. Textat: Adversarial training for natural language understanding with token-level perturbation. *CoRR*, abs/2004.14543.

Zongyi Li, Jianhan Xu, Jiehang Zeng, Linyang Li, Xiaoqing Zheng, Qi Zhang, Kai-Wei Chang, and Cho-Jui Hsieh. 2021a. Searching for an effective defender: Benchmarking defense against adversarial word substitution.

Zongyi Li, Jianhan Xu, Jiehang Zeng, Linyang Li, Xiaoqing Zheng, Qi Zhang, Kai-Wei Chang, and Cho-Jui Hsieh. 2021b. Searching for an effective defender: Benchmarking defense against adversarial word substitution. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3137–3147, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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 *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.

Maximilian Mozes, Pontus Stenetorp, Bennett Kleinberg, and Lewis Griffin. 2021. Frequency-guided word substitutions for detecting textual adversarial examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 171–186, Online. Association for Computational Linguistics.

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

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*.

Aleksandra Piktus, Necati Bora Edizel, Piotr Bojanowski, Edouard Grave, Rui Ferreira, and Fabrizio Silvestri. 2019. Misspelling oblivious word embeddings. 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 3226–3234, Minneapolis, Minnesota. Association for Computational Linguistics.

Vyas Raina and Mark Gales. 2022. Residue-based natural language adversarial attack detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3836–3848, Seattle, United States. Association for Computational Linguistics.

Vyas Raina, Yiting Lu, and Mark Gales. 2022. Grammatical error correction systems for automated assessment: Are they susceptible to universal adversarial attacks? In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 158–171, Online only. Association for Computational Linguistics.

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, Florence, Italy. Association for Computational Linguistics.

Alexander Robey, Fabian Latorre, George J. Pappas, Hamed Hassani, and Volkan Cevher. 2023. Adversarial training should be cast as a non-zero-sum game.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

Lu Sun, Mingtian Tan, and Zhe Zhou. 2018. A survey of practical adversarial example attacks. *Cybersecurity*, 1(1).

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks.

Samson Tan and Shafiq Joty. 2021. Code-mixing on sesame street: Dawn of the adversarial polyglots. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3596–3616, Online. Association for Computational Linguistics.

Samson Tan, Shafiq Joty, Kathy Baxter, Araz Taeihagh, Gregory A. Bennett, and Min-Yen Kan. 2021. Reliability testing for natural language processing systems. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4153–4169, Online. Association for Computational Linguistics.

Samson Tan, Shafiq Joty, Min-Yen Kan, and Richard Socher. 2020a. It's morphin' time! Combating linguistic discrimination with inflectional perturbations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2920–2935, Online. Association for Computational Linguistics.

Samson Tan, Shafiq Joty, Lav Varshney, and Min-Yen Kan. 2020b. Mind your inflections! Improving NLP for non-standard Englishes with Base-Inflection Encoding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5647–5663, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019a. Glue: A multi-task benchmark and analysis platform for natural language understanding.

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. 2020. Infobert: Improving robustness of language models from an information theoretic perspective. *CoRR*, abs/2010.02329.

William Yang Wang, Sameer Singh, and Jiwei Li. 2019b. Deep adversarial learning for NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 1–5, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Yicheng Wang and Mohit Bansal. 2018. Robust machine comprehension models via adversarial training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 575–581, New Orleans, Louisiana. Association for Computational Linguistics.

Jin Yong Yoo and Yanjun Qi. 2021. Towards improving adversarial training of NLP models. *CoRR*, abs/2109.00544.

Wei Emma Zhang, Quan Z. Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial attacks on deep-learning models in natural language processing: A survey. *ACM Trans. Intell. Syst. Technol.*, 11(3).

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*.

Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-Wei Chang, and Xuanjing Huang. 2020. Defense against adversarial attacks in NLP via dirichlet neighborhood ensemble. *CoRR*, abs/2006.11627.

Yichao Zhou, Jyun-Yu Jiang, Kai-Wei Chang, and Wei Wang. 2019. Learning to discriminate perturbations for blocking adversarial attacks in text classification. 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 4904–4913, Hong Kong, China. Association for Computational Linguistics.

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

A Detailed Performance Breakdown

Figure 2 presents the adversarial accuracy of ST models trained with different training temperatures. In this section, for reference, we provide the detailed breakdown (average across 3 seeds and standard deviation) of performances for the different training temperatures for each dataset: *rt* (Table 6), *emotion* (Table 7), *cola* (Table 8), *qnli* (Table 9), and *mrpc* (Table 10). These results are given for the DeBERTa model as in the main paper.

Temp	clean	bae	tf	pwws	dg
1	88.56 ±0.19	32.40 ±0.14	18.79 ±0.48	21.36 ±1.22	21.11 ±0.66
10	88.18 ±0.49	34.12 ±0.82	23.23 ±2.65	24.71 ±1.22	26.52 ±2.10
20	88.46 ±0.52	34.24 ±1.97	23.14 ±2.18	26.70 ±1.28	29.49 ±1.69
100	88.21 ±0.70	34.83 ±0.81	24.82 ±3.79	26.36 ±2.86	29.49 ±4.36
200	87.55 ±0.44	35.83 ±0.84	26.83 ±4.57	31.49 ±3.07	35.18 ±4.71
2000	86.40 ±0.89	35.46 ±1.79	25.02 ±0.76	29.99 ±1.38	30.81 ±1.05

Table 6: **rt:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy.

Temp	clean	bae	tf	pwws	dg
1	92.72 ±0.10	31.55 ±0.20	6.53 ±1.30	11.85 ±1.31	8.20 ±1.22
2	92.72 ±0.36	31.33 ±0.83	6.45 ±1.66	12.60 ±2.26	8.95 ±2.43
20	92.83 ±0.89	32.10 ±0.95	6.42 ±1.58	12.68 ±1.20	8.45 ±1.37
100	90.98 ±0.15	31.50 ±0.79	7.55 ±0.10	14.43 ±0.74	9.23 ±1.87
200	85.67 ±0.24	32.87 ±0.47	11.82 ±0.64	16.28 ±0.66	10.90 ±1.44

Table 7: **emotion:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy.

In Table 11, we further include results on a 6th dataset AGNews (Zhang et al., 2015), where there are four news classes, 96k training samples, 24k validation samples and 7.6k test samples. For this dataset, it can be observed that a high training temperature is not a successful method unless a fraction of the dataset (10k training samples) is used during

Temp	clean	bae	tf	pwws	dg
1	83.70 ±0.53	3.39 ±0.59	5.43 ±0.43	10.23 ±0.73	11.63 ±1.49
10	84.21 ±0.72	6.30 ±0.83	9.17 ±0.48	11.98 ±0.42	13.45 ±1.75
100	82.68 ±0.91	7.41 ±2.34	9.04 ±3.22	11.89 ±1.08	13.04 ±4.96

Table 8: **cola:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy.

Temp	clean	bae	tf	pwws	dg
1	93.17 ±0.26	35.71 ±1.88	20.71 ±3.17	19.79 ±2.38	17.92 ±4.39
10	92.92 ±0.94	35.00 ±0.66	18.92 ±1.56	17.38 ±1.02	16.38 ±2.76
20	92.75 ±0.66	35.75 ±0.38	20.17 ±1.54	19.29 ±0.90	16.75 ±1.19
50	93.00 ±0.75	36.67 ±1.39	23.46 ±1.70	22.54 ±0.26	19.79 ±1.56
100	92.75 ±0.33	37.83 ±1.19	22.96 ±0.38	22.00 ±1.44	21.63 ±1.11
150	92.38 ±0.70	35.54 ±2.00	22.08 ±2.12	19.71 ±2.32	18.29 ±3.59
200	92.96 ±0.47	35.92 ±1.21	21.33 ±3.54	21.04 ±2.48	19.75 ±3.06

Table 9: **qnli:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy.

Temp	clean	bae	tf	pwws	dg
1	87.46 ±0.26	46.42 ±1.94	38.83 ±3.92	28.63 ±3.25	33.50 ±6.11
5	87.79 ±0.44	46.17 ±3.05	38.88 ±4.58	30.42 ±2.20	31.50 ±1.11
10	88.21 ±0.31	48.67 ±3.62	43.92 ±5.63	32.75 ±3.69	35.04 ±5.03
50	86.92 ±0.29	45.54 ±4.15	31.88 ±8.15	24.63 ±5.00	29.21 ±7.22
100	86.21 ±0.94	48.25 ±2.58	35.63 ±6.39	26.08 ±5.59	27.96 ±4.74

Table 10: **mrpc:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy.

training. Future work is necessary to understand the nature of this specific dataset or other similar datasets that led to such a different behaviour for the temperature training approach.

Temp	clean	bae	tf	pwws	dg
1	93.88 ±0.22	81.50 ±0.25	29.46 ±0.19	43.00 ±2.19	39.08 ±2.89
1.5	93.75 ±0.13	80.92 ±0.51	29.08 ±3.26	42.13 ±5.20	38.58 ±3.19
2	93.92 ±0.07	80.04 ±0.56	25.00 ±3.80	40.38 ±3.19	38.54 ±5.69
20	93.83 ±0.36	79.25 ±1.02	23.50 ±2.07	37.58 ±2.60	34.58 ±4.56

Table 11: **agnews:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy.

Temp	clean	bae	tf	pwws	dg
1	93.17 ±0.38	78.00 ±0.54	32.33 ±2.32	42.08 ±0.47	40.54 ±2.53
10	92.08 ±0.40	79.00 ±0.66	38.33 ±2.89	50.42 ±2.09	46.54 ±0.63
20	92.46 ±0.19	77.92 ±0.69	38.33 ±2.63	49.21 ±2.89	45.67 ±1.61
100	92.13 ±0.38	77.50 ±0.00	30.33 ±2.81	41.46 ±0.76	40.38 ±2.34

Table 12: **agnews:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy. Training with 10k samples - 1/10th of default agnews training set size.

B Reproducing with Other Models

The main paper presents results using the DeBERTa model. Here we repeat the core experiments on other popular *base* models: BERT (Table 13) and RoBERTa (Table 14). The results here are presented for the *rt* dataset.

C Other Ablations

Augmentation based adversarial training approaches, such as dg-aug in the main paper, have twice as many training steps (due to there being double the training set size). To match the standard training setting, in Table 15 we evaluated the training with high temperature approach combined with dg-aug at half the number of training steps. Similarly, in Table 16 we consider the inverse setting, where we double the number of training iterations for the *ST* model (in standard training), as well as linearly scaling the learning rate scheduler across

Temp	clean	bae	tf	pwws	dg
1	85.08 ±0.50	30.52 ±0.76	21.01 ±0.32	21.20 ±0.34	23.14 ±2.14
10	84.79 ±0.58	32.16 ±0.66	25.88 ±1.23	23.96 ±1.89	27.48 ±1.67
100	84.76 ±0.54	33.01 ±0.78	27.12 ±1.99	25.88 ±2.45	28.92 ±2.02

Table 13: **BERT:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy. Result for *rt* dataset.

Temp	clean	bae	tf	pwws	dg
1	88.27 ±0.47	32.46 ±0.74	17.01 ±0.72	21.23 ±0.05	24.30 ±1.71
10	88.25 ±0.65	33.17 ±0.86	21.96 ±1.86	24.32 ±1.15	28.85 ±3.02
100	88.26 ±0.72	33.55 ±0.92	23.20 ±2.04	26.03 ±2.12	29.66 ±3.55

Table 14: **RoBERTa:** The use of a training temperature, T, is a simple adjustment in standard model training (ST), where the temperature parameter, T, is used to scale down predicted model logits. Higher training temperatures enhance model robustness against unseen adversarial attacks (*bae, tf, pwws, dg*) without requiring prior knowledge of these attack forms during training. Results here report the clean and adversarial accuracy. Result for *rt* dataset.

the increased number of iterations.

Method	iters	clean	pwws	dg
ST $\oplus T$	default	87.55 ±0.44	31.49 ±3.07	35.18 ±4.71
dg-aug $\oplus T$	default	87.09 ±0.22	31.43 ±1.67	36.40 ±1.90
	half	86.05 ±0.44	37.02 ±5.21	43.00 ±2.36

Table 15: Matched number of iterations for *ST* and high temperature training with dg-aug by halving the number of training steps for dg-aug.

D Illusion of Robustness

anonymous (2023) is an anonymous preprint. The abstract is provided here for reference:

Deep learning-based Natural Language Processing (NLP) models are vulnerable to adversarial attacks, where small perturbations can cause a model to misclassify. Adversarial Training (AT) is often used to increase model robustness. Despite the challenging nature of textual inputs, numerous AT approaches have emerged for NLP models. However, we have discovered an intriguing phenomenon: deliberately miscalibrating models such that they are extremely overconfident or underconfident in their predictions, disrupts adversarial attack search methods, giving rise to an illusion of robustness (IOR). This extreme miscalibration can also arise implicitly as part of existing AT schemes. However, we demonstrate that an adversary aware of this miscalibration can perform temperature calibration to modify the predicted model logits, allowing the adversarial attack search method to find adversarial examples whereby obviating IOR. Consequently, we urge adversarial robustness researchers to incorporate adversarial temperature scaling approaches into their evaluations to mitigate IOR.

Method	Epochs	clean	bae	tf	pwws	dg
ST	5	88.96 ±0.30	31.39 ±1.20	17.82 ±0.49	20.42 ±0.62	20.11 ±0.94
	10	88.34 ±0.62	33.61 ±0.52	18.76 ±0.50	22.39 ±0.61	23.45 ±0.86
$\oplus T$	5	87.55 ±0.44	35.83 ±0.84	26.83 ±4.57	31.49 ±3.07	35.18 ±4.71
	10	87.55 ±0.33	34.43 ±1.31	25.48 ±2.16	30.11 ±2.19	33.40 ±4.60
dg-aug	-	87.12 ±0.39	34.74 ±1.59	22.36 ±1.83	26.11 ±2.57	37.43 ±0.75
$\oplus T$	-	87.09 ±0.22	36.99 ±2.64	26.92 ±2.86	31.43 ±1.67	36.40 ±1.90

Table 16: Doubling training iterations for the ST model with scaled scheduler decay to match number of iterations in augmentation based AT.

E Base Class Definition with Temperature Training

```python
class BaseClassifier(nn.Module):
    def __init__(self, model_name='bert-base-uncased', num_labels=2, pretrained=True, temperature=1):
        super().__init__()
        self.model_name = model_name
        self.temperature = temperature
        if pretrained:
            self.model = AutoModelForSequenceClassification.from_pretrained(model_name, num_labels=num_labels)
            self.tokenizer = AutoTokenizer.from_pretrained(model_name)
        else:
            config = AutoConfig.from_pretrained(model_name, num_labels=num_labels) # returns config and not pretrained weights
            self.model = AutoModelForSequenceClassification.from_config(config)
            self.tokenizer = AutoTokenizer.from_pretrained(model_name)
        self.config = AutoConfig.from_pretrained(model_name, num_labels=num_labels)

    def forward(self, input_ids=None, attention_mask=None, inputs_embeds=None):
        logits = self.model(input_ids, attention_mask=attention_mask, inputs_embeds=inputs_embeds)[0]
        logits = logits / self.temperature
        return logits
```