Robust Lottery Tickets for Pre-trained Language Models

Rui Zheng[1*], **Rong Bao**[1*], **Yuhao Zhou**[1], **Di Liang**[2], **Sirui Wang**[2],
Wei Wu[2], **Tao Gui**[3†], **Qi Zhang**[1,4], **Xuanjing Huang**[1]
[1] School of Computer Science, Fudan University, Shanghai, China
[2] Meituan Inc., Beijing, China
[3] Institute of Modern Languages and Linguistics, Fudan University, Shanghai, China
[4] Shanghai Collaborative Innovation Center of Intelligent Visual Computing, Fudan University
{rzheng20, rbao18, tgui, qz,xjhuang}@fudan.edu.cn
zhouyh21@m.fudan.edu.cn

Abstract

Recent works on *Lottery Ticket Hypothesis* have shown that pre-trained language models (PLMs) contain smaller matching subnetworks (winning tickets) capable of reaching accuracy comparable to the original models. However, these tickets proved to be not robust to adversarial examples, and even worse than their PLM counterparts. To address this problem, we propose a novel method based on learning binary weight masks to identify Robust Tickets hidden in the original PLMs. Since the loss is not differentiable for the binary mask, we assign the hard concrete distribution to the masks and encourage their sparsity using a smoothing approximation of L_0 regularization. Furthermore, we design an adversarial loss objective to guide the search for Robust Tickets and ensure that the tickets perform well both in accuracy and robustness. Experimental results show the significant improvement of the proposed method over previous work on adversarial robustness evaluation.

1 Introduction

Large-scale pre-trained language models (PLMs), such as BERT (Devlin et al., 2019), Roberta (Liu et al., 2019) and T5 (Raffel et al., 2019) have achieved great success in the field of natural language processing. As more transformer layers are stacked with larger self-attention blocks, the complexity of PLMs increases rapidly. Due to over-parametrization of PLMs, some Transformer heads and even layers can be pruned without significant losses in performance (Michel et al., 2019; Kovaleva et al., 2019; Rogers et al., 2020).

The Lottery Ticket Hypothesis suggests an over-parameterized network contains certain subnetworks (i.e., winning tickets) that can match the performance of the original model when trained in isolation (Frankle and Carbin, 2019). Chen et al. (2020); Prasanna et al. (2020) also find these winning tickets exist in PLMs. Chen et al. (2020) prune BERT in an unstructured fashion and obtain winning tickets at sparsity from 40% to 90%. Prasanna et al. (2020) aim at finding structurally sparse tickets for BERT by pruning entire attention heads and MLP. Previous works mainly focused on using winning tickets to reduce model size and speed up training time (Chen et al., 2021), while little work has been done to explore more benefits, such as better adversarial robustness than the original model.

As we all know, PLMs are vulnerable to adversarial examples that are legitimately crafted by imposing imperceptible perturbations on normal examples (Jin et al., 2020; Garg and Ramakrishnan, 2020). Recent studies have shown that pruned subnetworks of PLMs are even less robust than their PLM counterparts (Xu et al., 2021; Du et al., 2021). Xu et al. (2021) observes that when fine-tuning the pruned model again, the model yields a lower robustness. Du et al. (2021) clarify the above phenomenon further: the compressed models overfit on shortcut samples and thus perform consistently less robust than the uncompressed large model on adversarial test sets.

In this work, our goal is to find robust PLM tickets that, when fine-tuned on downstream tasks, achieve matching test performance but are more robust than the original PLMs. In order to make the topology structure of tickets learnable, we assign binary masks to pre-trained weights to determine which connections need to be removed. To solve discrete optimization problem of binary masks, we assume the masks follow a hard concrete distribution (a soft version of the Bernoulli distribution), which can be solved using Gumbel-Softmax trick (Louizos et al., 2018). We then use an adversarial loss objective to guide the search for Robust Tickets and an approximate L_O regularization is used to encourage the sparsity of

[*] Equal contribution.
[†] Corresponding authors.

Robust Tickets. Robust Tickets can be used as a robust substitute of original PLMs to fine-tune downstream tasks. Experimental results show that Robust Tickets achieve a significant improvement in adversarial robustness on various tasks and maintain a matching accuracy. Our codes are publicly available at *Github*[1].

The main contributions of our work are summarized as follows:

- We demonstrate that PLMs contain Robust Tickets with matching accuracy but better robustness than the original network.
- We propose a novel and effective technique to find the Robust Tickets based on learnable binary masks rather than the traditional iterative magnitude-based pruning.
- We provide a new perspective to explain the vulnerability of PLMs on adversarial examples: some weights of PLMs do not contribute to the accuracy but may harm the robustness.

2 Related Work

2.1 Textual Adversarial Attack and Defense

Textual attacks typically generate explicit adversarial examples by replacing the components of sentences with their counterparts and maintaining a high similarity in semantics (Ren et al., 2019) or embedding space (Li et al., 2020). These adversarial attackers can be divided into character-level (Gao et al., 2018), word-level (Ren et al., 2019; Zang et al., 2020; Jin et al., 2020; Li et al., 2020) and multi-level (Li et al., 2018). In response to adversarial attackers, various adversarial defense methods are proposed to improve model robustness. Adversarial training solves a min-max robust optimization and is generally considered as one of the strongest defense methods (Madry et al., 2018; Zhu et al., 2020; Li and Qiu, 2020). Adversarial data augmentation (ADA) has been widely adopted to improve robustness by adding textual adversarial examples during training (Jin et al., 2020; Si et al., 2021). However, ADA is not sufficient to cover the entire perturbed search space, which grows exponentially with the length of the input text. Some regularization methods, such as smoothness-inducing regularization (Jiang et al., 2020) and information bottleneck regularization (Wang et al., 2021), are also beneficial for robustness. Different from the above methods, we dig a robust network from original BERT, and the subnetworks we find have better robustness through fine-tuning.

2.2 Lottery Ticket Hypothesis

Lottery Ticket Hypothesis (LTH) suggests the existence of certain sparse subnetworks (i.e., winning tickets) at initialization that can achieve almost the same test performance compared to the original model (Frankle and Carbin, 2019). In the field of NLP, previous work finds that the winning tickets also exist in Transformers and LSTM (Yu et al., 2020; Renda et al., 2020). Evci et al. (2020) propose a method to update the topology of the sparse network during training without sacrificing accuracy relative to existing dense-to-sparse training methods. (Chen et al., 2020) find that PLMs such as BERT contain winning tickets with a sparsity of 40% to 90%, and the winning tickets found in the mask language modeling task can universally be transfered to other downstream tasks. Prasanna et al. (2020) find structurally sparse winning tickets for BERT, and they notice that all subnetworks (winning tickets and randomly pruned subnetworks) have comparable performance when fine-tuned on downstream tasks. Chen et al. (2021) propose an efficient BERT training method using Early-bird lottery tickets to reduce the training time and inference time. Some recent studies have tried to dig out more features of winning tickets. Zhang et al. (2021) demonstrate that even in biased models (which focus on spurious correlations) there still exist unbiased winning tickets. Liang et al. (2021) observe that at a certain sparsity, the generalization performance of the winning tickets can not only match but also exceed that of the full model. Our work makes the first attempt to find the robust winning tickets for PLMs. (Du et al., 2021; Xu et al., 2021) show that the winning tickets that only consider accuracy are over-fitting on easy samples and generalize poorly on adversarial examples.

2.3 Robustness in Model Pruning

Learning to identify a subnetwork with high adversarial robustness is widely discussed in the field of computer vision. Post-train pruning approaches require a pre-trained model with adversarial robustness before pruning (Sehwag et al., 2019; Gui et al., 2019). In-train pruning methods integrate the pruning process into the robust learning process, which jointly optimize

[1] https://github.com/ruizheng20/Robust_Ticket

the model parameters and pruning connections (Vemparala et al., 2021; Ye et al., 2019). Sehwag et al. (20) integrate the robust training objective into the pruning process and remove the connections based on importance scores. In our work, we focus on finding Robust Tickets hidden in original PLMs rather than pruning subnetworks from a robust model.

3 The Robust Ticket Framework

In this section, we propose a novel pruning method to extract Robust Tickets of PLMs by learning binary weights masks with an adversarial loss objective. Furthermore, we articulate the robust lottery ticket hypothesis: the full PLM contains subnetworks (Robust Tickets) that can achieve better adversarial robustness and comparable accuracy.

3.1 Revisiting Lottery Ticket Hypothesis

Denote $f(\theta)$ as a PLM with parameters θ that has been fine-tuned on a downstream task. A subnetwork of $f(\theta)$ can be denoted as $f(m \odot \theta)$, where m are binary masks with the same dimension as θ and $\odot$ is the Hadamard product operator. LTH suggests that, for a network initialized with θ_0, the Iterative Magnitude Pruning (IMP) can identify a mask m, such that the subnetwork $f(x; m \odot \theta_0)$ can be trained to almost the same performance to the full model $f(\theta_0)$ in a comparable number of iterations. Such a subnetwork $f(x; m \odot \theta_0)$ is called as *winning tickets*, including both the structure mask m and initialization θ_0. IMP iteratively removes the weights with the smallest magnitudes from $m \odot \theta$ until a certain sparsity is reached. However, the magnitude-based pruning is not suitable for robustness-aware techniques (Vemparala et al., 2021; Sehwag et al., 20).

3.2 Discovering Robust Tickets

Our goal is to learn the sparse subnetwork, however, the training loss is not differentiable for the binary masks. A simple choice is to adopt a straight-through estimator to approximate the derivative (Bengio et al., 2013). Unfortunately, this approach ignores the Heaviside function in the likelihood and results in biased gradients.

In our method, we assume each mask m_i to be a independent random variable that follows a hard concrete distribution $\mathrm{HardConcrete}(\log \alpha_i, \beta_i)$ with temperature β_i and location α_i (Louizos et al., 2018):

$$\mu_i \sim \mathcal{U}\left(0,1\right), \tag{1}$$

$$s_i = \sigma\left(\frac{1}{\beta_i}\left(\log\frac{\mu_i}{1-\mu_i} + \log\alpha_i\right)\right), \tag{2}$$

$$m_i = \min\left(1, \max\left(0, s_i\left(\zeta - \gamma\right) + \gamma\right)\right), \tag{3}$$

where σ denotes the sigmoid, $\gamma = -0.1$, $\zeta = 1.1$ are constants, and u_i is the sample drawn from uniform distribution $\mathcal{U}(0, 1)$. The random variable s_i follows a binary concrete (or Gumbel-Softmax) distribution, which is a smoothing approximation of the discrete Bernoulli distribution (Maddison et al., 2017; Jang et al., 2017). Samples from the binary concrete distribution are identical to samples from a Bernoulli distribution with probability α_i as $\beta_i \to 0$. The location α_i in (2) allows for gradient-based optimization through reparametrization tricks. Using (3), the s_i larger than $\frac{1-\gamma}{\zeta-\gamma}$ is rounded to 1, whereas the value smaller than $\frac{-\gamma}{\zeta-\gamma}$ is rounded to 0. To encourage the sparsity, we penalize the L_0 complexity of masks based on the probability which are non-zero:

$$\mathcal{R}(m) = \frac{1}{|m|}\sum_{i=1}^{|m|}\sigma\left(\log\alpha_i - \beta_i \log\frac{-\gamma}{\zeta}\right). \tag{4}$$

During the inference stage, the mask $\hat{m}_i$ can be estimated through a hard concrete gate:

$$\min\left(1, \max\left(0, \sigma\left(\log\alpha_i\right)\left(\zeta - \gamma\right) + \gamma\right)\right). \tag{5}$$

3.2.1 Adversarial Loss Objective

To find the connections responsible for adversarial robustness, we incorporate the adversarial loss into the training objective of masks:

$$\min_{m} \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}} \max_{\|\delta\|\le\epsilon} \mathcal{L}\left(f(x+\delta; m\odot\theta), y\right)}_{\mathcal{L}_{adv}(m)}, \tag{6}$$

where (x, y) is a data point from dataset $\mathcal{D}$, δ is the perturbation that constrained within the ϵ ball. The inner maximization problem in (6) is to find the worst-case adversarial examples to maximize the classification loss, while the outer minimization problem in (6) aims at optimizing the masks to minimize the loss of adversarial examples, i.e., $\mathcal{L}_{adv}(m)$.

Adversarial attack method, typically with PGD, can be used to solve the inner maximization problem. PGD applies the K-step stochastic gradient descent to search for the perturbation δ (Madry et al., 2018):

$$\delta_{k+1} = \prod_{\|\delta\| \le \epsilon} \left(\delta_k + \eta \frac{g(\delta_k)}{\|g(\delta_k)\|} \right), \tag{7}$$

where $g(\delta_k) = \nabla_x \mathcal{L}(f(x + \delta_k; m \odot \theta), y)$, δ_k is the perturbation in k-th step and $\prod_{\|\delta\| \le \epsilon}(\cdot)$ projects the perturbation back onto the Frobenius normalization ball. Then robust training optimizes the network on adversarially perturbed input $x + \delta_K$. Through the above process, we can conveniently obtain a large number of adversarial examples for training.

By integrating the L_0 complexity regularizer into the training process of masks, our adversarial loss objective becomes:

$$\min_m \mathcal{L}_{adv}(m) + \mathcal{R}(m), \tag{8}$$

where λ denotes regularization strength.

3.2.2 Effect of Regularization Strength

The selection of the regularization strength λ decides the quality of Robust Tickets. Results carried on SST-2 in Fig.1 show that eventually more than 90% of the masks will be very close to 0 or 1, and the L_0 complexity regularizer $\mathcal{R}(m)$ will converge to a fixed value. As λ increases, $\mathcal{R}(m)$ decreases (the sparsity of subnetwork increases). In practice, the percentage of binary masks is a promising indicator to choose a suitable λ. This is because we can prune subnetworks at arbitrary sparsity based on the confidence of masks (we show it in the next section). The training of the adversarial loss objective in (8) is insensitive to the λ, and in all experiments, λ is chosen in the range $[0.1, 1]$. In the appendix A, we will show more about the learning process of masks.

3.3 Drawing and Retraining Winning Tickets

After training the masks m, we use the location parameters $\log \alpha$ of masks to extract robust subnetworks. For the Gumbel-Softmax distribution in (2), α_i is the expectation of random variable s_i, i.e, $\mathbb{E}\{s_i\} = \alpha_i$. Thus, we prune the weights whose masks have the smallest expectation. We prune all attention heads and intermediate neurons in an unstructured manner, which empirically has better performance than structured pruning. Unlike the Lottery Ticket Hypothesis that requires iterative magnitude pruning, the proposed method is a one-shot pruning method that can obtain subnetworks of any sparsity. Then we fine-tune the Robust Tickets $f(m \odot \theta_0)$ on downstream tasks.

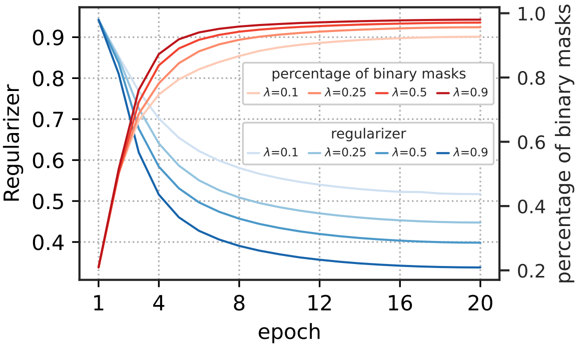


Figure 1: Effect of regularization strength λ on regularizer $\mathcal{R}(m)$, and the percentage of masks that exact 0 and 1.

3.4 Robust Lottery Tickets Hypothesis

In the context of adversarial robustness, we seek a winning ticket that balances accuracy and robustness, which is more challenging.

Robust Lottery Tickets Hypothesis: A pre-trained language model, such as BERT, contains some subnetworks (Robust Tickets) initialized by pre-trained weights, and when these subnetworks are trained in isolation, they can achieve better adversarial robustness and comparable accuracy. In addition, robust tickets retains an important characteristic of traditional lottery tickets —the ability to speed up the training process.

4 Experiments

Following the official BERT implementation (Devlin et al., 2019; Wolf et al., 2020), we use BERT$_{\text{BASE}}$ as our backbone model for all experiments. We fine-tune the original BERT and Robust Tickets using the default settings on downstream tasks, see appendix B.1 for details. We implement all models in MindSpore.

4.1 Datasets

We experiments on three text classification datasets: Internet Movie Database (IMDB, Maas et al., 2011) , AG News corpus (AGNEWS, Zhang et al., 2015) and Stanford Sentiment Treebank of binary classification (SST-2, Socher et al., 2013). We also test our method on other types of tasks in GLUE, such as MNLI, QNLI, QQP.

4.2 Baseline Methods

The proposed method is primarily compared with recently proposed adversarial defense methods and the standard LTH.

FreeLB(Zhu et al., 2020) An enhanced gradient-based adversarial training method which is not targeted at specific attack methods. **InfoBERT**(Wang et al., 2021) A learning framework for robust model fine-tuning from an information-theoretic perspective. This method claims that it has obtained a better representation of data features. **LTH**(Chen et al., 2020) For a range of downstream tasks, they find BERT contains winning tickets (matching subnetworks) at 40% to 90% sparsity. **RobustT** The **Robust T**ickets selected from the original BERT by our proposed method. **Random** The subnetworks with the same layer-wise sparsity of the above Robust Tickets, but their structures are randomly pruned from the original BERT.

4.3 Robust Evaluation

Three widely accepted attack methods are used to verify the ability of our proposed method against baselines (Li et al., 2021). The specific parameters of these attackers can be found in appendix B.2. **BERT-Attack** (Li et al., 2020) is a method using BERT to generate adversarial text, and thus the generated adversarial examples are fluent and semantically preserved. **TextFooler** (Jin et al., 2020) first identify the important words in the sentences, and then replace them with synonyms that are semantically similar and grammatically correct until the prediction changes. **TextBugger** (Li et al., 20 1) is an adversarial attack method that generates misspelled words by using character-level and word-level perturbations.

The evaluation metrics adopted in our experimental analyses are listed as follows: **Clean accuracy (Clean**$\%$**)** denotes the accuracy on the clean test dataset. **Accuracy under attack (Aua**$\%$**)** refers to the model's prediction accuracy facing specific adversarial attacks. **Attack success rate (Suc**$\%$**)** is the ratio of the number of texts successfully perturbed by an attack method to the total number of texts to be attempted. **Number of Queries (#Query)** is the average number of times the attacker queries the model, which means the more the average query number is, the harder the defense model is to be compromised.

For a robust method, higher clean accuracy, accuracy under attack, and query times are expected, as well as lower attack success rate.

4.4 Implementation Details

The K-step PGD requires K forward-backward passes through the network, which is time consuming. Thus, we turn to FreeLB, which accumulates gradients in multiple forward passes and then passing gradients backward once. For our approach, we prune robust networks in the range of 10$\%$ and 90$\%$ sparsity and report the best one in terms of robustness in our main experiments. For a fair comparison, the sparsity of LTH is the same as that of Robust Tickets. All experimental results are the average of 5 trials with different seeds. The hyperparameters of our proposed methods are listed in the appendix B.3.

4.5 Main Results on Robustness Evaluation

Table 1 shows the results of Robust Tickets and other baselines under adversarial attack. We can observe that: 1) original BERT and BERT-tickets fail to perform well on adversarial robustness evaluation, and the BERT-tickets even show lower robustness than BERT, indicating that it is difficult for the pruned subnetworks to fight against adversarial attacks when only test accuracy is considered. This result is consistent with the results in (Du et al., 2021; Xu et al., 2021). 2) The proposed Robust Ticket achieves a significant improvement of robustness over the original BERT and other adversarial defense methods. Robust Tickets use a better robust structure to resist adversarial attacks, which is different from the previous methods aimed at solving robust optimization problems. 3) In both AGNEWS and IMDB, the randomly pruned subnetwork loses only about 1 performance point in test accuracy, but performs poorly in adversarial robustness. Previous results suggest that all BERT tickets may be test accuracy winners, but our results show that only a small percentage of tickets are winners in terms of adversarial robustness. 4) Robust Tickets sacrifice accuracy performance in SST-2 and IMDB. We speculate that this may be due to the trade-off between accuracy and robustness (Tsipras et al., 2019).

We also evaluate the performance of our proposed method on more tasks. From Table 2, we can see that our proposed method yields significant improvements of robustness over the original BERT on QNLI, MNLI and QQP datasets. There is a significant improvement even compared with FreeLB.

Dataset	Method	Clean%	BERT-Attack			TextFooler			TextBugger		
			Aua%	Suc%	#Query	Aua%	Suc%	#Query	Aua%	Suc%	#Query
IMDB	Fine-tune	94.1	7.8	91.7	1572.2	12.2	87.0	1209.8	25.8	72.5	783.2
	LTH$_{20\%}$	94.0	3.6	96.2	1074.44	7.2	92.3	894.1	16.0	83.0	574.0
	FreeLB	94.8	22.6	76.2	1954.7	27.2	71.3	1479.1	36.0	62.0	907.3
	InfoBERT	**95.2**	26.0	72.7	2326.0	32.4	66.0	1572.2	43.6	54.2	969.8
	Rand$_{20\%}$	93.1	6.8	92.8	731.5	7.4	92.1	598.7	8.4	91.9	464.3
	RobustT$_{20\%}$	93.8	**55.2**	**41.2**	**3128.0**	**55.6**	**40.7**	**1988.4**	**57.6**	**38.6**	**1149.1**
AGNEWS	Fine-tune	94.7	3.8	96.0	436.7	14.9	84.2	333.2	41.5	56.1	178.3
	LTH$_{40\%}$	93.7	2.5	97.3	394.4	11.0	88.3	295.2	36.8	60.7	179.7
	FreeLB	**95.2**	10.8	88.6	563.9	24.3	74.4	394.6	51.7	45.5	190.4
	InfoBERT	94.4	11.1	88.3	517.0	25.1	73.4	374.7	47.9	49.3	193.1
	Rand$_{40\%}$	94.0	1.3	98.6	357.2	6.3	93.2	275.1	27.5	70.1	148.7
	RobustT$_{40\%}$	94.9	**12.1**	**87.2**	**607.7**	**28.5**	**70.0**	**442.1**	**53.4**	**43.7**	**207.8**
SST-2	Fine-tune	92.0	2.9	96.8	114.2	5.0	94.6	98.4	29.4	68.3	49.7
	LTH$_{60\%}$	92.1	2.2	97.6	98.9	4.1	95.5	90.5	29.1	68.4	49.6
	FreeLB	91.6	10.2	88.9	154.6	14.4	84.2	123.8	**42.4**	**53.7**	**54.9**
	InfoBERT	**92.1**	14.4	84.4	162.3	18.3	80.1	121.4	40.3	56.3	51.2
	Rand$_{30\%}$	83.2	2.1	97.5	89.4	2.4	97.1	75.6	16.5	80.2	44.2
	RobustT$_{30\%}$	90.9	**17.9**	**80.3**	**164.9**	**26.7**	**70.6**	**149.8**	42.1	53.7	53.9

Table 1: Main results on adversarial robustness evaluation. Fine-tuning **RobustT** for downstream tasks achieves a significant improvement of robustness. The percentage on the subscript denotes the sparsity of the subnetworks. The best performance is marked in bold. **Suc**% lower is better.

Dataset	Method	Clean%	Aua%	
			TextFooler	TextBugger
QNLI	Fine-tune	**91.6**	4.7	10.5
	FreeLB	90.5	12.8	12.0
	InfoBERT	91.5	16.4	20.9
	RobustT$_{30\%}$	91.5	**17.0**	**25.9**
MNLI	Fine-tune	**84.4**	7.7	4.3
	FreeLB	82.9	11.0	8.4
	InfoBERT	84.1	10.8	8.4
	RobustT$_{30\%}$	84.0	**18.4**	**22.6**
QQP	Fine-tune	91.3	24.8	27.8
	FreeLB	91.2	27.4	28.1
	InfoBERT	**91.9**	34.4	35.9
	RobustT$_{30\%}$	91.5	**47.2**	**46.0**

Table 2: Performance of RobustT on QNLI, MNLI and QQP datasets. Compared with the original BERT, fine-tuning on Robust Tickets improves the adversarial robustness on different tasks.

Dataset	Method	Clean%	Aua%
IMDB	**RobustT**$_{20\%}$	93.8	**55.6**
	w/o Mask Training	**94.0**	15.1
	w/o Adv	93.4	5.4
AGNEWS	**RobustT**$_{40\%}$	**94.9**	**28.5**
	w/o Mask Training	94.2	16.1
	w/o Adv	94.5	8.8
SST-2	**RobustT**$_{30\%}$	90.9	**26.7**
	w/o Mask Training	**92.2**	6.2
	w/o Adv	91.2	3.5

Table 3: Ablation study on text classification datasets. **Aua**% is obtained after using TextFooler attack.

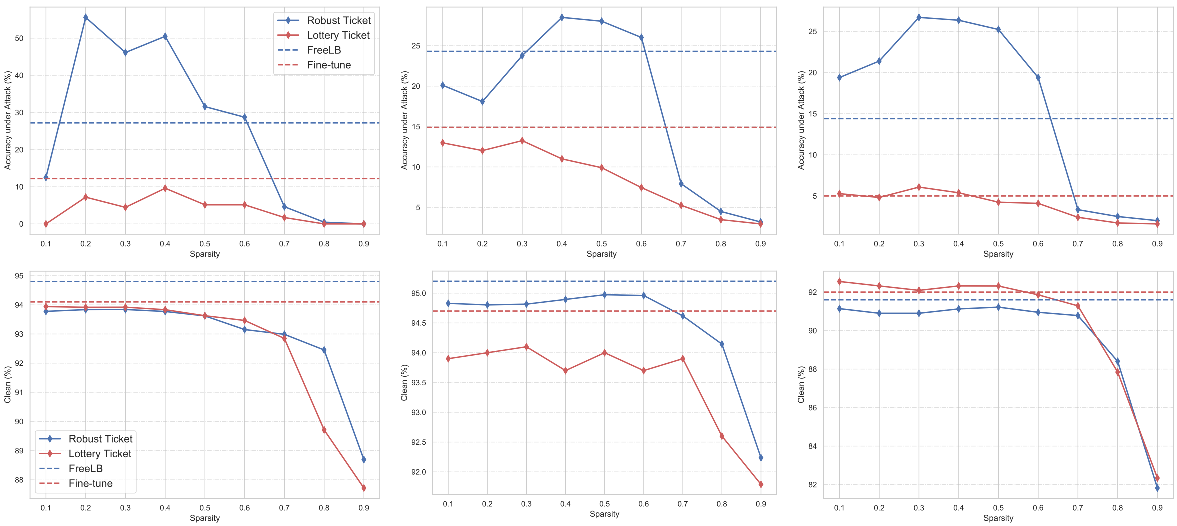
4.6 Ablation Study

To better illustrate the contribution of each component of the proposed method, we perform the ablation study by removing the following components: mask training (but replacing it with IMP in traditional LTH), adversarial loss objective (Adv). We can observe that: 1) Mask training is important for performance and imp does not identify robust subnetworks well. 2) Without adversarial loss objective, the proposed method identifies subnetworks that perform well in terms of clean accuracy, but does not provide any improvement in terms of robustness.

5 Discussion

5.1 Impact of Sparsity on Robust Tickets

The proposed method can prune out a subnetwork with arbitrary sparsity based on the confidence of masks. In Fig.2, we compare the Robust Tickets and randomly pruned subnetwork across all sparsities. Robust Tickets have better robustness even at low sparsity, which confirms that some structures of BERT are useless for accuracy and



(a) IMDB (b) AGNEWS (c) SST-2

Figure 2: Fine-tuning evaluation results of the robust winning tickets (blue), the random (orange), and the original BERT (green) under various sparsity levels. The adversarial robustness improves as the compression ratio grows until a certain threshold, then the robustness deteriorates. **Aua**% is obtained after using TextFooler attack.

hurt robustness. When the sparsity increases to a certain level, the robustness decreases faster than the accuracy, which indicates that the robustness is more likely to be affected by the model structure than the accuracy. Therefore, it is more difficult to find a Robust Ticket from BERT. The accuracy of the subnetwork is slowly decreasing with increasing sparsity, but the robustness shows a different trend. The change in robustness can be roughly divided into three phases: The robustness improves as the sparsity grows until a certain threshold; beyond this threshold, the robustness deteriorates but is still better than that of the random tickets, In the end, when being highly compressed, the robust network collapses into a random network. A similar phenomenon is also be observed (Liang et al., 2021). The robustness performance curve is not as smooth as the accuracy, this may be due to the gap between the adversarial loss objective and the real textual attacks.

Dataset	**Method**	**Clean**%	**Aua**%
IMDB	**RobustT**$_{20\%}$	93.7	55.6
	w/o Initialization	87.9	0.2
	w/o Structure	93.7	13.4
	w/o Structure+Longer	93.6	18.6
AGNEWS	**RobustT**$_{40\%}$	94.9	28.5
	w/o Initialization	92.4	0.4
	w/o Structure	94.9	21.8
	w/o Structure+Longer	94.8	24.6
SST-2	**RobustT**$_{30\%}$	90.9	26.7
	w/o Initialization	83.1	2.1
	w/o Structure	**92.0**	15.7
	w/o Structure+Longer	91.9	**25.5**

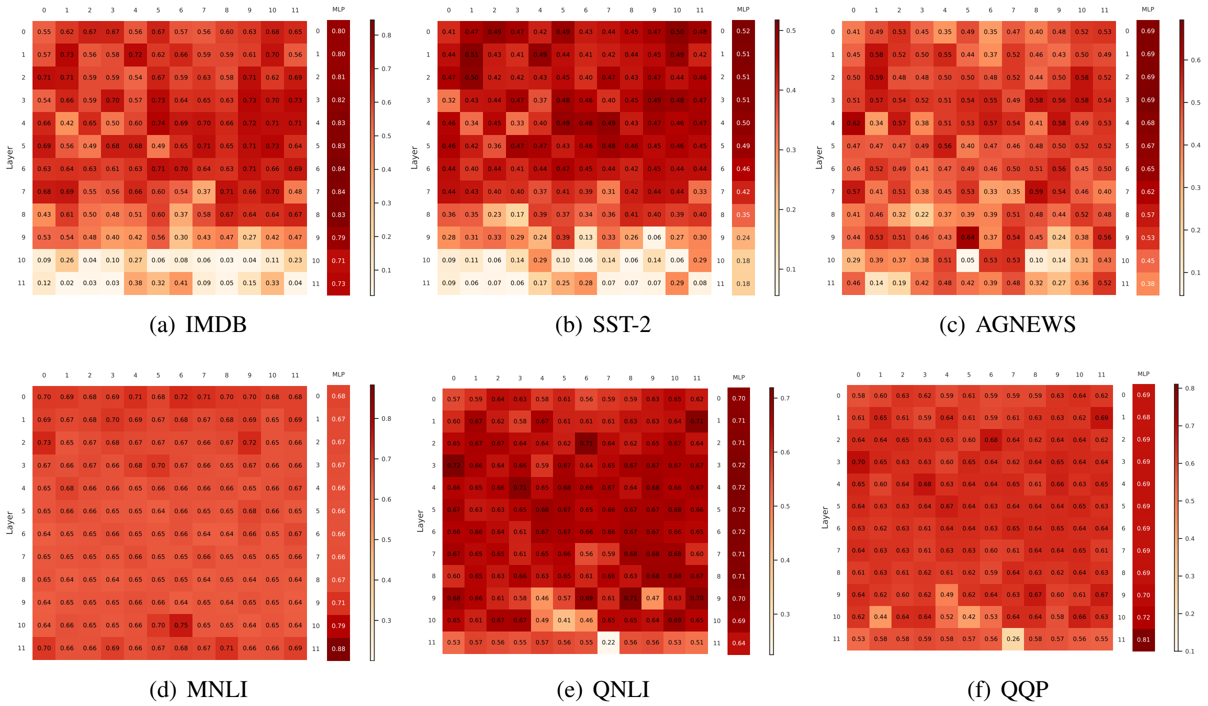
Table 4: Importance of Robust Ticket initialization and structure. Our results show that the initialization of Robust Tickets seems to be more important than the structure, although both of them play a role. **Aua**% is obtained after using TextFooler attack.

5.2 Sparsity Pattern

Fig.3 shows the sparsity patterns of Robust Tickets on all datasets. We can clearly find that the pruning rate increases from bottom to top on the text classification tasks (IMDB, SST2, AGNEWS), while it is more uniform in the natural language inference tasks (MNLI and QNLI) and Quora question pairs (QQP). Recent work shows that BERT encodes a rich hierarchy of linguistic information. Taking the advantage of the probing task, Jawahar et al. (2019) indicates that the surface information features are encoded at the bottom, syntactic information features are in the middle network, and semantic information features in the top. Therefore, we speculate that the sparsity pattern of Robust Tickets is task-dependent.

5.3 Speedup Training Process

An important property of winning tickets is to accelerate the convergence of the training process



(a) IMDB (b) SST-2 (c) AGNEWS

(d) MNLI (e) QNLI (f) QQP

Figure 3: Heatmaps of sparsity patterns found on different tasks, each cell gives the percentage of surviving weights in self-attention heads and MLPs. The sparsity patterns on IMDB and SST-2 are similar, which may be due to the fact that they are both text classification datasets based on movie reviews.

(Chen et al., 2021; You et al., 2019). The training curve in Fig.4 shows that the convergence speed of Robust Tickets is much faster compared with the default fine-tuning methods. And both the convergence of both accuracy and robustness are accelerated. In addition, the convergence of the randomly pruned subnetwork is not accelerated, which points out that sparse structures and smaller models do not always lead to a faster training.

5.4 The Importance of Robust Ticket Initialization and Structure

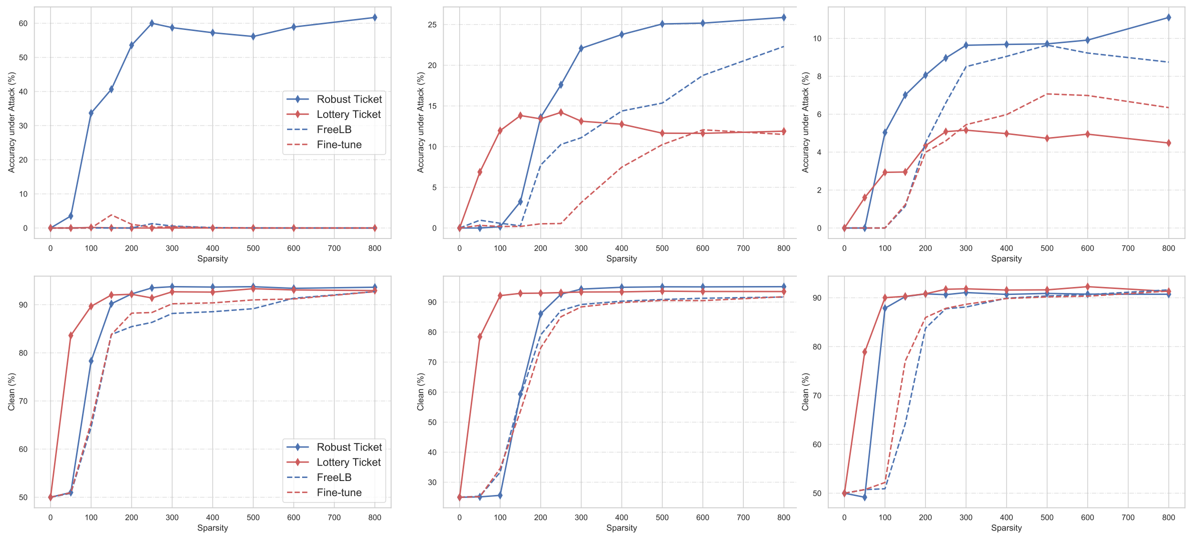
To better understand which factor, initialization or structure, has a greater impact on the Robust Ticket, we conduct corresponding analysis studies. We avoid the effect of initialization by re-initializing the weights of robust tickets. To avoid the effect of structures and preserve the effect of initializations, we use the full BERT and re-initialize the weights that are not contained in the Robust Tickets. The results are shown in Table 4.

Importance of initialization LTH suggests that the winning tickets can not be learned effectively without its original initialization. For our robust BERT tickets, their initializations are pre-trained weights. Table 4 shows the failure of Robust Tickets when the random re-initialization is performed.

Importance of structure Frankle and Carbin (2019) hypothesize that the structure of winning tickets encodes an inductive bias customized for the learning task at hand. Although removing this inductive bias reduces performance compared to the robust tickets, it still outperforms the original BERT, and this performance improves further with longer training time. It can be seen that the initializations of some pre-training weights may lead to a decrease in the robustness of the model.

6 Conclusion

In this paper, we articulate and demonstrate the Robust Lottery Ticket Hypothesis for PLMs: the full PLM contains a subnetwork (Robust Ticket) that can achieve a better robustness performance. We propose an effective method to solve the ticket selection problem by encouraging weights that are not responsible for robustness to become exactly zero. Experiments on various tasks corroborate the effectiveness of our method. We also find that pre-trained weights may be a key factor affecting the robustness on downstream tasks. The Robust Tickets are good defenders.



(a) IMDB (b) AGNEWS (c) SST-2

Figure 4: Clean accuracy and accuracy under attack as training proceeds. The method proposed by us is less time-consuming during the training process of both two metrics. **Aua**% is obtained after using TextFooler attack.

Acknowledgements

The authors wish to thank the anonymous reviewers for their helpful comments. This work was partially funded by National Natural Science Foundation of China (No. 62076069, 61976056). This research was supported by Meituan, Beijing Academy of Artificial Intelligence(BAAI), and CAAI-Huawei MindSpore Open Fund.

References

Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *ArXiv preprint*, abs/1308.3432.

Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. 2020. The lottery ticket hypothesis for pre-trained BERT networks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Xiaohan Chen, Yu Cheng, Shuohang Wang, Zhe Gan, Zhangyang Wang, and Jingjing Liu. 2021. Early-BERT: Efficient BERT training via early-bird lottery tickets. 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 2195–2207, Online. Association for Computational Linguistics.

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

Mengnan Du, Subhabrata Mukherjee, Yu Cheng, Milad Shokouhi, Xia Hu, and Ahmed Hassan Awadallah. 2021. What do compressed large language models forget? robustness challenges in model compression. *ArXiv preprint*, abs/2110.08419.

Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. 2020. Rigging the lottery: Making all tickets winners. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 2943–2952. PMLR.

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

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

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.

Shupeng Gui, Haotao Wang, Haichuan Yang, Chen Yu, Zhangyang Wang, and Ji Liu. 2019. Model compression with adversarial robustness: A unified optimization framework. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1283–1294.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

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. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8018–8025. AAAI Press.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of BERT. 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 4365–4374, Hong Kong, China. Association for Computational Linguistics.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 20 1. TextBugger: Generating Adversarial Text Against Real-world Applications. *ArXiv preprint*, abs/ 1812.

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

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6193–6202, Online. Association for Computational Linguistics.

Linyang Li and Xipeng Qiu. 2020. Tavat: Token-aware virtual adversarial training for language understanding. *ArXiv preprint*, abs/2004.14543.

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

Chen Liang, Simiao Zuo, Minshuo Chen, Haoming Jiang, Xiaodong Liu, Pengcheng He, Tuo Zhao, and Weizhu Chen. 2021. Super tickets in pre-trained language models: From model compression to improving generalization. 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 6524–6538, Online. 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. *ArXiv preprint*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Christos Louizos, Max Welling, and Diederik P. Kingma. 2018. Learning sparse neural networks through l_0 regularization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2017. The concrete distribution: A continuous relaxation of discrete random variables. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14014–14024.

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, Online. Association for Computational Linguistics.

Sai Prasanna, Anna Rogers, and Anna Rumshisky. 2020. When BERT Plays the Lottery, All Tickets Are Winning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3208–3229, Online. Association for Computational Linguistics.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv preprint*, abs/1910.10683.

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.

Alex Renda, Jonathan Frankle, and Michael Carbin. 2020. Comparing rewinding and fine-tuning in neural network pruning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Vikash Sehwag, Shiqi Wang, Prateek Mittal, and Suman Sekhar Jana. 20. Hydra: Pruning adversarially robust neural networks. *ArXiv preprint*, abs/.

Vikash Sehwag, Shiqi Wang, Prateek Mittal, and Suman Sekhar Jana. 2019. Towards compact and robust deep neural networks. *ArXiv preprint*, abs/1906.06110.

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, Online. Association for Computational Linguistics.

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, Seattle, Washington, USA. Association for Computational Linguistics.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2019. Robustness may be at odds with accuracy. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Manoj Rohit Vemparala, Nael Fasfous, Alexander Frickenstein, Sreetama Sarkar, Qi Zhao, Sabine Kuhn, Lukas Frickenstein, Anmol Singh, Christian Unger, Naveen Shankar Nagaraja, Christian Wressnegger, and Walter Stechele. 2021. Adversarial robust model compression using in-train pruning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 66–75.

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. 2021. INFOBERT: IMPROVING ROBUSTNESS OF LANGUAGE MODELS FROM AN INFORMATION THEORETIC PERSPECTIVE. page 23.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Canwen Xu, Wangchunshu Zhou, Tao Ge, Ke Xu, Julian McAuley, and Furu Wei. 2021. Beyond preserved accuracy: Evaluating loyalty and robustness of bert compression. In *EMNLP*.

Shaokai Ye, Xue Lin, Kaidi Xu, Sijia Liu, Hao Cheng, Jan-Henrik Lambrechts, Huan Zhang, Aojun Zhou, Kaisheng Ma, and Yanzhi Wang. 2019. Adversarial robustness vs. model compression, or both? In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 111–120. IEEE.

Haoran You, Chaojian Li, Pengfei Xu, Y. Fu, Yue Wang, Xiaohan Chen, Yingyan Lin, Zhangyang Wang, and

Richard Baraniuk. 2019. Drawing early-bird tickets: Towards more efficient training of deep networks. *ArXiv preprint*, abs/1909.11957.

Haonan Yu, Sergey Edunov, Yuandong Tian, and Ari S. Morcos. 2020. Playing the lottery with rewards and multiple languages: lottery tickets in RL and NLP. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080, Online. Association for Computational Linguistics.

Dinghuai Zhang, Kartik Ahuja, Yilun Xu, Yisen Wang, and Aaron C. Courville. 2021. Can subnetwork structure be the key to out-of-distribution generalization? In *ICML*.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, 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 *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

A The Effect of Regularization Strength during Mask Training

In section 3.2.2, we show the mask training curves for various regularization strengths λ in SST-2 dataset. The results on more datasets are shown in the Fig.5, where we can observe that the mask training process is insensitive to the regularization strength, and the convergence of masks is eventually achieved.

B Implementation Details

B.1 Details for finetuning models

We report in Table 5 the values of hyperparameters used to fine-tune the models. All the models were trained on single Nvidia Ge-Force RTX-3090 Graphical Card with 24G graphical memory.

Hypeparameters	**Values**
Optimizer	Adamw(Loshchilov and Hutter, 2019)
Learning rate	2×10^{-5}
Dropout	0.1
Weight decay	1×10^{-2}
Batch size	16 or 32
Gradient clip	$(-1, 1)$
Epochs	3
Bias-correction	True

Table 5: Training hyperparameters for fine-tuning the models.

B.2 Details for text attack

We use textattack (Morris et al., 2020) to implement the following attack methods. For most attack methods, we use the default parameters of third-party libraries. It should be noted that the implementation in textattack also takes special attack parameters, so the attack performance of each method can not be identical in their original papers. The parameters for the various attack methods are defined as follows: neighbour size N, modify ratio M, sentence similarity S. They are listed in the Table 6.

	Textfooler	**Textbugger**	**BERT-Attack**
N	50	5	50
M	–	–	0.9
S	0.84	0.8	0.2

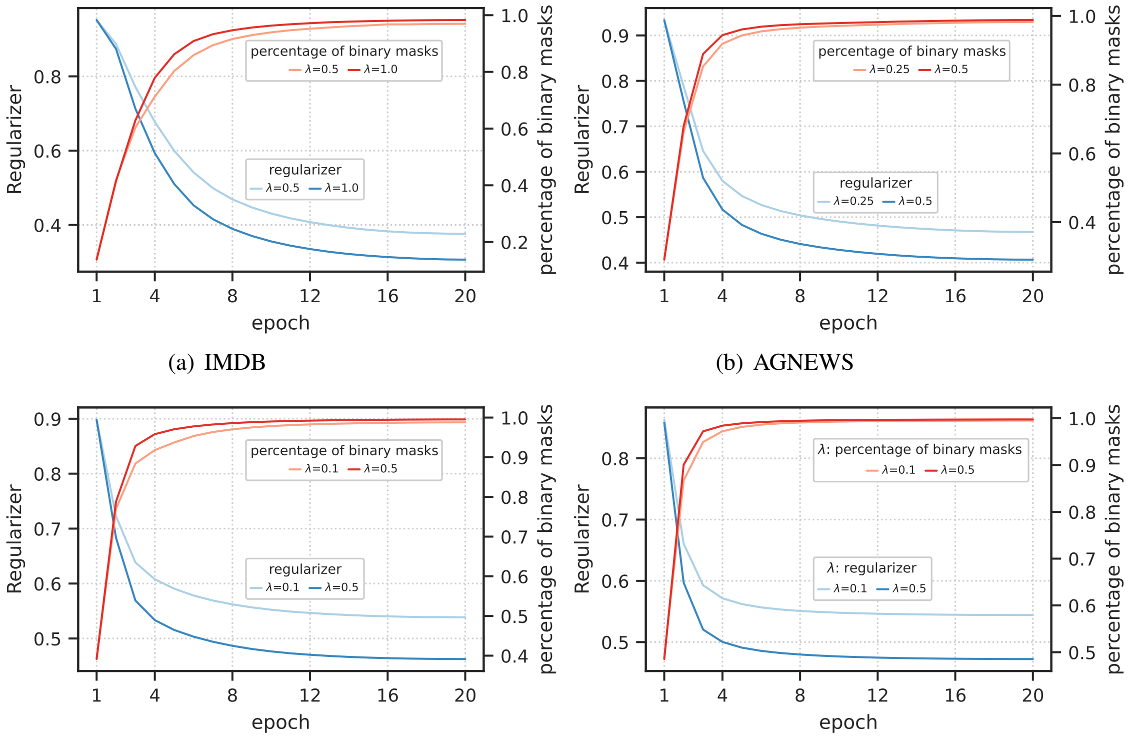
Table 6: Setting of attack parameters for attack methods.

Datasets	K	β	λ	ϵ	m	w
SST2	0.03	0.05	0.1	0.05	3	$1e-6$
AGNEWS	0.03	0.05	0.1	0.05	3	$1e-6$
IMDB	0.03	0.1	0.1	0.05	2	$1e-6$
QQP	0.04	0.05	0.1	0.05	3	$1e-6$
QNLI	0.04	0.05	0.1	0.05	3	$1e-6$
MNLI	0.2	0.1	0.1	0.05	2	$1e-6$

Table 7: The combinations of hyperparameters for finding Robust Tickets.

B.3 Hyperparameters for pruning model

As some adversarial training method, introduces four widely used hyperparameters: adversarial step size K, initiate adversarial margin ϵ, number of adversarial steps m. In addition, we also report two important hyperparameters in the pruning period. They are mask learning rate β and regularization penalty coefficient λ. The weight decay w in the optimizer are also changed compared with default settings to make mask sparsity rate converge better. We list the best combinations of hyperparameters for each tasks in Table 7.



(a) IMDB

(b) AGNEWS

(c) QNLI

(d) QQP

Figure 5: Model masks convergence rate on four datasets.