Adversarial Distillation: Combining Two-Stage Knowledge Distillation with Conditional Generative Adversarial Networks

Wei Wang[a], Yu Xiang[b,*], Yonghao Wu[c], Tongzhu Zhao[d], Tiancai Zhu[e]

School of Information Science and Technology, Yunnan Normal University, Kunming, China
[a]2324100048@ynnu.edu.cn, [b]xiangyu@ynnu.edu.cn, [c]2324100051@ynnu.edu.cn,
[d]2324100063@ynnu.edu.cn, [e]2324100067@ynnu.edu.cn
[*]Corresponding author

Abstract: *In traditional knowledge distillation, a significant capacity gap between the teacher and student models often leads to information loss and performance degradation. To address this issue, this study proposes a two-stage knowledge distillation framework. In the first stage, a progressive distillation strategy is employed, transferring knowledge from RoBERTa to BERT and then to BiLSTM, gradually reducing model complexity while sharing model weights to enhance knowledge transfer. In the second stage, a Conditional Generative Adversarial Network (CGAN) is introduced, utilizing the first-stage student model's output as a conditional input for adversarial training, guiding optimization between RoBERTa and BiLSTM to improve the classification performance of the student model. Additionally, zscore normalization is applied to ensure that the student model focuses on relative relationships between classes rather than absolute logits values, effectively mitigating performance bottlenecks caused by the capacity gap. Experimental results on multiple NLP datasets demonstrate that the proposed method significantly enhances the classification performance of the student model, achieving 85%-90% of the teacher model's performance, while substantially reducing model parameters, outperforming traditional knowledge distillation methods.*

Keywords: *Knowledge Distillation, Genarative Adversatial Networks, Text Classification, Artificial Intelligence*

1. Introduction

Knowledge distillation, proposed by Hinton et al.[1], is a model compression technique that trains a light weight student model to mimic a more powerful teacher model, aiming to maintain model performance while reducing computational costs. However, traditional knowledge distillation methods may suffer from information loss or performance degradation when handling complex tasks. For instance, when there is a significant capacity gap between the teacher and student models, the student model may fail to capture the intricate knowledge encoded in the teacher model's outputs, leading to suboptimal distillation results.

To address these challenges, this paper introduces a two-stage distillation framework designed to progressively reduce model complexity through phased distillation, enabling smoother knowledge transfer to the student model. Experimental results demonstrate that, compared to single-stage distillation, the two-stage approach allows the student model to acquire more comprehensive knowledge from the teacher model. Furthermore, by integrating a Conditional Generative Adversarial Network (CGAN)[2], the output of the student model from the first distillation stage is utilized to guide the optimization process in the second stage, further enhancing the student model's classification performance. Experimental results show that the proposed method outperforms traditional single-stage distillation across multiple datasets.

Additionally, to improve the student model's ability to learn inter-class relationships, this paper employs logits normalization techniques. By applying the z-score function[3] to standardize the logits outputs of both the teacher and student models, the student model focuses on mimicking the relative relationships between categories in the teacher model rather than their absolute values. This approach effectively mitigates the capacity gap between the teacher and student models, preventing performance bottlenecks caused by the student model directly imitating the logits values. The main contributions of this paper include:

1) Proposing a Two-Stage Dual Knowledge Distillation Framework: Distilling from RoBERTa to BERT in the first stage, and then from BERT to BiLSTM in the second stage, progressively reducing model complexity. During the dual distillation process, model weights are shared to ensure effective knowledge transfer. Additionally, the z-score function is used to standardize the logits output by the teacher and student models, helping the student model focus on learning the class relationships predicted by the teacher model.

2) Introducing a Conditional Generative Adversarial Network (CGAN): Leveraging the output of the first-stage student model (BiLSTM), the CGAN guides the adversarial optimization between RoBERTa and BiLSTM during the second-stage distillation, thereby enhancing the generalization ability and classification performance of the student model.

3) Multi-Task Joint Optimization: Combining classification loss, adversarial loss, and distillation loss during the distillation process to further improve the expressiveness and classification effectiveness of the student model.

2. Related Work

2.1 Knowledge Distillation

Knowledge distillation aims to transfer “dark knowledge” from a complex teacher model to a lightweight student model. By learning from the soft labels provided by the teacher model, the student can achieve better performance than training solely with hard labels. In the field of classification networks, knowledge distillation focuses on the transfer of soft label knowledge and intermediate feature layer knowledge. The solution emphasizes effective supervised learning and knowledge transfer from the teacher network to the student network.

After the introduction of the traditional Knowledge Distillation (KD) method, researchers have explored various improvements based on its core ideas. For example, compared to the soft label output of traditional KD[1], Park W et al. proposed Relational Knowledge Distillation (RKD)[4], which focuses on the structural output information of the model. By transferring distance and angular information from the model’s outputs, RKD improves the measurement of distillation loss and achieves better distillation performance than KD. Zhao B et al. proposed Decoupled Knowledge Distillation (DKD)[5], which decomposes the classical KD loss into Target Class Knowledge Distillation (TCKD) and Non-Target Class Knowledge Distillation (NCKD). This method explains and verifies that TCKD is responsible for adjusting the difficulty level of training samples during knowledge transfer, while NCKD is the main component responsible for transferring soft label knowledge.

Additionally, Seyed-Iman Mirzadehetal. Proposed Teacher-Assistant Knowledge Distillation(TAKD) [6], which introduces an assistant model between the traditional teacher model and student model to facilitate a smoother transition of ”knowledge” to the student model. Zheng Li et al. proposed Curriculum Temperature for Knowledge Distillation (CTKD)[7], which employs an adversarial learning module to predict sample temperature, adapting to varying sample difficulties. Shangquan Sun et al.[3] proposed logit distillation preprocessing to adaptively allocate temperature between the teacher and student models as well as across samples.

2.2 Generate Adversarial Networks

Generative Adversarial Networks (GANs) [8] are an innovative unsupervised learning framework inspired by the zero-sum game theory in game theory. The adversarial training concept of GANs allows the generator and discriminator to optimize their parameters through mutual competition, generating high-quality samples and demonstrating stronger feature learning and expressive capabilities compared to traditional algorithms. Through continuous iterative optimization, GANs eventually reach a Nash equilibrium, completing the modeling of the sample distribution.

Researchers have proposed various variants to address issues such as training difficulties in GANs. Mirza M et al. introduced Conditional Generative Adversarial Nets (CGAN)[2], which adds constraints to the original GAN, addressing the problem of excessive freedom in GANs and guiding the network to generate samples in a desired direction. Zhu W et al. proposed Deep Convolutional GANs (DCGAN) [9], which combines Convolutional Neural Networks (CNNs) with GANs, ensuring the quality and diversity of generated images. Zhao J B et al. proposed Energy-Based GANs (EBGAN) [10], which interprets GANs from an energy perspective, assigning low energy to real samples and high energy to generated samples, thereby improving model stability.

Berthelot D et al. introduced Boundary Equilibrium GANs (BEGAN)[11], which does not directly estimate the distance between the generated distribution $p\ g$ and the real distribution $p\ x$, but instead estimates the error distribution between the generated and real data distributions and optimizes the error distribution. Additionally, they proposed a hyperparameter that balances sample diversity and quality, as well as a method to measure model convergence. Mao X D et al. proposed Least Squares GANs (LSGAN)[12], which replaces the cross-entropy loss function of traditional GANs with a least squares loss function. This approach effectively addresses the issues of low-quality image generation and unstable training in traditional GANs.

3. Methods

3.1 Background

3.1.1 Knowledge Distillation

Knowledge distillation guides the training of the student model through the softened output labels from the teacher model and the true training labels. Specifically, knowledge distillation introduces the concepts of temperature T and softened labels. Under the condition of parallel training with true labels, it achieves the transfer of soft label knowledge from the teacher network to the student network. For a given dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, the Kullback-Leibler (KL) [13]divergence loss is used to minimize the difference between the soft output probabilities of the student model and the teacher model:

$$L_{kd}(q^t, q^s, \tau) = \sum_{i=1}^{N} \tau^2 \, KL\left(\sigma\left(\frac{q_i^t}{\tau}\right), \sigma\left(\frac{q_i^s}{\tau}\right)\right) \tag{1}$$

Here, q^t represents the soft labels generated by the teacher model, q^s represents the soft labels generated by the student model, and $\sigma(\cdot)$ denotes the Softmax function [14], which converts logits into probability distributions. T is the temperature parameter used to scale the logits output by both the teacher and student models. When T is small, the probability distribution output by the teacher model becomes sharper, approaching hard labels (where the probability of one class is close to 1, and the probabilities of other classes are close to 0). Conversely, when T is large, the probability distribution output by the teacher model becomes smoother, with the probabilities of high-probability categories decreasing and the probabilities of low-probability categories increasing, causing the overall distribution to tend toward uniformity.

3.1.2 z-score Function

In traditional knowledge distillation, the student model is typically required to directly mimic the logits values output by the teacher model. However, due to the differences in parameter scale and capability between the teacher model and the student model, direct imitation may lead to performance degradation. To address this issue, this paper introduces the z-score [3] normalization method. By standardizing the logits, the student model is enabled to focus on learning the relative relationships between different categories in the teacher model, rather than the absolute values.

$$\hat{\mathbf{v}} = \frac{\mathbf{v}-\mu(\mathbf{v})}{\sigma(\mathbf{v})}, \quad \hat{\mathbf{z}} = \frac{\mathbf{z}-\mu(\mathbf{z})}{\sigma(\mathbf{z})} \tag{2}$$

Here, given the output logits vector $\mathbf{v}$ from the teacher model and the output logits vector $\mathbf{z}$ from the student model, where $\mu(\cdot)$ represents the mean of the logits vector and $\sigma(\cdot)$ represents the standard deviation of the logits vector. Through this standardization operation, the student model can ignore the absolute differences in logits and focus solely on the relative ranking and differences between categories in the teacher model.

3.1.3 Conditional Generative Adversarial Nets

Based on the original GAN, constraints are added to address the issue of excessive freedom in GANs. In this paper, the selected constraint y is the true label. By incorporating y as an additional input layer to both the discriminator and the generator, regulation is achieved. The output of the student model from the first stage is utilized as the conditional information for the CGAN. The generator attempts to produce logits that closely resemble those of the teacher model, while the discriminator distinguishes between the real teacher logits and the generated logits.

$$\min_{G}\max_{D}V(D,G)=\mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)]+\mathbb{E}_{z\sim p_{z}(z)}[\log(1-D(G(z)))] \quad (3)$$

Here, D is the discriminator, G is the generator, z is the noise vector, and y is the constraint condition. The objective function of the Generative Adversarial Network (GAN) is optimized by minimizing the generator G and maximizing the discriminator D through the adversarial loss $V(D,G)$. Specifically, the discriminator D aims to maximize the probability $\log D(x)$ of correctly identifying real data x (sampled from the data distribution $p_{\text{data}}(x)$), while minimizing the probability $\log(1-D(G(z)))$ of incorrectly identifying generated data $G(z)$ (sampled from the noise distribution $p_z(z)$). The generator G, on the other hand, aims to minimize the discriminator's probability $\log(1-D(G(z)))$ of identifying generated data, thereby making the generated data closer to the real data distribution.

3.2 Adversarial Distillation

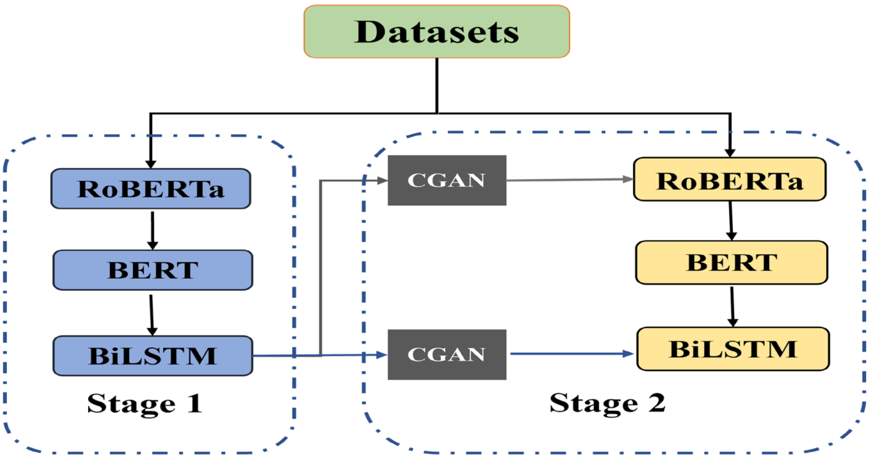


Figure 1: Distillation route

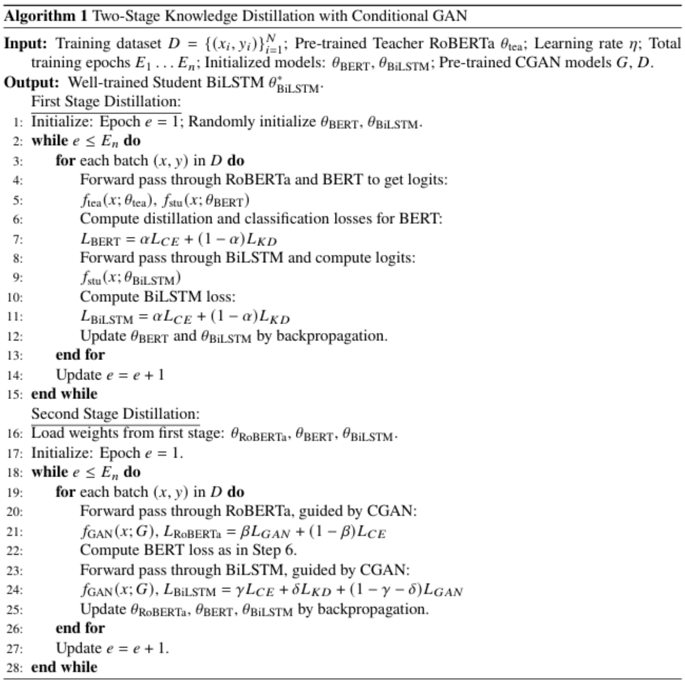
Based on the above background, to address the challenges of knowledge distillation caused by the capacity gap between the teacher model and the student model, this paper proposes a two-stage distillation framework with two distillation steps. In each distillation step, the student model transitions from complex to simple architectures, specifically from RoBERTa to BERT, and then to BiLSTM. Additionally, a Conditional Generative Adversarial Network (CGAN) is introduced to guide the student model in optimizing its output through adversarial training(as shown in figure 1).Furthermore, to enable the student model to more effectively learn the relative relationships between categories rather than absolute values from the teacher model's output, this paper incorporates z-score normalization on the outputs of each model, allowing the model to focus on the relative relationships between categories.

As shown in figure 1,in the first stage of distillation, the process goes from RoBERTa to BERT, and then from BERT to BiLSTM, allowing the student model to progressively learn the soft labels from the teacher model while maintaining model performance. The logits output by each model are standardized using the z-score function, enabling the student model to focus on learning the relative relationships between categories. Specifically:

RoBERTa uses only the classification loss L_{CE} [15].

BERT and BiLSTM use a weighted sum of the classification loss and the distillation loss (As shown in Equation (4)):

$$L_{\text{total}}=\alpha L_{CE}+(1-\alpha)L_{KD} \quad (4)$$



```
Algorithm 1 Two-Stage Knowledge Distillation with Conditional GAN
Input: Training dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$; Pre-trained Teacher RoBERTa $\theta_{\text{tea}}$; Learning rate $\eta$; Total
    training epochs $E_1 \dots E_n$; Initialized models: $\theta_{\text{BERT}}, \theta_{\text{BiLSTM}}$; Pre-trained CGAN models $G, D$.
Output: Well-trained Student BiLSTM $\theta^*_{\text{BiLSTM}}$.
    First Stage Distillation:
 1: Initialize: Epoch $e = 1$; Randomly initialize $\theta_{\text{BERT}}, \theta_{\text{BiLSTM}}$.
 2: while $e \le E_n$ do
 3:     for each batch $(x, y)$ in $D$ do
 4:         Forward pass through RoBERTa and BERT to get logits:
 5:         $f_{\text{tea}}(x; \theta_{\text{tea}}), f_{\text{stu}}(x; \theta_{\text{BERT}})$
 6:         Compute distillation and classification losses for BERT:
 7:         $L_{\text{BERT}} = \alpha L_{CE} + (1 - \alpha) L_{KD}$
 8:         Forward pass through BiLSTM and compute logits:
 9:         $f_{\text{stu}}(x; \theta_{\text{BiLSTM}})$
10:         Compute BiLSTM loss:
11:         $L_{\text{BiLSTM}} = \alpha L_{CE} + (1 - \alpha) L_{KD}$
12:         Update $\theta_{\text{BERT}}$ and $\theta_{\text{BiLSTM}}$ by backpropagation.
13:     end for
14:     Update $e = e + 1$
15: end while
    Second Stage Distillation:
16: Load weights from first stage: $\theta_{\text{RoBERTa}}, \theta_{\text{BERT}}, \theta_{\text{BiLSTM}}$.
17: Initialize: Epoch $e = 1$.
18: while $e \le E_n$ do
19:     for each batch $(x, y)$ in $D$ do
20:         Forward pass through RoBERTa, guided by CGAN:
21:         $f_{\text{GAN}}(x; G), L_{\text{RoBERTa}} = \beta L_{GAN} + (1 - \beta) L_{CE}$
22:         Compute BERT loss as in Step 6.
23:         Forward pass through BiLSTM, guided by CGAN:
24:         $f_{\text{GAN}}(x; G), L_{\text{BiLSTM}} = \gamma L_{CE} + \delta L_{KD} + (1 - \gamma - \delta) L_{GAN}$
25:         Update $\theta_{\text{RoBERTa}}, \theta_{\text{BERT}}, \theta_{\text{BiLSTM}}$ by backpropagation.
26:     end for
27:     Update $e = e + 1$.
28: end while
```

Here, L_{total} represents the total loss of the task, which consists of the classification loss L_{CE} and the distillation loss L_{KD}. The weight parameter α is used to control the balance between the two, ensuring that the total loss effectively optimizes both the classification task and the distillation task.

In the second stage of distillation, RoBERTa, BERT, and BiLSTM all load the weights of their corresponding models saved from the first distillation, achieving weight sharing between the two distillation stages. Meanwhile, a Conditional Generative Adversarial Network (CGAN) is introduced, using hard labels as conditional information to guide RoBERTa and BiLSTM in the second distillation to generate features different from the outputs of the first-stage student model BiLSTM, thereby avoiding redundant learning of ineffective information. By jointly optimizing the classification loss, distillation loss, and adversarial loss, the classification performance and generalization ability of the student model are further enhanced. Specifically:

RoBERTa: Uses a weighted sum of the classification loss and the adversarial loss(as shown in Equation(5)):

$$L_{\text{total}} = \beta L_{CE} + (1 - \beta) L_{GAN} \tag{5}$$

Here, βcontrols the weight between the classification loss and the adversarial loss.

BERT: Uses a weighted sum of the classification loss and the distillation loss, as shown in Equation(4).

BiLSTM: Combines the classification loss, distillation loss, and adversarial loss, with the sum of their weights equal to 1, ensuring a balanced loss function(as shown in Equation(6)).

$$L_{\text{total}} = \alpha L_{KD} + \beta L_{GAN} + (1 - \alpha - \beta) L_{CE} \tag{6}$$

Here, the weights of the three components are controlled by α and β.

4. Experiment

4.1 Benchmark Datesets and Models

We evaluate our proposed distillation approach on five natural language processing (NLP) benchmark datasets. These five datasets as shown in table 3 include:SST-2 [16], SST-5 [16], TREC-coarse[17], AG-news[18],SUBJ[19],all of which come from HuggingFace Datasets or the official website of the dataset. We selected RoBERTabase[20] as the teacher model, BERT-base-uncased[21]as the intermediate model, and BiLSTM[22] as the student model. For the generative adversarial network, we employed a Conditional Generative Adversarial Network (CGAN). As a knowledge distillation framework, we compared our approach with vanilla Knowledge Distillation (vanilla KD) and Teacher-Assistant Knowledge Distillation (TAKD), achieving significant performance improvements across multiple datasets.

Table1: Parameters Comparison of RoBERTa, BERT, and BiLSTM

Model	Parameters
RoBERTa	476.8M
BERT	418.2M
BiLSTM	165.59

Table 2: A Comparison of the Accuracy of two Distillation Paths on the SST-5 and SST-2 Datasets.

Datasets Route	SST-5	SST-2
Roberta--BiLSTM	36.02	50.41
Roberta—Bert--BiLSTM	37.69	52.83

4.2 Main Results

Table 1 presents a comparison of the parameter counts for the three models used in the experiments.

As shown in the table 1, RoBERTa has the largest number of parameters (476.8M), followed by BERT (418.2M), while BiLSTM has the fewest parameters (165.59M). This design, transitioning from complex to simple model architectures, helps to gradually compress the model size while maintaining performance. Table 2 compares the accuracy of two distillation paths on the SST-5 and SST-2 datasets. The results demonstrate that TAKD (Roberta–Bert–BiLSTM) achieves higher accuracy on both SST-5 and SST-2 datasets compared to the traditional Vanilla KD (Roberta–BiLSTM), indicating that the introduction of the intermediate model BERT effectively enhances the performance of the student model BiLSTM.

Table 3: Dataset information including sample size and number of classes

Dataset	**Sample Size**	**Number of Classes**
SST-2	9613	2
SST-5	11855	5
TREC-coarse	5952	6
SUBJ	10000	2
AG-news	120000	4

Table 4: Accuracy of RoBERTa on SST-5 and SST-2 datasets

	β	**SST-5**	**SST-2**
Stage 1	-	55.34	94.23
Stage 2	0.9	**56.56**	**94.78**
	0.8	55.7	94.67
	0.7	55.93	94.34
	0.6	56.52	92.26
	0.5	55.48	94.67
	0.4	56.11	94.34
	0.3	53.12	94.4
	0.2	51.72	94.56
	0.1	51.52	94.07

After the completion of the first stage of distillation, we utilize the output of the student model to introduce a Conditional Generative Adversarial Network (CGAN) to guide the output of the teacher model in the second stage, thereby completing the training of the second stage. As shown in Table 4, after introducing the Conditional Generative Adversarial Network, the teacher model (RoBERTa) in the second stage achieves better classification performance under certain β values compared to the teacher model (RoBERTa) in the first stage. This result demonstrates that the introduction of the Conditional Generative Adversarial Network effectively helps the model learn more useful features.

Table 5: Accuracy comparison of different weight-sharing strategies on SST-5 and SST-2 datasets

	SST-5			SST-2		
	No Sharing	**Partial**	**Full**	**No Sharing**	**Partial**	**Full**
Stage 1	36.61	36.61	36.61	52.83	52.83	52.83
Stage 2 α						
0.9	37.74	37.78	39.14	75.67	74.3	**76.39**
0.8	38.69	38.19	**39.41**	74.46	74.85	76.17
0.7	38.82	37.6	39.32	74.46	74.63	76.22
0.6	37.65	38.14	39.37	75.01	74.24	76.33
0.5	37.87	37.96	38.91	75.29	74.24	76.17
0.4	36.29	38.24	39.14	74.52	73.59	76.28
0.3	37.74	37.87	39.32	74.9	74.52	76
0.2	37.15	37.56	39	74.52	56.34	76.11
0.1	37.16	38.01	**39.41**	74.96	73.86	75.78
Ave Improvement	+1.07	+1.32	**+2.35**	+22.03	+19.45	**+23.33**

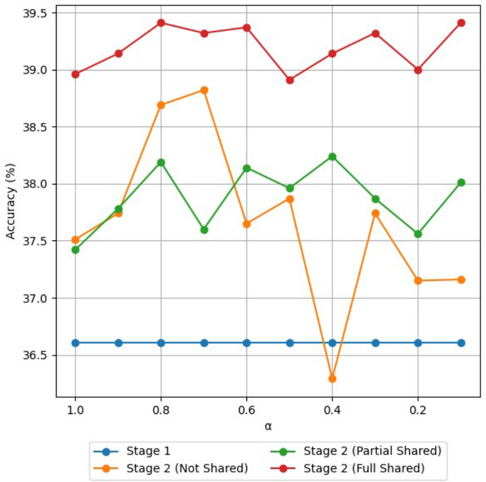


Figure 2: Accuracy Trends of the Second-Stage Student Model Under Different Weight-Sharing Strategies on SST-5

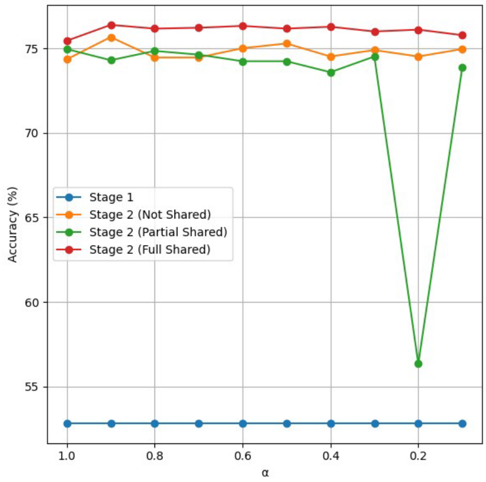


Figure 3: Accuracy Trends of the Second-Stage Student Model Under Different Weight-Sharing Strategies on SST-2

During the training process of the second stage, we discovered that weight sharing between corresponding models in the two stages could enhance performance. By combining adversarial loss with three different weight-sharing strategies (no sharing, partial weight sharing, and full weight sharing), we found that full weight sharing significantly improved the accuracy of the BiLSTM model in the second stage compared to no weight sharing. As shown in Table 5 and Figures 2 and 3, the full weight-sharing strategy achieved greater accuracy improvements on both the SST-2 and SST-5 datasets compared to the other two strategies, and it demonstrated higher robustness to changes in the α values. Specifically, the highest accuracy on the SST-2 dataset reached 76.39%, while the highest accuracy on the SST-5 dataset was 39.41%.

As shown in Figure 4, while adopting a full weight-sharing strategy, we introduce a CGAN to guide the second-stage teacher model using the output of the first-stage student model. Additionally, we further incorporate another CGAN to enable the first-stage student model to guide the learning process of the second-stage student model. Experimental results demonstrate that this improved strategy significantly enhances classification performance.

Table6: BiLSTM classification performance on SST-5 with different hyperparameter values

Stage 1		36.61		
	β α	0.1	0.2	0.3
Stage 2	0.9	38.19	-	-
	0.8	36.83	36.65	-
	0.7	39.95	38.37	36.74
	0.6	39.19	39.28	39
	0.5	40.63	**40.72**	40.68
	0.4	37.69	40.41	40.05
	0.3	37.83	37.65	40.09
	0.2	37.96	37.51	38.55
	0.1	38.60	33.76	37.47
	0	38.42	39.00	34.39
Ave Improvement		**+2.53**	+1.54	+1.76

Table7: BiLSTM classification performance on SST-2 with different hyperparameter values

Stage 1		52.83		
	β α	0.1	0.2	0.3
Stage 2	0.9	76.77	-	-
	0.8	75.51	77.21	-
	0.7	77.21	76.22	77.16
	0.6	79.21	77.59	74.3
	0.5	78.75	79.35	76.39
	0.4	76.61	78.75	77.05
	0.3	78.69	79.24	79.13
	0.2	79.46	79.02	78.379
	0.1	79.02	79.13	79.13
	0	**79.74**	79.19	79.13
Ave Improvement		+25.27	**+25.5**8	+24.755

Table8: BiLSTM classification performance on TREC-coarse with different hyperparameter values

Stage 1		62.6		
	β α	0.1	0.2	0.3
Stage 2	0.9	81.8	-	-
	0.8	84	72.6	-
	0.7	85	82.6	72.4
	0.6	85.4	84	81.4
	0.5	87.2	83.4	82.6
	0.4	87.6	85	82.6
	0.3	**88.2**	87	85.6
	0.2	86.4	85.2	85.6
	0.1	86.4	86.2	84.8
	0	77.8	85.4	85.4
Ave Improvement		**+22.3**	+20.8	+19.9

Table9: BiLSTM classification performance on AG-news with different hyperparameter values

Stage 1		90.36		
Stage 2	β / α	0.1	0.2	0.3
	0.9	89.43	-	-
	0.8	90.24	90.49	-
	0.7	90.80	90.95	89.47
	0.6	**91.57**	91.21	90.33
	0.5	91.53	91.05	90.92
	0.4	**91.57**	90.96	89.67
	0.3	91.26	91.12	90.82
	0.2	89.64	91.11	90.28
	0.1	89.79	91.08	90.88
	0	89.17	90.78	90.14
Ave Improvement		+0.57	**+0.61**	-0.05

We applied the final proposed distillation approach to five datasets and conducted experiments under different hyperparameter settings. The results indicate that in the total loss of the second-stage student model, the best classification performance is consistently achieved when the adversarial loss weight is set to 0.1 or 0.2, suggesting that an excessively large adversarial loss weight may be detrimental.

As shown in Tables 6 -10, on the SST-5 dataset, the highest accuracy increased from 39.41% to 40.72%, with the average improvement rising from 2.35 to 2.53.On the SST-2 dataset, the highest accuracy improved from 76.39% to 79.74%, with the average improvement increasing from 23.33% to 25.27%. In the remaining three datasets, the second-stage student model consistently outperformed the first-stage student model across different hyperparameter settings. Specifically, the maximum average improvement reached 22.38% on the TREC-coarse dataset, 0.57% on the AG-news dataset, and 40.24% on the SUBJ dataset.

Table10: BiLSTM classification performance on SUBJ with different hyperparameter values

Stage 1		62.6		
Stage 2	β / α	0.1	0.2	0.3
	0.9	89.55	-	-
	0.8	90.45	89.45	-
	0.7	90.8	90.05	85.6
	0.6	91.3	90.55	86.2
	0.5	91.35	91.2	86
	0.4	90.95	91.25	85.8
	0.3	91.35	90.75	85.4
	0.2	91.4	91.45	86.4
	0.1	91.2	**91.9**	86.2
	0	91	91.25	87.2
Ave Improvement		**+40.24**	+40.17	+35.4

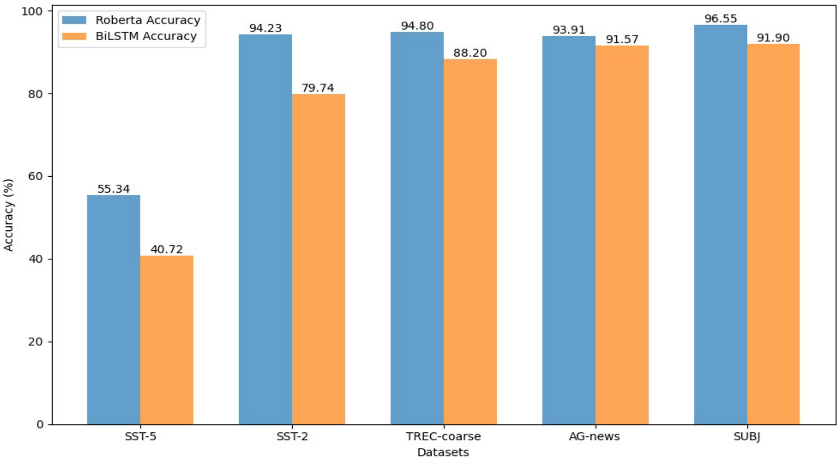


Figure 4: Accuracy Comparison Between the First-Stage Teacher Model(Roberta) and the Second-Stage Student Model(BiLSTM) Across Different Datasets

As shown in Table 1 and Figure 4, with the proposed method in this paper, the student model requires only 35% of the teacher model's parameters, while the second-stage student model achieves 85% to 90% of the teacher model's performance in terms of accuracy on most datasets.

Table11:Accuracy of BiLSTM atβ=0.1 on SST-5 Dataset with Adversarial Training Between Two Student Models Only

α	β	BiLSTM
1	0	39.05
0.9	0.1	38.19
0.8	0.1	36.68
0.7	0.1	39.95
0.6	0.1	39.19
0.5	0.1	**40.63**
0.4	0.1	37.69
0.3	0.1	37.83
0.2	0.1	37.96
0.1	0.1	38.6
0	0.1	38.42
Ave Improvement		**-0.521**

Table12:Accuracy of BiLSTM at β=0.2 on SST-5 Dataset with Adversarial Training Between Two Student Models Only

α	β	BiLSTM
1	0	39.05
0.8	0.2	38.28
0.7	0.2	36.7
0.6	0.2	**40.05**
0.5	0.2	38.51
0.4	0.2	39.86
0.3	0.2	36.11
0.2	0.2	36.74
0.1	0.2	37.19
0	0.2	38.33
Ave Improvement		**-1.0756**

4.3 Ablation Study

Furthermore, we conducted an in-depth investigation into the classification performance of different distillation routes on the SST-5 dataset. Building upon our previous experimental setup, we retained only the adversarial training module for the student model during both stages of distillation. As shown in Table 11 - 12, introducing the adversarial module led to a decline in classification performance for the student model in the second distillation stage across various hyperparameter settings compared to the scenario without adversarial training.

Additionally, we explored the effectiveness of directly incorporating a Conditional Generative Adversarial Network (CGAN) after a single distillation to guide the student model for secondary classification.

Table 13: Comparison of Secondary Classification Performance with CGAN after Single Distillation

Before		36.61	
	Adversarial	Classification	
After	0.9	0.1	37.42
	0.8	0.2	32.65
	0.7	0.3	38.05
	0.6	0.4	38.05
	0.5	0.5	37.01
	0.4	0.6	38.82
	0.3	0.7	38.14
	0.2	0.8	39.46
	0.1	0.9	38.55
Ave Improvement			+0.96

As shown in Table 13, after adversarial training, the student model achieved an average classification accuracy improvement of 0.96% across different hyperparameter settings compared to when adversarial training was not applied.

Finally, we further introduced three adversarial modules, allowing the same models from both distillation stages to participate in adversarial training. However, experimental results showed that the classification performance of the student model after the second distillation did not surpass that of the student model after the first distillation across various hyperparameter settings. This suggests that while adversarial training can enhance model performance in a single distillation, incorporating excessive adversarial modules in multiple distillation stages may lead to performance degradation. A possible explanation is that the overly complex adversarial training interfered with the effective transfer of knowledge.

5. Discussion

In this paper, we propose an adversarial distillation framework that integrates two-stage knowledge distillation with Conditional Generative Adversarial Networks (CGAN) to address the performance degradation caused by the capacity gap between the teacher and student models. Through a two-stage distillation process (from RoBERTa to BERT, and then to BiLSTM) and z-score normalization, the student model can more effectively learn the inter-class relationships from the teacher model. The introduction of CGAN significantly enhances the generalization ability and classification performance of the student model. On the SST-2 and SST-5 datasets, the student model achieves 85%-90% of the teacher model's accuracy while using only 35% of the teacher model's parameters. Experimental results demonstrate that the full weight-sharing strategy performs the best, and moderate adversarial training facilitates knowledge transfer, while overly complex adversarial mechanisms may hinder effective knowledge transfer. This framework has made significant progress in model compression and knowledge transfer, and future research can further explore how to optimize the adversarial distillation framework for different tasks and model architectures to achieve broader applicability.

References

[1] Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
[2] Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784

(2014)

[3] Sun, S., Ren, W., Li, J., Wang, R., Cao, X.: Logit standardization in knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15731–15740 (2024)

[4] Park, W., Kim, D., Lu, Y., Cho, M.: Relational knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3967–3976 (2019)

[5] Zhao, B., Cui, Q., Song, R., Qiu, Y., Liang, J.: Decoupled knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11953–11962 (2022)

[6] Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 5191–5198 (2020)

[7] Li, Z., Li, X., Yang, L., Zhao, B., Song, R., Luo, L., Li, J., Yang, J.: Curriculum temperature for knowledge distillation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, pp. 1504–1512 (2023)

[8] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems 27 (2014)

[9] Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)

[10] Zhao, J., Mathieu, M., LeCun, Y.: Energy-based generative adversarial network. arXiv preprint arXiv:1609.03126 (2016)

[11] Berthelot, D., Schumm, T., Metz, L.: Began: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717 (2017)

[12] Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2794–2802 (2017)

[13] Kullback, S., Leibler, R.A.: On information and sufficiency. The annals of mathematical statistics 22(1), 79–86 (1951)

[14] Bridle, J.S.: Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In: Neurocomputing: Algorithms, Architectures and Applications, pp. 227–236. Springer, Berlin, Heidelberg (1990)

[15] Shannon, C.E.: A mathematical theory of communication. The Bell system technical journal 27(3), 379–423 (1948)

[16] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A.Y., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1631–1642 (2013)

[17] Li, X., Roth, D.: Learningquestionclassifiers. In: COLING2002: The 19th Internationa lConference on Computational Linguistics (2002)

[18] Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. Advances in neural information processing systems 28 (2015)

[19] Conneau, A., Kiela, D.: Senteval: An evaluation toolkit for universal sentence representations. arXiv preprint arXiv:1803.05449 (2018)

[20] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)

[21] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: 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), pp. 4171–4186 (2019)

[22] Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional lstm networks. In: Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005., vol. 4, pp. 2047–2052 (2005). IEEE