

Periodic Segment-Wise Knowledge Distillation with Adaptive Temperature

Guiyu Li

School of Computer Science, Sichuan University Jinjiang College, Meishan, China

Email: lguiyu37@163.com

How to cite this paper: Li, G.Y. (2026) Periodic Segment-Wise Knowledge Distillation with Adaptive Temperature. *Journal of Computer and Communications*, 14, 98-122.

<https://doi.org/10.4236/jcc.2026.1410005>

Received: September 14, 2026

Accepted: October 24, 2026

Published: October 27, 2026

Copyright © 2026 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Pre-trained language models (PLMs) achieve strong performance but are difficult to deploy because of their large parameter sizes and high computational costs. Existing knowledge distillation methods generally use fixed temperatures and uniform training strategies, limiting their ability to handle differences in sample difficulty and teacher reliability. To address this issue, this study proposes Periodic Segment-wise Knowledge Distillation with Adaptive Temperature (SAT). SAT introduces a Segment-wise Focal Loss (SFL) to periodically adjust the learning focus across training stages and an Adaptive Temperature Distillation Loss (ATL) to assign sample-specific temperatures according to teacher uncertainty. By jointly modeling sample difficulty and prediction reliability, SAT improves the efficiency and robustness of knowledge transfer. Experiments on language modeling and natural language understanding tasks show that SAT improves the performance and generalization ability of student models while preserving their inference efficiency.

Keywords

Pre-Trained Language Models, Knowledge Distillation, Adaptive Temperature, Sample Difficulty, Teacher Uncertainty

1. Introduction

Recent years have witnessed remarkable progress by pre-trained language models (PLMs) in natural language understanding, text generation, and language modeling by leveraging large-scale pre-training [1]-[3]. However, their increasing parameter size, computational complexity, and storage requirements hinder deployment on mobile devices, edge platforms, and low-latency services. Knowledge distillation (KD) addresses this issue by transferring the output distributions, class relationships, and intermediate representations of high-capacity teacher models

to lightweight student models [4]. Compared with one-hot labels, the soft labels generated by teacher models provide richer inter-class similarity information and have therefore been widely used in model compression and PLM optimization [5]-[7].

Conventional KD methods generally minimize the discrepancy between teacher and student outputs using temperature-scaled cross-entropy or Kullback-Leibler divergence. Increasing the distillation temperature smooths the teacher distribution and exposes inter-class relationships [4]. Representative methods such as DistilBERT and TinyBERT demonstrate that distillation can substantially reduce model size and inference cost while preserving language understanding ability [8] [9]. However, existing methods commonly rely on fixed temperatures, uniform loss weights, and static data-processing strategies, which are insufficient for handling variations in sample difficulty and class distribution.

In natural language data, minority classes may be dominated by majority-class gradients, while samples can differ considerably in semantic complexity, contextual dependency, and classification ambiguity. Easy samples are generally suitable for stabilizing early training, whereas difficult samples provide valuable information for learning complex decision boundaries and improving generalization. In addition, teacher predictions may be unreliable because of noise, distribution shifts, ambiguity, and model bias. Directly using uncertain or incorrect predictions as soft supervision may therefore propagate erroneous knowledge to the student model. Although existing studies have explored teacher ensembles, label smoothing, confidence calibration, and uncertainty modeling [10]-[13], many still use globally fixed temperatures or uniform distillation weights.

To address these limitations, this paper proposes periodic segment-wise knowledge distillation with adaptive temperature (SAT). The method incorporates a Segment-wise Focal Loss (SFL) and an Adaptive Temperature Distillation Loss (ATL). SFL periodically adjusts the optimization focus according to sample difficulty. Easy samples are emphasized during the early stage to establish stable representations, difficult samples receive greater attention during the middle stage to facilitate the learning of complex patterns and decision boundaries, and easy samples are revisited during the later stage for fine-grained refinement. This periodic strategy balances stable optimization, difficult-sample learning, and knowledge refinement.

ATL assigns a sample-specific temperature according to the predictive uncertainty of the teacher model. Relatively low temperatures are used for highly uncertain samples to strengthen class distinctions and reduce the influence of unreliable predictions, whereas relatively high temperatures are used for confident samples to preserve informative inter-class relationships. By jointly modeling sample difficulty and teacher reliability, SAT dynamically determines both which samples should be prioritized and how strongly they should be supervised.

The main contributions of this study are as follows:

- 1) This paper proposes SAT, an adaptive-temperature, periodic segment-wise

knowledge distillation framework that jointly considers sample difficulty and teacher-prediction uncertainty.

2) This paper develops SFL to dynamically adjust the learning emphasis between easy and difficult samples across different training stages.

3) This paper develops ATL to determine the distillation temperature according to sample-level predictive uncertainty, thereby reducing the negative effects of unreliable teacher predictions.

4) This paper conducts systematic experiments on natural language understanding and language modeling tasks to evaluate the proposed method in terms of predictive performance and inference efficiency, as well as to verify the contribution of each component.

2. Related Work

This section reviews related research on knowledge distillation, focal loss, periodic training, and dynamic-temperature distillation.

2.1. Knowledge Distillation

Knowledge distillation (KD), introduced by Hinton *et al.* [4], transfers the output distribution of a teacher model to a lightweight student model through temperature-scaled soft labels. In this way, the student learns not only from ground-truth labels but also from the “dark knowledge” encoded in the relationships among classes. Beyond output-level distillation, researchers have extended knowledge transfer to intermediate features and structural relationships. FitNets introduced intermediate representations from the teacher model as hints, enabling the student to learn deeper feature representations [6]. Attention Transfer further transfers knowledge through the attention maps of the teacher and student models, allowing the student to learn the teacher’s patterns of focus on input information [14]. Relational Knowledge Distillation goes beyond matching the features of individual samples by constraining the distances and angular relationships between samples, thereby transferring the structured knowledge learned by the teacher model [15]. These studies demonstrate that the knowledge of a teacher model is not limited to its final predictions but is also encoded in intermediate representations, attention distributions, and inter-sample relationships.

Driven by progress in pre-trained language models, knowledge distillation is now commonly employed to compress and accelerate natural language processing models. During pre-training, DistilBERT combines the language modeling loss, distillation loss, and representation-level constraints to compress BERT into a smaller student model, reducing the number of parameters while preserving strong language understanding capabilities [8]. TinyBERT further performs multi-level distillation on the embedding representations, hidden states, attention matrices, and output layer of Transformer models. It also improves the performance of student models on natural language understanding tasks through the combination of general distillation and task-specific distillation [9]. MobileBERT designs a network

architecture suitable for resource-constrained devices and incorporates knowledge distillation to balance model performance and deployment efficiency [16]. In addition, MiniLM distills the self-attention relations and self-attention values of the teacher model, thereby preserving important linguistic knowledge of Transformer models at a relatively low computational cost [17].

Existing knowledge distillation methods primarily focus on maximizing the capacity and information coverage of knowledge transfer, but relatively little attention has been given to two critical factors that limit distillation performance. First, student models may have limited ability to learn from difficult samples or challenging semantic segments. Second, teacher predictions inevitably contain uncertainty, which may cause erroneous or low-reliability knowledge to be propagated to the student model. To overcome these challenges, we propose a knowledge distillation framework that makes joint use of the Segment-wise Focal Loss (SFL) and the Adaptive Temperature Distillation Loss (ATL). The proposed framework jointly optimizes the distillation process from two perspectives—imbalanced sample difficulty and uncertainty in teacher predictions—thereby improving the robustness and effectiveness of knowledge transfer.

2.2. Focal Loss

Focal loss was originally proposed by Lin *et al.* [18] to address the severe imbalance between positive and negative samples in object detection. In most cases, network training is guided by the cross-entropy loss. However, focal loss generally achieves superior performance to cross-entropy loss in scenarios involving imbalanced class distributions, foreground-background imbalance in the image domain [19], or imbalanced positive and negative samples in multi-label classification [20]. By modifying the cross-entropy loss, focal loss encourages neural networks to focus more on difficult-to-classify samples and assign greater attention to samples that are prone to misclassification. The conventional cross-entropy loss treats all samples equally, allowing the large number of easy negative samples to dominate the training process and consequently limiting the model's ability to learn from difficult samples. Focal loss introduces an adjustable modulating factor that dynamically reduces the weights of easy samples while increasing the relative emphasis placed on difficult ones. Its mathematical formulation is given as follows:

$$L_{lc} = -(1 - p_i)^{y_{lc}} y_i \log(p_i) \quad (1)$$

here, L_{lc} (Low-Confidence Loss) primarily focuses on the loss values of low-confidence samples. p_i denotes the predicted probability of a sample i , while y_i denotes its correct label. Focal loss modifies the cross-entropy loss by introducing a weighting term $(1 - p_i)^{y_{lc}}$. When the predicted probability p_i of a training sample approaches 1, indicating that the model has high confidence in its prediction, the loss weight assigned to that sample decreases rapidly. Consequently, compared with conventional cross-entropy loss, its loss term declines more quickly. This weighting mechanism directs the training process toward the relatively small

number of low-confidence samples, thereby improving the model's ability to learn difficult-to-classify examples.

Ridnik *et al.* [19] applied focal loss to multi-label classification tasks to address the imbalance between positive and negative labels. In such tasks, each image typically contains only a small number of positive labels, while most labels are negative, resulting in a much larger number of negative samples than positive samples. To address this issue, the authors proposed a variant of Focal Loss called Asymmetric Loss (ASL), which achieved better performance than the original Focal Loss in multi-label classification tasks. ASL separates the loss terms for positive and negative samples in Focal Loss and assigns an independent focusing parameter to each, thereby improving the modeling of positive-negative sample imbalance and enhancing classification performance.

$$L_{ASL} = -\sum_i \left[y_i (1 - p_i)^{\gamma^+} \log p_i + (1 - y_i) p_i^{\gamma^-} \log (1 - p_i) \right] \quad (2)$$

here, y_i denotes the ground-truth label of the sample, p_i denotes the predicted probability obtained after applying the sigmoid function to the sample, γ^+ is the focusing parameter for positive samples, which enhances the learning of difficult positive samples, and γ^- is the focusing parameter for negative samples, which suppresses the loss contribution of easily classified negative samples. Nevertheless, existing methods still cannot fully account for the characteristics of different training stages or flexibly adjust the degree of attention assigned to samples throughout the training process. Inspired by curriculum learning, this paper proposes a periodic segment-wise focal loss that divides the training process into multiple stages and dynamically adjusts the weighting assigned to the focal loss at each stage.

2.3. General Cyclical Training

General Cyclical Training (GCT) [20] refers to a general machine learning training paradigm that partitions the training procedure into a series of suitably ordered stages. In the initial phase, the model prioritizes samples that are easy to classify.; in the middle stage, its attention expands to more challenging samples; and in the final stage, it returns to high-confidence samples for fine-tuning, thereby improving generalization and convergence efficiency. This strategy can be regarded as a combination of Curriculum Learning [21] and fine-tuning, while also leveraging the entire dataset for comprehensive optimization during the middle stage of training [22]. Studies have shown that the key learning processes of neural networks typically occur during the early stages of training. Therefore, properly scheduling the difficulty of samples can help improve the model's optimization trajectory and final performance.

Within the General Cyclical Training framework, the Cyclical Learning Rate (CLR) [23] has become one of the mainstream approaches. By dynamically adjusting the learning rates for different samples, ASL can balance the importance of different samples. In addition, Learning Rate Warmup [24] and Stochastic Gra-

dient Descent with Warm Restarts (SGDR) [25] are also regarded as variants of CLR. These methods adjust the learning rate during optimization to improve convergence speed and generalization ability. In recent years, the concept of adaptive hyperparameters has gradually been extended to weight decay [26] and dynamic batch-size adjustment strategies, further enhancing the convergence stability of models. Based on these developments, this paper further extends the principles of General Cyclical Training and explores their application to loss-function optimization. Specifically, cyclical focal loss is employed to adaptively adjust loss weights at different training stages, thereby improving the model's ability to learn from imbalanced datasets.

2.4. Dynamic Temperature Distillation

The temperature parameter plays an important role in knowledge distillation. By controlling the smoothness of the soft labels output by the teacher model, it affects the learning process of the student model. Traditional knowledge distillation methods typically use a fixed temperature parameter τ , and the soft-label generation formula is given as follows:

$$q_i = \frac{\exp(z_i/\tau)}{\sum_j \exp(z_j/\tau)} \quad (3)$$

here, z_i denotes the predicted probability output for class i by the teacher model, and q_i denotes the generated soft label. Although a fixed temperature parameter is simple to use, it has clear limitations in practical applications. For samples with high predictive uncertainty, an excessively high temperature value may amplify the uncertainty of the soft labels, causing the student model to inherit incorrect predictions from the teacher model. For easily classified samples, a fixed temperature may fail to fully exploit the relational information among different classes.

To address this issue, researchers have proposed various dynamic temperature scaling methods. For example, Zhang *et al.* [27] proposed a temperature adjustment method based on sample uncertainty. This method dynamically adjusts the temperature parameter according to the prediction entropy of each sample, as shown in the following equation:

$$\tau_{\text{dynamic}} = \tau_{\text{base}} \cdot (1 + \alpha \cdot H(q)) \quad (4)$$

here, $H(q)$ denotes the entropy of the soft labels and is used to measure the predictive uncertainty of each sample; τ_{base} denotes the base temperature, and α is a scaling coefficient. This method can dynamically adjust the temperature according to the uncertainty of each sample, thereby reducing the impact of incorrect predictions on the student model. However, existing methods typically focus only on predictive uncertainty and fail to fully consider the difficulty of each sample and its influence on temperature adjustment. Therefore, this paper proposes an adaptive temperature distillation loss that further extends the concept of

dynamic temperature adjustment. Specifically, it assigns a customized temperature parameter to each sample and dynamically adjusts it according to both the predictive uncertainty and difficulty of the sample.

3. Methodology

As shown in **Figure 1**, the proposed periodic segment-wise with adaptive temperature knowledge distillation method mainly consists of a segment-wise focal loss and an adaptive temperature distillation loss. The segment-wise focal loss module dynamically adjusts the loss weights of samples with different difficulty levels according to the training stage. Meanwhile, the adaptive temperature distillation loss module constructs a confusion measure based on the soft labels generated by the teacher model and adaptively selects the distillation temperature according to the teacher model's predictive uncertainty for the current sample.

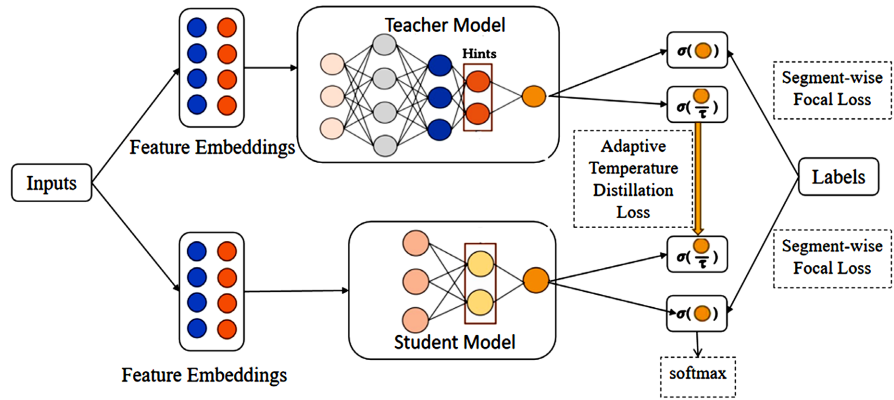


Figure 1. Framework diagram of periodic segment-wise knowledge distillation with adaptive temperature.

3.1. Segment-Wise Focal Loss

The imbalanced distribution of easy and difficult samples is prevalent in most natural language processing tasks. Conventional focal loss prioritizes difficult samples; if this bias is maintained throughout the entire training process, the model may easily overfit noisy samples, overlook the fundamental semantic features contained in easy samples, and consequently suffer a decline in generalization performance. Following the cyclical training strategy, the model should dynamically adjust the weights of easy and difficult samples at different training stages. Accordingly, this paper designs a novel loss-weighting term aimed at guiding the teacher model during fine-tuning and the student model during knowledge transfer to prioritize learning from easy samples in the early stages of training. Its specific form is given as follows:

$$L_{hc} = -(1 + p_i)^{\gamma_{hc}} y_i \log(p_i) \quad (5)$$

Equation (1) represents the focal loss, which encourages the model to focus more on low-confidence samples during training. In contrast, the loss term L_{hc} (High-Confidence Loss) places greater emphasis on high-confidence samples. **Figure 2**

compares the standard cross-entropy loss (CE) with the low-confidence loss L_{lc} and high-confidence loss L_{hc} under different parameter settings ($\gamma = 2$ and $\gamma = 4$).

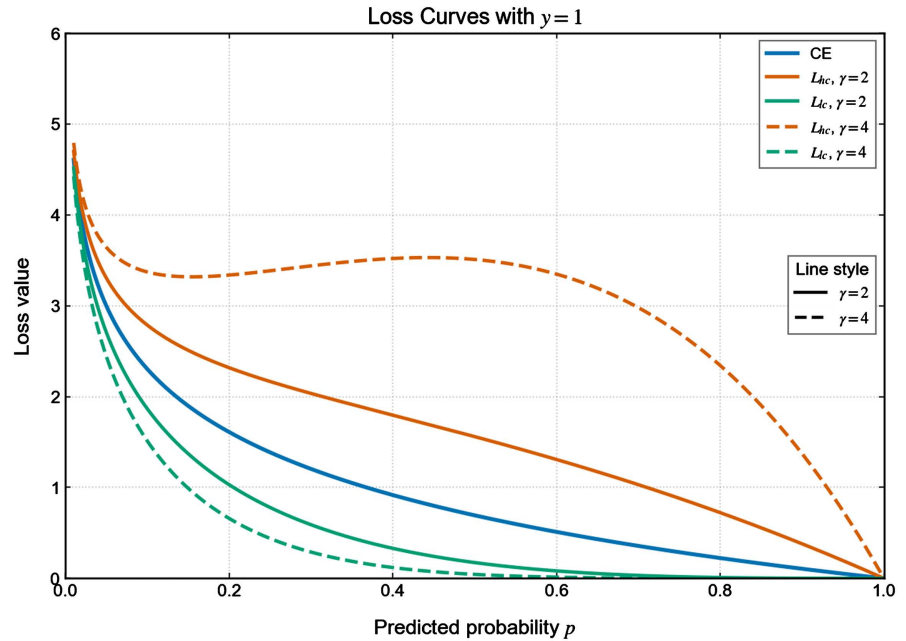


Figure 2. This figure compares the variations of cross-entropy loss, focal loss, and the proposed high-confidence weighting term across different ground truth probabilities.

As shown in **Figure 2**, different loss-weighting terms have varying effects on the loss function as the true class probability p_i changes. The blue solid line represents the cross-entropy loss (CE), whose value decreases monotonically as p_i increases. L_{lc} primarily reduces the loss weight assigned to high-confidence samples ($p_i \rightarrow 1$) while increasing the loss for low-confidence samples ($p_i \rightarrow 0$), thereby enabling the model to focus more on difficult-to-classify samples. In contrast, L_{hc} increases the loss for high-confidence samples while suppressing the influence of low-confidence samples. In particular, when γ_{hc} takes a relatively large value (e.g., $\gamma_{hc} = 4$), the loss weight of high-confidence samples increases significantly.

The objective of segment-wise focal loss is to dynamically adjust the model's attention to samples at different stages of training, thereby improving learning performance. During the early training stage, SFL primarily focuses on high-confidence samples to ensure that the model's gradient updates are more stable and that the model converges rapidly. During the middle stage, SFL gradually shifts its focus toward difficult-to-classify samples, namely those that are more likely to be misclassified. This enhances the model's ability to learn complex patterns and improves its generalization performance.

To achieve this objective, SFL combines the two loss terms in Equation (1) and Equation (5), dynamically balancing them through a reasonable scheduling strategy. In this study, a linear scheduling strategy is adopted, allowing the loss-func-

tion weights to smoothly shift from easy-to-classify samples to difficult-to-classify samples throughout the training process. This approach not only ensures training stability but also prevents the model from becoming trapped in a local optimum, thereby further improving classification performance. Accordingly, this paper defines a parameter ξ , which varies with the training epoch as follows:

$$\xi = \begin{cases} 1 - f_c \frac{e_i}{e_n} & \text{if } f_c \times e_i \leq e_n \\ \left(f_c \frac{e_i}{e_n} - 1 \right) / (f_c - 1) & \text{otherwise} \end{cases} \quad (6)$$

here, e_i denotes the current training epoch, while e_n denotes the total number of training epochs. We introduce a cycle factor $f_c \geq 1$ to provide flexibility for cyclic scheduling. When $f_c = 1$, ξ decreases linearly from 1 at the beginning of training to 0 at the end. When $f_c = 2$, the cyclic curve resembles an inverted equilateral triangle: ξ decreases from 1 to 0 during the first half of training, and then increases linearly from 0 to 1 during the second half. When $f_c = 4$, ξ decreases from 1 to 0 during the first quarter of the training epochs, and then increases linearly to 1 over the remaining three quarters.

Figure 3 is plotted according to Equation (6), illustrating the effects of different cycle factors, with the total number of training epochs set to $e_n = 20$. The curves correspond to $f_c = 1$ (blue), $f_c = 2$ (red), and $f_c = 4$ (green), reflecting the cyclical scheduling patterns of ξ as the training epochs change. Based on Equations (1) (5), and (6), this paper defines the segment-wise focal loss as follows:

$$L_{\text{SFL}}(p, y) = \xi L_{\text{hc}} + (1 - \xi) L_{\text{lc}} \quad (7)$$

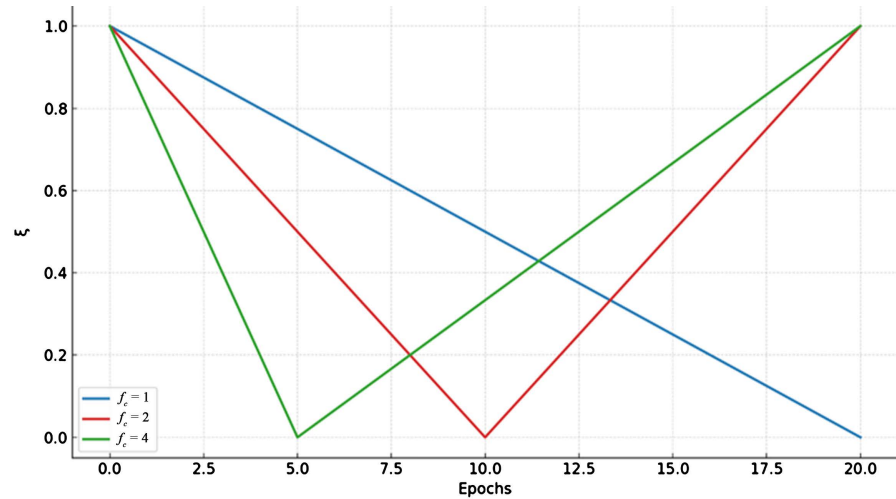


Figure 3. Hyperparameter change curves under different settings.

here, p denotes the probability predicted by the model, and y denotes the true label of the sample. By introducing the linear scheduling parameter ξ , the weighting ratio of the two focal losses is dynamically adjusted at different training stages, guiding the model to focus on samples with varying confidence levels. By

introducing the scheduling parameter ξ , the model dynamically adjusts its attention to samples of varying difficulty, as well as their corresponding loss weights, at different stages of training. During the early stage, the weights assigned to easy samples are appropriately increased to facilitate the rapid establishment of stable classification representations. During the middle stage, the learning focus gradually shifts toward hard-to-classify samples, thereby enhancing the model's ability to identify complex patterns. During the late stage, attention is again placed on easy samples to consolidate the learned feature representations and improve training stability and generalization performance.

3.2. Adaptive Temperature Distillation Loss

Before transferring knowledge to the student model, the teacher model may produce uncertain or erroneous predictions, leading to inherited errors during the knowledge distillation process. Consequently, the student model may also face a high probability of uncertainty in its predictions. Some of the uncertainty in the soft labels arises from an excessively high temperature setting during distillation. Therefore, this paper introduces an adaptive-temperature distillation loss, which adaptively adjusts the temperature according to the prediction confidence of each sample, thereby mitigating the negative impact of uncertain predictions.

1) Assume that the teacher model outputs the class probabilities as $p = [p_1, p_2, \dots, p_C]$, where C denotes the number of classes. To quantify the degree of uncertainty in the teacher model's predictions for a sample, this paper defines the confusion degree w_x , which is calculated as follows:

$$w_i = 1 - \frac{1}{C-1} \sum_{i \neq y} |p_y - p_i| \quad (8)$$

here, p_y denotes the predicted probability corresponding to the true label y of the sample, and p_i denotes the predicted probability for another class i ($i \neq y$). This formula calculates the average of the absolute differences between the predicted probability of the true label and those of the other classes, thereby reflecting the degree of confusion in the model's predictions. When w_i is close to 0, the model's prediction for the sample is relatively clear and confident; when w_i is close to 1, the model's prediction for the sample is relatively confused.

2) To further optimize the temperature adjustment mechanism in the knowledge distillation process, this paper introduces an adaptive temperature parameter based on the confusion degree w_i , which is calculated as follows:

$$\tau_i = \tau_{\max} - \left(\frac{w_i + 1}{2} \cdot (\tau_{\max} - \tau_{\min}) \right) \quad (9)$$

here, $\tau_{\max}$ and $\tau_{\min}$ represent the minimum and maximum temperature values, respectively. When the degree of confusion is high, the temperature approaches the lower value, thereby widening the probability gap between classes and helping the student model focus on the classes in which the teacher model has higher confidence. When the degree of confusion is low, the temperature approaches the higher value, enhancing the smoothness of the class probabilities and

enabling the student model to learn more inter-class relational information.

3) The final adaptive temperature distillation loss function is defined as follows:

$$\mathcal{L}_{\text{ATL}}(p_i^t, p_i^s) = \tau_i^2 \text{KL}(p_i^t / \tau_i \| p_i^s / \tau_i) \quad (10)$$

here, p_i^t and p_i^s denote the output probability distributions for sample i produced by the teacher model and the student model, respectively. $\text{KL}(\cdot \| \cdot)$ denotes the Kullback-Leibler divergence, which is used to measure the difference between the output distributions of the teacher and student models. τ_i is the adaptive temperature parameter calculated using Equation (9), which is dynamically adjusted according to the confusion degree w_x of the sample.

3.3. Training Objective

As illustrated by the overall framework in **Figure 1**, the SAT training procedure introduced in this study is primarily composed of two stages, which are detailed below:

1) Stage 1: Fine-tuning of the teacher model

Before knowledge distillation, the teacher model needs to be fine-tuned according to the task-related knowledge so that it can subsequently transfer this knowledge to the student model. Previously, the teacher model used the cross-entropy loss function to adjust its parameters. However, because the task contains samples with different levels of difficulty, the teacher model should focus on samples with different confidence levels at different stages. Therefore, during the fine-tuning process, this paper adopts the segment-wise focal loss as the training objective.

Given the training sample set $\{(x_i, y_i)\}_{i=1}^N$, where x_i denotes the input sample and y_i denotes its corresponding ground-truth label, the teacher model outputs a class-probability distribution $p_i^t = f_{\theta_t}(x_i)$, where θ_t denotes the parameters of the teacher model. The optimization objective of the teacher model is defined as minimizing the periodic focal loss:

$$\begin{aligned} \min_{\theta_t} L_{\text{SFL}}(p^t, y) &= \min_{\theta_t} (\xi L_{lc} + (1 - \xi) L_{hc}) \\ &= \min_{\theta_t} -\frac{1}{N} \left[\xi \left(-\left(1 - p_i^t\right)^{\gamma_{lc}} y_i \log(p_i^t) \right) + (1 - \xi) \left(-\left(1 + p_i^t\right)^{\gamma_{hc}} y_i \log(p_i^t) \right) \right] \end{aligned} \quad (11)$$

here, N denotes the total number of samples. L_{SFL} consists of two components: the low-confidence focal loss L_{lc} and the high-confidence focal loss L_{hc} . The periodic parameter ξ guides the model to adjust its attention to easy and difficult samples at different stages. To balance the training weights between hard and easy samples during the fine-tuning phase of the teacher model, the period factor ℓ_c is set to 2. This allows the parameter ξ to be dynamically adjusted throughout the full training cycle: the first quarter of the epoch focuses on easy samples, the middle half focuses on hard samples, and the final quarter focuses on easy samples again.

2) Stage 2: Training the student model

After the teacher model has been fine-tuned for the specific task, its knowledge

is transferred to the student model. The student model is usually smaller, and its training objective mainly consists of one part: imitating the prediction results of the teacher model.

Similar to the teacher model, given the same training sample set $\{(x_i, y_i)\}_{i=1}^N$, where x_i denotes the input sample and y_i denotes its corresponding ground-truth label, the student model outputs a class probability distribution

$p_i^s = f_{\theta_s}(x_i)$, where θ_s denotes the parameters of the student model. To optimize the student model, two main loss functions are designed: the periodic focal loss and the adaptive temperature confusion loss. The overall objective is defined by combining these two loss functions, enabling the student model to learn from the ground-truth labels while acquiring knowledge from the teacher model's soft labels. The specific loss function is defined as follows:

$$\min_{\theta_s} L_{\text{SAT}}(p^s, y) = \min_{\theta_s} (\alpha L_{\text{SFL}}(p^s, y) + (1 - \alpha) L_{\text{ATL}}(p^t, p^s)) \quad (12)$$

here, α is a weighting coefficient used to balance the contributions of the two losses. During training, the student model updates its parameters θ_s through backpropagation to minimize the overall loss L_{SAT} . Based on the updated student-model parameters, the process is repeated until the student model converges. Within the knowledge distillation framework, to prevent the student model from overfitting simple samples, this mechanism directs the model to learn simple samples briefly during the early stages of training before shifting its learning focus to difficult samples, thereby enhancing the model's generalization performance. To this end, this paper adopts a periodicity factor of $f_c = 4$ for the periodic focal loss. This configuration enables the model to dynamically shift its learning focus throughout the training cycle: prioritizing simple samples during the first one-eighth of the cycle, focusing on hard samples during the middle five-eighths, and returning to learning simple samples during the final quarter.

4. Method Validation on Natural Language Understanding Task

To verify the effectiveness of the proposed method on natural language understanding tasks, we adopt the BERT model as the baseline and conduct systematic experiments on the GLUE benchmark dataset. This section describes the experimental setup in detail, including datasets, evaluation metrics, implementation details, compared methods and experimental results.

4.1. Experiment Environment

All experiments are performed on a machine with an Intel Xeon E3-1231 CPU and an NVIDIA RTX 3060 GPU. We implement the proposed method and baselines in PyTorch using PyCharm.

4.2. Datasets and Metrics

This work validates the proposed approach on a suite of tasks from the General

Language Understanding Evaluation (GLUE) benchmark [28]. The benchmark consists of single-sentence tasks including [29] and SST-2 [30], natural language inference tasks covering MNLI [31], QNLI [32], and RTE [33], as well as similarity and paraphrase tasks such as MRPC [34], STS-B [35], and QQP [36]. Multiple metrics are adopted to measure model performance: accuracy is reported for MNLI, SST-2, QNLI, QQP and RTE; F1-score for MRPC; Matthews correlation coefficient for CoLA; and Spearman rank correlation coefficient for STS-B. This combination of metrics enables a holistic assessment of our method. Detailed descriptions of these datasets are presented in **Table 1**.

Table 1. Details of GLUE datasets.

Task	#Train	#Dev	#Test	#Label	Metrics
Single-Sentence Tasks					
CoLA	8.5 k	1 k	1 k	2	Matthews corr
SST-2	67 k	872	1.8 k	2	Accuracy
Inference Tasks					
MNLI	393 k	20 k	20 k	3	Accuracy
QNLI	108 k	5.7 k	5.7 k	2	Accuracy
RTE	2.5 k	276	3 k	2	Accuracy
Similarity and Paraphrase Tasks					
MRPC	3.7 k	408	1.7 k	2	F1
STS-B	7 k	1.5 k	1.4 k	1	Spearman corr
QQP	364 k	40 k	391 k	2	Accuracy

Furthermore, to further analyze the effect of our method on mitigating the imbalance between easy and hard samples as well as prediction uncertainty, we adopt multiple evaluation metrics for comparison. Among them, Top-1 denotes the prediction accuracy of the category with the fewest samples in the training set; a higher value indicates stronger capability of the model to learn hard samples. In addition to Top-1, we introduce the Area Under Risk-Coverage Curve (AURC, $\times 10^3$). A lower AURC implies that the model can more effectively distinguish correct predictions from incorrect ones, thereby alleviating the impacts brought by high-uncertainty samples.

4.3. Implementation Details

Stage 1: This paper uses the pretrained BERT-base-uncased model as the teacher model. Its architecture consists of a 12-layer Transformer encoder, with a hidden-layer dimension of 768 and approximately 110M parameters. To adapt the model to different tasks, it is fine-tuned separately on six specific tasks from the GLUE

benchmark. During fine-tuning, the learning rate is uniformly set to $3e-5$, and the training period is set to 20 epochs to ensure that the model can learn and converge sufficiently on each task.

In addition, two new hyperparameters, γ_{lc} and γ_{hc} , are introduced into the periodic focal loss. According to the study by Smith *et al.* [20], the focal-loss parameter γ_{lc} is not highly sensitive, and setting $\gamma_{lc} = 4$ can achieve good performance on most tasks. Therefore, this paper sets $\gamma_{lc} = 4$. Furthermore, experimental results show that the model generally performs well when γ_{hc} is set to 2 or 3.

Stage 2: This paper uses four-layer BERT models, BERT_{base} (4/768; 52 M) as student models. Each student model is trained on a specific task using the corresponding teacher model fine-tuned in Stage 1, and knowledge distillation is performed according to Equation (12). The learning rate is selected from $\{2e-5, 3e-5, 4e-5\}$. The hyperparameter α is selected from $\{0.3, 0.4, 0.5\}$, while the temperature parameters are selected from $\tau_{max} \in \{6, 8, 10\}$ and $\tau_{min} \in \{3, 4, 5\}$. In addition, for the two hyperparameters γ_{lc} and γ_{hc} in the periodic focal loss, the same settings as those used during teacher-model fine-tuning are adopted: γ_{lc} is fixed at 4, while γ_{hc} is selected as either 2 or 3.

4.4. Comparison Methods

To validate the effectiveness of our method, we adopt multiple well-performed knowledge distillation methods for NLP tasks as baselines.

- Vanilla KD [4] employs basic knowledge distillation methods that transfer knowledge through the prediction distributions.
- TinyBERT [9] adopts layer-wise distillation based on hidden representations and self-attention matrices. It proposes two distillation stages—general learning and task-specific learning—and employs a data augmentation strategy.
- CKD [37] and MGSKD [38] extract pairwise and triplet structural relationships between the hidden representations of the teacher and student models, and then calculate the corresponding loss.
- MiniLMv2 [39] first performs task-agnostic distillation and then fine-tunes the model on task-specific datasets. By introducing multi-head self-attention relations, it allows greater diversity in the number of attention heads used by the student model.

Note that we re-implement several methods (MGSKD, MiniLMv2) using their original codes for fair comparisons.

4.5. Main Results

This paper compresses the model size from 110 M to 52 M parameters, yielding the student model SAT_{Small}. Its overall accuracy on the GLUE benchmark is presented in Table 2. Compared with the teacher model BERT-base, SAT_{Small} reduces the model size by 52.7%, while its average performance decreases by only 3.2%, demonstrating excellent model compression effectiveness.

Detailed comparisons with other student models show that SAT_{Small} achieves superior performance on the majority of tasks, with an average accuracy of 80.8%. This represents improvements of 1.8% over BERT_{Small}, 2.1% over TinyBERT₄, 3.3% over MiniLMv2₄, 3.8% over CKD₄, and 2.8% over MGSKD₄. In particular, SAT_{Small} achieves the best results on the sentiment classification task SST-2, the semantic matching task MRPC, and the natural language inference task QNLI, with accuracies of 90.0%, 90.6%, and 90.4%, respectively. These results demonstrate clear performance gains over the other student models.

Across all evaluated tasks, SAT_{Small} achieves its most significant improvement on the MRPC dataset, attaining an F1 score of 90.6%. This is 1.0% higher than BERT_{Small}, 4.8% higher than TinyBERT₄, 5.3% higher than MiniLMv2₄, 8.0% higher than CKD₄, and 8.2% higher than MGSKD₄, representing a substantial improvement over the baseline methods. These results demonstrate that SAT_{Small} has stronger representational capability on tasks that rely on capturing semantic similarity and learning fine-grained features.

Table 2. Comparative results show that the student model obtained in this study reduces the number of model parameters from 110 M to 52 M. The optimal value of the student model is shown in bold.

Model	#Params	CoLA	MNLI (m/mm)	SST-2	QNLI	MRPC	QQP	RTE	STS-B	Avg
		Mcc (%)	Acc (%)	Acc (%)	Acc (%)	F1 (%)	Acc (%)	Acc (%)	Spear (%)	(%)
BERT _{base} (Teacher)	110 M	60	84.8/84.6	92.9	91.5	92	91.3	70.6	89.1	84
BERT _{Small} (Student)	52 M	45	79.9/80.2	89	88.3	89.6	89.4	63.2	87.4	79
TinyBERT ₄	52 M	50.5	80.8 /80.1	89.6	87.7	85.8	85.3	64.9	85.3	78.7
MiniLMv2 ₄	52 M	50.6	79.7/80.4	90.3	85.5	85.3	80.2	63.1	85.3	77.5
CKD ₄	52 M	50.1	79.9/79.3	89.4	85.1	82.6	84.2	62.3	82.8	77
MGSKD ₄	52 M	49.8	78.9/80.1	89.9	85.2	82.4	88.5	63.1	86.1	78
SAT _{Small}	52 M	50.9	80.1/ 80.9	90.0	90.4	90.6	90.9	63.9	89.5	80.8

Table 3 presents the performance of our proposed student model SAT_{Small}. The model compresses the parameter count from 110M to 52M and achieves notable improvements on hard-sample classification tasks across all datasets of the GLUE benchmark. Compared with the teacher model BERT-base, SAT_{Small} only suffers a 3.7% drop in Top-1 accuracy (78.6% vs. 82.3%), indicating that it can retain performance close to the teacher model even with drastic parameter compression. When compared with the undistilled student model BERT_{Small} (74.4%), SAT_{Small} obtains a 4.2% accuracy gain, which further verifies its effectiveness for model compression while preserving performance on hard samples.

For individual tasks, SAT_{Small} achieves an accuracy of 88.8% on QQP, outperforming BERT_{Small} (86.4%) and achieving comparable results to the state-of-the-art MiniLMv2₄ (90.3%). Although the positive and negative samples in QQP are relatively balanced, the promising performance of SAT_{Small} demonstrates its strong robustness when handling class imbalance. The model can effectively capture hard-to-distinguish semantic patterns and improve prediction performance for minority classes. On the MRPC semantic matching task, SAT_{Small} reaches an F1 score of 90.2%, substantially exceeding BERT_{Small}'s 84.6%. The MRPC dataset contains many low-frequency non-equivalent sentence pairs. This result further proves that knowledge distillation enables SAT_{Small} to boost prediction accuracy for hard and minority-class samples.

In summary, SAT_{Small} achieves strong Top-1 performance. Especially on hard-sample tasks such as QQP and MRPC, it delivers better classification ability than the undistilled BERT_{Small}. The experiments demonstrate that SAT_{Small} achieves remarkable advantages in both model compression and hard-sample classification.

Table 3. The comparison results indicate that the second student model obtained in this paper compresses the model parameters from 110 M to 52 M and achieves Top-1 prediction accuracy.

Model	#Params	CoLA	MNLI (m/mm)	SST-2	QNLI	MRPC	QQP	RTE	STS-B	Avg
		Mcc (%)	Acc (%)	Acc (%)	Acc (%)	F1(%)	Acc (%)	Acc (%)	Spear (%)	(%)
BERT _{base} (Teacher)	110 M	62.4	82.8/83.6	92.5	80.6	87.9	93.5	69.6	89.2	82.3
BERT _{Small} (Student)	52 M	44	71.9/70.2	89	73.8	84.6	86.4	61.2	84.4	74.4
TinyBERT ₄	52 M	39.5	69.8/ 80.1	89.6	77.7	84.8	86.3	62.9	84.3	74.4
MiniLMv2 ₄	52 M	39.6	68.7/70.4	90.6	75.5	84.3	87.2	62.1	85.3	74.2
CKD ₄	52 M	41.1	68.9/69.3	89.3	76.1	81.6	85.2	62.8	85.8	73.9
MGSKD ₄	52 M	42.8	67.9/70.1	91.1	75.2	83.4	86.5	62.1	86.1	74.4
SAT _{Small}	52 M	47.7	76.6 /78.6	90.4	80.4	90.2	88.8	64.7	89.8	78.6

Table 4 evaluates the ability of the 4×768 student model to distinguish correct and incorrect predictions, *i.e.*, the model's capability to quantify prediction uncertainty, via AURC (Area Under the Risk-Coverage Curve). Experimental results demonstrate that the SAT_{Small} student model achieves outstanding performance in uncertainty estimation, with an average AURC of 50.1. This value is markedly lower than that of the undistilled BERT_{Small} (average AURC = 55.4) and close to the performance of the teacher model BERT-base (average AURC = 45.9).

Specifically, on the SST-2 task, SAT_{Small} achieves an AURC of 59.2, which is substantially lower than BERT_{Small}'s 64.2 and close to the teacher model's 51.5. For the

RTE task, SAT_{Small} obtains an AURC of 55.7, outperforming BERT_{Small} (61.2) and approaching the teacher model's 55.6. These observations indicate that SAT_{Small} effectively learns the teacher model's ability to quantify prediction uncertainty through knowledge distillation. While reducing parameter size, the model significantly improves its capability to identify wrong predictions. In contrast, without supervision from the teacher model, the undistilled BERT_{Small} yields generally higher AURC values, which reflects its weaker uncertainty estimation performance.

Overall, SAT_{Small} not only narrows the performance gap with the teacher model on uncertainty estimation tasks but also surpasses the undistilled student model by a notable margin. The results verify that by introducing the adaptive temperature distillation loss, the SAT method successfully alleviates the "error inheritance" problem from the teacher model during knowledge distillation.

Table 4. The ability of the student model to discriminate between correct and incorrect predictions was assessed by AURC.

Model	#Params	CoLA	MNLI (m/mm)	SST-2	QNLI	MRPC	QQP	RTE	STS-B	Avg
BERT _{base} (Teacher)	110 M	41.4	40.8/41.6	51.5	48.5	36.9	43.5	55.6	49.2	45.9
BERT _{Small} (Student)	52 M	50.2	52.9/50.2	64.2	55.3	44.6	56.4	61.2	58.4	55.4
TinyBERT ₄	52 M	50.5	50.8/50.1	61.6	57.7	44.8	56.3	62.9	60.3	55.6
MiniLMv2 ₄	52 M	49.6	49.7/50.4	62.3	55.5	44.3	57.2	62.1	59.3	55
CKD ₄	52 M	48.1	49.9/49.3	63.4	56.1	41.6	55.2	62.8	58.8	54.5
MGSKD ₄	52 M	46.8	48.9/50.1	63.9	55.2	43.4	56.5	63.1	59.1	54.6
SAT _{Small}	52 M	45.6	46.6/47.6	59.2	49.1	39.2	48.8	55.7	56.8	50.1

5. Method Validation on Language Modeling Task

To verify that the proposed SAT method can mitigate the negative impacts of easy-hard sample imbalance and prediction uncertainty on model performance in language modeling tasks, this paper adopts GPT-2 as the baseline model. With self-supervised pre-training, the model gradually learns the ability to generate text sequences on large-scale text datasets and improves its understanding of semantics and grammar. Built upon conventional knowledge distillation, we introduce periodic focal loss to dynamically adjust the attention weights assigned to easy and hard samples at different training stages. Meanwhile, an adaptive temperature distillation loss is adopted to customize temperature parameters for each sample, distinguishing between correct and incorrect predictions. To comprehensively evaluate the effectiveness of the SAT method, we conduct systematic experiments under two typical scenarios: zero-shot learning and transfer learning.

5.1. Datasets and Metrics

To verify the language modeling capability of the small language model, we perform pre-training distillation on the Open WebText dataset. This dataset contains

approximately 8 million training documents with 38 GB of text, which is suitable for large-scale natural language processing tasks. Furthermore, we further evaluate the generalization ability of the model through zero-shot task evaluation and transfer learning experiments. The main datasets used include LAMBADA [40] (for evaluating text comprehension and generation, with perplexity PPL and accuracy ACC as metrics), WikiText-103 [41] (for evaluating language modeling performance, with perplexity PPL as the metric), and The Children's Book Test [42] (for evaluating text comprehension and generation, with accuracy ACC as the metric).

Perplexity (PPL) is a commonly used evaluation metric for language models [26], primarily used to measure a model's ability to predict text sequences. PPL reflects the model's average degree of uncertainty on test data; a smaller value indicates stronger predictive ability.

For a given text sequence $X = (x_1, x_2, \dots, x_n)$, perplexity is calculated as:

$$\text{PPL} = \exp\left(-\frac{1}{n} \sum_{i=1}^n \log P(x_i | x_1, x_2, \dots, x_{i-1})\right) \quad (13)$$

where n denotes the total length of the text sequence, and $P(x_i | x_1, x_2, \dots, x_{i-1})$ denotes the model's conditional probability prediction for the i -th word. Perplexity measures a language model's ability to predict a text sequence. A smaller value indicates greater confidence when the model generates or predicts text. Generally, a lower PPL means that the text generated by the language model is closer to real-world text.

5.2. Implementation Details

Within the framework of knowledge distillation for language modeling, the distillation process utilizes a large-scale general-purpose text corpus (specifically, the Open WebText dataset) with the aim of transferring the general language modeling knowledge acquired by the teacher model—such as implicit knowledge regarding contextual dependencies and word probability distributions—to the student model.

Stage 1: This paper uses a pretrained 12-layer GPT-2 model as the teacher model. Each hidden layer has a dimension of 768, and the model contains 125M parameters. During fine-tuning, the learning rate is fixed at $3e-5$, and the training period is set to 10 epochs to ensure that the model can learn sufficiently and reach convergence on each task.

For the two hyperparameters in the periodic focal loss, the teacher model continues to use the same fine-tuning settings for natural language understanding tasks: $\gamma_{lc} = 4$, while γ_{hc} is set to either 2 or 3. As with natural language understanding tasks, the period factor f_c is set to 2.

Stage 2: This paper uses a 4-layer GPT-2 (4/312, 19.6 M) model as the student model and performs distillation from the teacher model after it has been fine-tuned on the specific task, following Equation (12). The learning rate is selected from $\{2e-5, 3e-5, 4e-5, 5e-5\}$, with the best configuration being chosen. The hyperparameter α

is selected from $\{0.4, 0.5, 0.6, 0.7\}$, while the two temperature hyperparameters in Equation (9) are selected from $\tau_{\max} \in \{6, 8, 10\}$ and $\tau_{\min} \in \{3, 4, 5\}$.

For the two hyperparameters in the periodic focal loss, γ_{lc} and γ_{hc} , the same settings as those used during teacher-model fine-tuning are adopted. The low-confidence hyperparameter is fixed at $\gamma_{lc} = 4$, while the high-confidence hyperparameter γ_{hc} is selected as either 2 or 3. In contrast to the training weight allocation for natural language processing tasks—where simple and difficult samples are treated equally for the student model—the period factor f_c is set to 2.

5.3. Comparison Methods

To verify the effectiveness of the proposed method, the following approaches are selected for comparison:

- Vanilla KD [4] is trained by minimizing the discrepancy between the probability distributions output by the student and teacher models.
- Layer-wise Distillation (LWD) matches the hidden vectors at each layer and optimizes the student model based on the consistency of hidden-layer features.
- TED [43] uses a fully connected layer to select hidden features and calculates the knowledge discrepancy between the selected features and the corresponding teacher-model features.
- CI-WA [44] employs a dynamic extractor to obtain word-level correlations from each sample and transfers them as knowledge to the student model. It further enhances low-level feature representations through a top-down cross-layer integration mechanism.
- A2D [45] aligns attention heads to avoid heuristic selection of layer mappings.

5.4. Main Results

Table 5 presents the zero-shot learning and transfer learning performance of the GPT-2₄ model on the test set, compared with the teacher model GPT-2₁₂. The experimental results show that the teacher model performs well on both zero-shot and transfer learning tasks, whereas the baseline student model (GPT-2₄) performs poorly without knowledge distillation. This is particularly evident in the zero-shot learning task on the LAMBADA dataset, where its perplexity (PPL) reaches 99.8 and its accuracy is only 16.4%.

In the zero-shot learning task, SAT achieves the best overall performance across the three datasets. This indicates that dynamically emphasizing sample difficulty at different stages and appropriately adjusting inter-class differences can help the model better adapt to unseen classes or tasks.

Through knowledge distillation (KD), layer-wise distillation (LWD), task-related distillation (TED), word-level association knowledge distillation (CI-WA), and self-adaptive temperature periodic knowledge distillation (SAT), the performance of the student model is significantly improved. For example, GPT-2₄ (SAT) achieves a transfer-learning accuracy of 80.3% on the CBT dataset, the highest among all methods.

Table 5. The zero-shot and transfer learning performance of the GPT-2₄ model on the test set.

Method	Zero-Shot				Transfer Learning			
	LAMBADA		WikiText-103	CBT	LAMBADA		WikiText-103	CBT
	PPL	Acc (%)	PPL	Acc (%)	PPL	Acc (%)	PPL	Acc (%)
GPT-2 ₁₂ (Teacher)	35.1	46	37.5	83.4	37.2	34.8	15.9	85.1
GPT-2 ₄ (Student)	99.8	16.4	70.1	65.8	63.2	27.9	35.7	69
GPT-2 ₄ (KD)	91.9	21.9	54.8	75.1	51.5	30.7	34.2	70.4
GPT-2 ₄ (LWD)	90.8	22	51.7	72.8	50.4	30.5	35.9	75.2
GPT-2 ₄ (TED)	89.5	22.4	47.2	73.3	49.4	30.4	31.6	78.6
GPT-2 ₄ (A2D)	87.9	22.1	48.1	74.9	49.6	31.6	31.8	78.9
GPT-2 ₄ (CI-WA)	88.2	21.8	47.2	75.4	48.2	31	29.8	79
GPT-2 ₄ (SAT)	87.1	22.3	46.1	76.8	49.7	33.8	30.0	80.3

6. Discussion

6.1. Effectiveness of Segment-Wise Focal Loss

This ablation study verifies the effectiveness of segment-wise focal loss for knowledge distillation on multiple NLU tasks. The results are presented in **Table 6**. The undistilled baseline Student4 underperforms Teacher12 severely, reaching only 63.2% on RTE and 89.6% on MRPC and struggling with imbalanced low-frequency samples. Vanilla KD brings moderate gains, lifting SST-2 from 89% to 90.6%.

FL + KD further improves RTE and MRPC to 62.9% and 90%. Our SFL + KD achieves the best or near-best results across all tasks, with 91.2% accuracy on SST-2 and 50.9% MCC on CoLA. Segment-wise focal loss dynamically reweights samples and boosts prediction accuracy for hard and uncertain instances.

Table 6. The impact of segment-wise focal loss on the BERT4 student model in balanced and imbalanced datasets.

Methods	Balanced Test Sets			Imbalanced Test Sets		
	SST-2 (%)	RTE (%)	MNLI-(m/mm) (%)	QNLI (%)	CoLA (%)	MRPC (%)
Teacher12	92.9	70.6	84.8/84.6	91.5	60	92
Student4	89	63.2	79.9/80.2	88.3	45	89.6
KD	90.6	62.1	79.2/79.1	89.7	41.5	89.2
CrossEntropy + KD	90.0	62.8	72.4/73.5	89.5	43.2	89.5
FL + KD	90.2	62.9	72.9/73.2	89.7	45.1	90
SFL + KD	91.2	63.9	80.1/80.9	90.4	50.9	90.6

6.2. Mitigating Dataset Imbalances

To investigate the effects of segment-wise focal loss (SFL) and adaptive temperature distillation loss (ATL) on imbalanced datasets, this paper conduct experiments on QNLI, CoLA and MRPC. Following Kim *et al.* [46], besides Top-1 and AURC, we also report Expected Calibration Error (ECE, %). Lower ECE indicates better calibration, where model confidence aligns closely with actual accuracy. Top-1 accuracy denotes the prediction accuracy of the majority class, which contains the largest number of samples in the dataset.

1) ECE

Expected Calibration Error (ECE) is a commonly used model calibration metric, primarily used to measure the agreement between a classification model's predicted confidence and its actual accuracy. ECE reflects whether the model tends to be overly confident or overly conservative in its predictions. It divides the model's predicted confidence scores into M bins, and then calculates the difference between the average confidence and accuracy within each bin. The formula is:

$$ECE = \sum_{m=1}^M \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)| \quad (14)$$

where B_m denotes the sample set in the m -th bin, $|B_m|$ denotes the number of samples in the m -th bin, n denotes the total number of samples, $\text{acc}(B_m)$ denotes the average accuracy of the samples in the m -th bin, and $\text{conf}(B_m)$ denotes the average confidence of the samples in the m -th bin. The value of ECE ranges from $[0, 1]$. A smaller value indicates that the model's confidence is closer to the actual probability, suggesting better calibration.

2) AURC

AURC (Area Under the Reliability Curve) is a metric used to measure the reliability of a classification model's predictions. It is typically used to evaluate whether the model's confidence is consistent with its actual performance. Similar to Expected Calibration Error (ECE), AURC focuses on the relationship between the model's predicted confidence and its actual accuracy. The formula for calculating AURC is:

$$AURC = \int_0^1 \text{acc}(\text{conf}) d\text{conf} \quad (15)$$

where $\text{acc}(\text{conf})$ is the actual accuracy at a given confidence level conf . The value of AURC generally ranges from 0 to 1. A value closer to 1 indicates a larger discrepancy between the model's confidence and its actual accuracy, suggesting that the model's predictions are unreliable. Conversely, an AURC value closer to 0 indicates that the model's predicted confidence is consistent with its actual accuracy, suggesting that the model is relatively reliable.

As shown in **Table 7** for QNLI, either SFL or ATL alone improves Top-1 accuracy but yields limited gains for ECE and AURC. Combining SFL and ATL (SAT) achieves the best results across all metrics: Top-1 accuracy of 80.4%, ECE of 2.39%,

and AURC of 49.1. This verifies the complementarity and effectiveness of SFL and ATL in knowledge distillation.

Table 7. Comparative experiments of the student model BERT4 on the QNLI dataset.

Methods	SFL	ATL	Top-1	ECE	AURC
Abl.1	×	×	76.8%	11.87%	67.6
Abl.2	×	√	79.2%	10.81%	57.2
Abl.3	√	×	77.6%	6.21%	61.7
SAT	√	√	80.4%	2.39%	49.1

7. Conclusions and Future Work

To address the imbalance between easy and hard samples in training data, this chapter proposes a periodic segment-wise focal loss knowledge distillation method with adaptive temperature. Built upon conventional knowledge distillation, it contains two improvements. The segment-wise focal loss dynamically adjusts the weights for high- and low-confidence samples, enabling the model to learn comprehensive knowledge across different training stages. The adaptive temperature distillation loss dynamically tunes the temperature according to sample difficulty to scale inter-class differences and boost distillation performance. By mitigating the easy-hard sample imbalance, this method optimizes the distillation pipeline. Experiments demonstrate that the proposed approach effectively reduces the adverse impact of sample imbalance and improves the quality and generalization of the compressed model.

In this work, sample difficulty is characterized mainly based on model confidence. Future work can incorporate multiple factors such as semantic complexity of samples, class distribution, and prediction discrepancy between the teacher and student models to construct a more accurate and robust sample difficulty evaluation mechanism, so as to achieve finer-grained knowledge allocation.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

- [1] Devlin, J., Chang, M., Lee, K. and Toutanova, K. (2019) BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *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)*, Minneapolis, 2 June-7 June 2019, 4171-4186. <https://doi.org/10.18653/v1/n19-1423>
- [2] Liu, Y., Ott, M., Goyal, N., *et al.* (2019) RoBERTa: A Robustly Optimized BERT Pre-training Approach. arXiv:1907.11692.
- [3] Brown, T.B., Mann, B., Ryder, N., *et al.* (2020) Language Models Are Few-Shot Learners. *Advances in Neural Information Processing Systems*, **33**, 1877-1901.

- [4] Hinton, G., Vinyals, O. and Dean, J. (2015) Distilling the Knowledge in a Neural Network. arXiv:1503.02531.
- [5] Buciluă, C., Caruana, R. and Niculescu-Mizil, A. (2006) Model Compression. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Philadelphia, 20-23 August 2006, 535-541. <https://doi.org/10.1145/1150402.1150464>
- [6] Romero, A., Ballas, N., Kahou, S.E., *et al.* (2015) FitNets: Hints for Thin Deep Nets. arXiv:1412.6550.
- [7] Gou, J., Yu, B., Maybank, S.J., *et al.* (2021) Knowledge Distillation: A Good Teacher Is Patient and Consistent. *Pattern Recognition*, **116**, Article 107880.
- [8] Sanh, V., Debut, L., Chaumond, J., *et al.* (2019) DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. arXiv:1910.01108.
- [9] Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., *et al.* (2020) TinyBERT: Distilling BERT for Natural Language Understanding. *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online, 16-20 November 2020, 4163-4174. <https://doi.org/10.18653/v1/2020.findings-emnlp.372>
- [10] Müller, R., Kornblith, S. and Hinton, G.E. (2019) When Does Label Smoothing Help? *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Vancouver, 8-14 December 2019, 4696-4705.
- [11] Malinin, A. and Gales, M. (2018) Predictive Uncertainty Estimation via Prior Networks. *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montréal, 3-8 December 2018, 7047-7058.
- [12] Lakshminarayanan, B., Pritzel, A. and Blundell, C. (2017) Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, 4-9 December 2017, 6405-6416.
- [13] Yang, C., An, Z., Cai, L., *et al.* (2023) Knowledge Distillation Using Neural Network Ensembles. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Vancouver, 1383-1392.
- [14] Zagoruyko, S. and Komodakis, N. (2017) Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. *International Conference on Learning Representations*, Toulon, 1-13.
- [15] Park, W., Kim, D., Lu, Y. and Cho, M. (2019) Relational Knowledge Distillation. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, 15-20 June 2019, 3967-3976. <https://doi.org/10.1109/cvpr.2019.00409>
- [16] Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y. and Zhou, D. (2020) MobileBERT: A Compact Task-Agnostic BERT for Resource-Limited Devices. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, 5-10 July 2020, 2158-2170. <https://doi.org/10.18653/v1/2020.acl-main.195>
- [17] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N. and Zhou, M. (2020) MINILM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers. *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Vancouver, 6-12 December 2020, 5776-5788.
- [18] Lin, T., Goyal, P., Girshick, R., He, K. and Dollar, P. (2017) Focal Loss for Dense Object Detection. 2017 *IEEE International Conference on Computer Vision (ICCV)*, Venice, 22-29 October 2017, 2999-3007. <https://doi.org/10.1109/iccv.2017.324>
- [19] Ridnik, T., Ben-Baruch, E., Zamir, N., Noy, A., Friedman, I., Protter, M., *et al.* (2021) Asymmetric Loss for Multi-Label Classification. 2021 *IEEE/CVF International Con-*

- ference on Computer Vision (ICCV)*, Montreal, 10-17 October 2021, 82-91. <https://doi.org/10.1109/iccv48922.2021.00015>
- [20] Smith, L.N. (2023) General Cyclical Training of Neural Networks. *Advances in Artificial Intelligence and Machine Learning*, **3**, 958-976. <https://doi.org/10.54364/aaiml.2023.1157>
- [21] Bengio, Y., Louradour, J., Collobert, R. and Weston, J. (2009) Curriculum Learning. *Proceedings of the 26th Annual International Conference on Machine Learning*, Montreal, 14-18 June 2009, 41-48. <https://doi.org/10.1145/1553374.1553380>
- [22] Frankle, J., Schwab, D.J. and Morcos, A.S. (2020) The Early Phase of Neural Network Training. *Proceedings of the International Conference on Learning Representations (ICLR)*, Addis Ababa, 1-20.
- [23] Smith, L.N. (2017) Cyclical Learning Rates for Training Neural Networks. 2017 *IEEE Winter Conference on Applications of Computer Vision (WACV)*, Santa Rosa, 24-31 March 2017, 464-472. <https://doi.org/10.1109/wacv.2017.58>
- [24] Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y. and He, K. (2017) Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, 472-481.
- [25] Loshchilov, I. and Hutter, F. (2017) SGDR: Stochastic Gradient Descent with Warm Restarts. arXiv:1608.03983.
- [26] Zhang, G., Wang, C., Xu, B. and Grosse, R. (2019) Three Mechanisms of Weight Decay Regularization. arXiv:1810.12281.
- [27] Zhang, O., Wu, M., Bayrooti, J. and Goodman, N. (2021) Temperature as Uncertainty in Contrastive Learning. *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*, Sydney, 4 p.
- [28] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O. and Bowman, S. (2018) GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, Brussels, 1 November 2018, 353-355. <https://doi.org/10.18653/v1/w18-5446>
- [29] Warstadt, A., Singh, A. and Bowman, S.R. (2018) Neural Network Acceptability Judgments. *Transactions of the Association for Computational Linguistics*, **7**, 625-641. https://doi.org/10.1162/tacl_a_00290
- [30] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., et al. (2013) Recursive Deep Models for Semantic Compositionality over a Sentiment Treebank. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Seattle, 18-21 October 2013, 1631-1642. <https://doi.org/10.18653/v1/d13-1170>
- [31] Williams, A., Nangia, N. and Bowman, S. (2018) A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, New Orleans, 1-6 June 2018, 1112-1122. <https://doi.org/10.18653/v1/n18-1101>
- [32] Rajpurkar, P., Zhang, J., Lopyrev, K. and Liang, P. (2016) Squad: 100,000+ Questions for Machine Comprehension of Text. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, 1-5 November 2016, 2383-2392. <https://doi.org/10.18653/v1/d16-1264>
- [33] Bentivogli, L., Clark, P., Dagan, I. and Giampiccolo, D. (2011) The Seventh PASCAL

- Recognizing Textual Entailment Challenge. *Proceedings of the 4th Text Analysis Conference*, Gaithersburg, 1-6.
- [34] Dolan, W.B. and Brockett, C. (2005) Automatically Constructing a Corpus of Sentential Paraphrases. *Proceedings of the Third International Workshop on Paraphrasing*, Korea, 9-16.
- [35] Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I. and Specia, L. (2017) Semeval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Vancouver, 3-4 August 2017, 1-14. <https://doi.org/10.18653/v1/s17-2001>
- [36] Chen, Z., Zhang, H., Zhang, X. and Zhao, L. (2018) Quora Question Pairs. <https://www.kaggle.com/c/quora-question-pairs>
- [37] Park, G., Kim, G. and Yang, E. (2021) Distilling Linguistic Context for Language Model Compression. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Punta Cana, 7-11 November 2021, 364-378. <https://doi.org/10.18653/v1/2021.emnlp-main.30>
- [38] Liu, C., Tao, C., Feng, J. and Zhao, D. (2022) Multi-Granularity Structural Knowledge Distillation for Language Model Compression. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, 22-27 May 2022, 1001-1011. <https://doi.org/10.18653/v1/2022.acl-long.71>
- [39] Wang, W., Bao, H., Huang, S., Dong, L. and Wei, F. (2021) MiniLMv2: Multi-Head Self-Attention Relation Distillation for Compressing Pretrained Transformers. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, Online, 1-6 August 2021, 2140-2151. <https://doi.org/10.18653/v1/2021.findings-acl.188>
- [40] Paperno, D., Kruszewski, G., Lazaridou, A., Pham, N.Q., Bernardi, R., Pezzelle, S., et al. (2016) The LAMBADA Dataset: Word Prediction Requiring a Broad Discourse Context. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, 7-12 August 2016, 1525-1534. <https://doi.org/10.18653/v1/p16-1144>
- [41] Merity, S., Xiong, C., Bradbury, J. and Socher, R. (2016) Pointer Sentinel Mixture Models. arXiv:1609.07843.
- [42] Hill, F., Bordes, A., Chopra, S. and Weston, J. (2016) The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations. arXiv:1511.02301.
- [43] Liang, C., Zuo, S., Zhang, Q., He, P., Chen, W. and Zhao, T. (2022) Less Is More: Task-Aware Layer-Wise Distillation for Language Model Compression. *International Conference on Machine Learning*, Honolulu, 20852-20867.
- [44] Li, G., Zheng, S., Zou, H., Yu, H. and Gao, S. (2025) Model Compression through Distillation with Cross-Layer Integrated Guidance at Word Level. *Neurocomputing*, **619**, Article 129162. <https://doi.org/10.1016/j.neucom.2024.129162>
- [45] Jin, H., Son, S., Park, J., Kim, Y., Noh, H. and Lee, Y. (2024) Align-to-Distill: Trainable Attention Alignment for Knowledge Distillation in Neural Machine Translation. *Proceedings of the Language Resources and Evaluation Conference*, Turin, 20-25 May 2024, 722-732. <https://doi.org/10.63317/496bh35o3ybj>
- [46] Kim, S. and Kim, H. (2017) Transferring Knowledge to Smaller Network with Class-distance Loss. *Proceedings of the International Conference on Learning Representations Workshop*, 3 p Extended Abstract.