Lightweight Contenders: Navigating Semi-Supervised Text Mining through Peer Collaboration and Self Transcendence

Qianren Mao[1,*], **Weifeng Jiang**[2,*], **Junnan Liu**[4], **Chenghua Lin**[3], **Qian Li**[4], **Xianqing Wen**[4], **Jianxin Li**[1,4,†], **Jinhu Lu**[1]

[1] Zhongguancun Laboratory, Beijing, P.R.China.
[2] CCDS, Nanyang Technological University, Singapore.
[3] Department of Computer Science, University of Manchester, UK.
[4] School of Computer Science and Engineering, Beihang University, Beijing, P.R.China.
maoqr@zgclab.edu.cn, weifeng001@e.ntu.edu.sg

Abstract

The semi-supervised learning (SSL) strategy in lightweight models requires reducing annotated samples and facilitating cost-effective inference. However, the constraint on model parameters, imposed by the scarcity of training labels, limits the SSL performance. In this paper, we introduce PS-NET, a novel framework tailored for semi-supervised text mining with lightweight models. PS-NET incorporates online distillation to train lightweight student models by imitating the Teacher model. It also integrates an ensemble of student peers that collaboratively instruct each other. Additionally, PS-NET implements a constant adversarial perturbation schema to further self-augmentation by progressive generalizing. Our PS-NET, equipped with a 2-layer distilled BERT, exhibits notable performance enhancements over SOTA lightweight SSL frameworks of FLiText and DisCo in SSL text classification with extremely rare labelled data.

1 Introduction

Deep and sizeable pre-trained language models (PLMs), such as BERT (Devlin et al., 2019) and GPT-3 (Radford et al., 2018; Brown et al., 2020) have exhibited impressive empirical performance in diverse natural language processing tasks. However, the considerable size of these PLMs poses challenges in terms of fine-tuning and online deployment due to latency and cost constraints. Additionally, the efficacy of PLMs in downstream tasks hinges on a fully supervised setup, necessitating abundant manually annotated datasets. Acquiring well-annotated labels is both costly and typically demands domain-knowledgeable professionals. The labour-intensive process of labelling each sentence is susceptible to errors arising from subjective human judgments.

Recent endeavours have been directed towards concurrently addressing two challenges in low-resource applications: ***how to effectively utilize compressed small models with limited labelled data to achieve model generalization***. To overcome the challenges of model size reduction and label data scarcity for PLMs, knowledge distillation is employed to compress an original large model (teacher) into a smaller counterpart. Subsequently, lightweight models can be optimized under semi-supervised learning (SSL) conditions using limited labelled data and an abundance of unlabeled data. However, two primary challenges emerge when employing the compressed small model for semi-supervised learning: ♣ the scarcity of labelled data samples provides insufficient supervision for the lightweight student model, impeding the acquisition of more nuanced task-specific knowledge, ♣ the smaller compressed model lacks generic regularization for semi-supervised learning, hindering the attainment of enhanced model generalization.

We present PS-NET[1], a semi-supervised text mining framework that refines lightweight student models using minimal labelled data for effective inference. PS-NET employs supervised optimization with task-specific labelled data, followed by online knowledge distillation to derive student cohorts from unlabeled data.

Within this semi-supervised framework, we incorporate an ensemble of lightweight student models that engage in reciprocal instruction, enabling collaborative optimization through mutual learning. We also introduce adversarial perturbations to gradually increase learning difficulty, promoting self-improvement in the students. In a nutshell, online distillation produces lightweight student networks, while mutual learning and adversarial perturbations refine the optimization process, helping avoid sub-

* Equal contribution.
† Jianxin Li is the corresponding author.

[1] Code and data available at: https://github.com/LiteSSLHub/PSNET.

Findings of the Association for Computational Linguistics: NAACL 2025, pages 5586–5600

optimal solutions and overcome optimization barriers. These processes enhance the generalization capabilities among lightweight student cohorts.

Extensive experiments across various semi-supervised text classification and semi-supervised extractive summarization tasks substantiate the exceptional performance of lightweight models within our PS-NET framework, even under the constraints of limited labelled data. PS-NET showcase fewer parameters (12.30× smaller) in comparison to the complete 12-layer BERT models. Additionally, our PS-NET, equipped with a 2-layer distilled BERT, surpasses competitors, specifically the state-of-the-art lightweight SSL frameworks FLiText and DisCo, by large margins in performance gains in text classification tasks that utilize only 10 labelled data samples per class.

2 Background & Related Work

2.1 Semi-Supervised Learning

Ongoing efforts are dedicated to mitigating the need for extensive human supervision through Semi-Supervised Learning (SSL) (Board and Pitt, 1989). The success of SSL methods in the visual domain (Sajjadi et al., 2016; Laine and Aila, 2017; Tarvainen and Valpola, 2017; Qiao et al., 2018; Miyato et al., 2019; Berthelot et al., 2019, 2022; Wang et al., 2023c; Chen et al., 2023) has spurred research interest in the NLP community. Noteworthy techniques, such as UDA (Xie et al., 2020), operate under the low-density separation assumption, enabling them to achieve comparable performances to fully supervised counterparts while utilizing only a fraction of labelled samples. Deep Mutual Learning (DML) (Board and Pitt, 1989; Zhang et al., 2018) facilitates knowledge transfer among diverse cohort models, demonstrating superior performance in category recognition domains like image classification tasks (Zhang et al., 2018; Park et al., 2020; Zhang et al., 2022a; Wang et al., 2023a). Park et al. (2020); Guo et al. (2023); He et al. (2024) show that diversity enhances generalization, and Jiang et al. (2023) confirm peer-teaching yields superior performance by collaborative learning among cohort models.

2.2 Faster and Lighter SSL

In recent years, heightened attention has been directed towards faster and lighter semi-supervised learning (SSL). FLiText (Liu et al., 2021) introduces an inspector network integrated with a consistency regularization framework. DisCo (Jiang et al., 2023) stands out as a notable framework. It utilizes a novel co-training technique to promote knowledge sharing among students through diverse data and model views. By employing meticulous data augmentation perturbations, such as adversarial attacks (Kurakin et al., 2017), token shuffling (Lee et al., 2020), cutoff (Shen et al., 2020), and dropout (Hinton et al., 2012), it achieves enhanced generalization capability. However, DisCo (Jiang et al., 2023) necessitates the deployment of large-scale offline models as external sources of knowledge.

3 Methodology

PS-NET applies supervised optimization with task-specific labelled data, followed by online knowledge distillation to generate student cohorts from unlabeled data in a phased semi-supervised framework. To strengthen the robustness of these lightweight models, PS-NET integrates ***peer collaboration*** and ***self-transcendence*** strategies. Student cohorts engage in mutual learning for collaborative optimization, where multiple students are trained together using complementary knowledge distilled from a shared teacher model. Additionally, we introduce adversarial perturbations (Zhang et al., 2022b; Chen et al., 2024) to progressively increase learning complexity among those lightweight models, fostering self-improvement and enhancing the generalization capabilities of the student cohorts.

We illustrate the dual-student PS-NET process for training two lightweight students (see Figure 1). Extending this to multiple students is straightforward, as detailed in Section 3.3. The PS-NET framework is outlined as follows:

- PS-NET involves semi-supervised learning (SSL) of knowledge optimization and knowledge distillation. This sequential approach establishes intermediate objectives that guide the optimization process between student and teacher models.
- PS-NET trains student cohorts through deep mutual learning (DML) by collaboratively mimicking each other's output logits. This enables the cohorts to exchange diversified knowledge, enhancing their generalization ability.
- PS-NET integrates curriculum adversarial training (CAT), which iteratively generates adversarial noise using gradient-based methods. These small perturbations in input embeddings promote

continuous self-improvement of student models, reducing their susceptibility to overfitting.

3.1 Knowledge Optimization Procedure

Formally, given a semi-supervised dataset $\mathcal{D}$, $\mathcal{D} = \mathcal{S} \cup \mathcal{U}$. $\mathcal{S} = \{(\hat{x}, \hat{y})\}$ is labeled data and $\mathcal{U} = \{x^*\}$ is unlabeled data, and both the student and teacher use all data identically. In supervised learning, we employ the cross-entropy loss for optimizing students f^S, and teacher f^T simultaneously with the labelled data $(\hat{x}, \hat{y})$ sampled from $\mathcal{S}$.

Before being input into the model, the data is augmented by the Curriculum Adversarial Noise Function ANF, as shown in Algorithm 1. For labelled data, adversarial noise is generated based on the ground truth labels. The difficulty of the adversarial examples is controlled in a step-wise manner based on the current training step n, gradually increasing the training difficulty from no augmentation to challenging adversarial examples:

$$\delta_T^S = \mathtt{ANF}(f^T, \hat{x}, \hat{y}, n), \tag{1}$$

$$\delta_S^S = \mathtt{ANF}(f^S, \hat{x}, \hat{y}, n), \tag{2}$$

$$\mathcal{L}_T = \sum_{(\hat{x},\hat{y})\in\mathcal{S}} \mathtt{CE}(f^T(\hat{x} + \delta_T^S), \hat{y}), \tag{3}$$

$$\mathcal{L}_S = \sum_{(\hat{x},\hat{y})\in\mathcal{S}} \mathtt{CE}(f^S(\hat{x} + \delta_S^S), \hat{y}). \tag{4}$$

3.2 Model Compression Procedure

In knowledge distillation, we utilize a uniform function to map the teacher (N layers) and student (M layers, $M < N$) through (i) the output of the embedding layer, (ii) the hidden states, and (iii) attention matrices. We set 0 to be the index of the embedding layer. We set $N+1$ and $M+1$ to be the index of the prediction layer for the teacher and student. Hence, the student can acquire knowledge from the teacher by minimizing the MSE objective:

$$\mathcal{L}_{F-KD} = \sum_{x^*\in\mathcal{U}} \sum_{m=0}^{M} \mathtt{MSE}\left(f_m^S(x^*), f_{g(m)}^T(x^*)\right), \tag{5}$$

where, $g(m)$ is defined as the mapping function between indices from student layers to teacher layers, indicating that the m-th layer of the student model emulates information from the $g(m)$-th layer of the teacher model. The mean squared error loss function (MSE) serves as the distance metric, measuring the similarity of two learned features. We distil the embedding layer, the hidden states, and attention matrices from teacher $f_{g(m)}^T$ to students f_m^S.

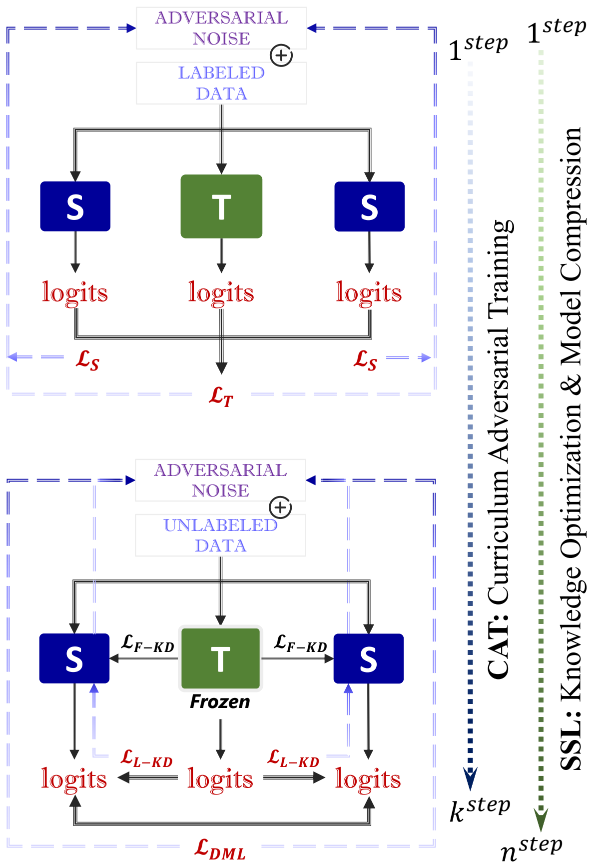


Figure 1: Framework of PS-NET. It integrates online distillation within an SSL framework, following phased steps of supervised knowledge optimization and unsupervised knowledge distillation. PS-NET allows the student networks to improve generalization through DML in a peer collaboration manner. In each step, PS-NET utilizes CAT, which iteratively generates adversarial noise using gradient-based methods, facilitating continuous self-improvement of the lightweight models.

In addition to imitating the behaviors of intermediate layers, we also use knowledge distillation to align the logits of the teacher. The penalty term $\mathcal{L}_{L-KD}$ is defined as the MSE loss between the logits of the student network's logits f_{M+1}^S against the teacher's logits f_{N+1}^T over the unlabeled data $\mathcal{U}$.

Similarly to labeled data, unlabeled data also need to be processed by ANF before being fed into the model, with difficulty controlled according to the current training step n. However, as shown in Algorithm 1, since there is no ground truth label for unlabeled data, we compute adversarial noise based on the model's prediction of the original data.

$$\delta_T^{\mathcal{U}} = \mathtt{ANF}(f^T, x^*, n), \tag{6}$$

$$\delta_S^{\mathcal{U}} = \mathtt{ANF}(f^S, x^*, n), \tag{7}$$

Algorithm 1 Calculation of the Curriculum Adversarial Noise Function ANF

Input: Model f, input embedding x, current training step n, curriculum step period λ_k, curriculum step factor γ, variance of the noise initialization σ^2, noise boundary ϵ, adversarial gradient ascent learning rate η, ground truth label y if labeled data is used. $\lfloor\ \rfloor$ denotes the floor function.

Output: Curriculum adversarial noise δ

1: $\delta \sim \mathcal{N}(0, \sigma^2)$
2: **for** $k \leftarrow 0$ to $\lfloor \frac{n}{\lambda_k} * \gamma \rfloor$ **do**
3:   **if** y exists **then**
4:     $\delta \leftarrow \delta + \eta \nabla_\delta \mathtt{CE}(f(x+\delta), y)$
5:   **else**
6:     $\delta \leftarrow \delta + \eta \nabla_\delta \mathtt{MSE}(f(x+\delta), f(x))$
7:   **end if**
8:   $\delta \leftarrow \Pi_{||\delta||_\infty \leq \epsilon}(\delta)$
9: **end for**
10: **return** δ

$$\mathcal{L}_{L-KD} = \sum_{x^* \in \mathcal{U}} \mathtt{MSE}\left(f^S_{M+1}\left(x^* + \delta^{\mathcal{U}}_S\right), f^T_{N+1}\left(x^* + \delta^{\mathcal{U}}_T\right)\right). \tag{8}$$

After the student cohorts mimic the behaviours of the teacher network from varied perspectives, they then engage in deep mutual learning for collaborative optimization. The student cohorts share diverse knowledge obtained from the teacher and learn from each other, compensating for individual shortcomings to ultimately emulate the teacher. Considering two students (S_1, S_2) as an example:

$$\mathcal{L}_{DML} = \sum_{x^* \in \mathcal{U}} \mathtt{MSE}\left(f^{S_1}_{M+1}\left(x^* + \delta^{\mathcal{U}}_{S_1}\right), f^{S_2}_{M+1}\left(x^* + \delta^{\mathcal{U}}_{S_2}\right)\right). \tag{9}$$

Overall Training Objective. Finally, we combine supervised knowledge optimization loss $\mathcal{L}^{\text{ko}}_\Theta$ and unsupervised model compression loss $\mathcal{L}^{\text{mc}}_\Theta$:

$$\mathcal{L}^{\text{ko}}_\Theta = \mathcal{L}_T + \mathcal{L}_S, \tag{10}$$

$$\mathcal{L}^{\text{mc}}_\Theta = \mathcal{L}_{KD} + \mu(t,n) \cdot \lambda \cdot \mathcal{L}_{DML}, \tag{11}$$

where $\mathcal{L}_{KD} = \mathcal{L}_{L-KD} + \mathcal{L}_{F-KD}$. The term $\mu(t,n) = min(\frac{n}{t}, 1)$, signifies the ramp-up weight initiating from zero and following a linear curve during the initial n training steps. The rationale behind incorporating a ramp-up is that the student models, initially initialized randomly, render their mutual learning ineffective. The hyperparameter λ balances mutual learning. The training of PS-NET involves minimizing the joint loss $\mathcal{L}^{\text{ko}}_\Theta + \mathcal{L}^{\text{mc}}_\Theta$.

3.3 Co-training of Multi-student Peers

PS-NET can seamlessly accommodate multiple students in the cohort. Considering K networks $\Theta_1, ..., \Theta_i, ..., \Theta_K (K \geq 2)$, the objective function for optimising all Θ_k, $(1 \leq k \leq K)$, becomes:

$$\mathcal{L}^{\text{ko}}_{\Theta_k} = \mathcal{L}_T + \sum_{k=1}^{K} \mathcal{L}^k_S, \tag{12}$$

$$\mathcal{L}^{\text{mc}}_{\Theta_k} = \sum_{k=1}^{K} \left(\mathcal{L}^k_{KD} + \mu(t,n) \cdot \lambda \cdot \mathcal{L}^k_{DML}\right), \tag{13}$$

$$\mathcal{L}^k_{DML} = \frac{1}{K-1} \sum_{i=1, i\neq k}^{K} \mathtt{MSE}(f^{S_k}_{M+1}(x^* + \delta^{\mathcal{U}}_{S_k}), f^{S_i}_{M+1}(x^* + \delta^{\mathcal{U}}_{S_i})). \tag{14}$$

Equation (11) is now a particular case of (13) with $k = 2$. When extending the cohort to include more than two networks, a learning strategy for each student of PS-NET takes the ensemble of other $K-1$ student peers to provide mimicry targets. In essence, each student learns individually from all other students within the cohort.

4 Experiments

4.1 Datasets

As shown in Table 1, we evaluate PS-NET on semi-supervised extractive summarization and semi-supervised text classification tasks.

In semi-supervised summarization, models are trained using 100 labelled examples from the CNN/DailyMail dataset (Hermann et al., 2015), with the remaining unlabeled examples serving as unsupervised data. We keep the standard splits of the target corpus for validation and testing. We conduct experiments on five semi-supervised text classification benchmarks AG News (Zhang et al., 2015) and Yahoo! Answers (Chang et al., 2008) for news classification, and DBpedia (Mendes et al., 2012) for topic classification, training models with 10, 30, and 200 labelled examples per class. We also use the USB benchmark to test Amazon (McAuley and Leskovec, 2013) and Yelp[2] review for SSL evaluation, training models with 50 and 200 labelled examples per class.

4.2 Hyperparameters

We commence by segmenting sentences through CoreNLP[3] and preprocessing the dataset. The source text's maximum sentence length is set to

[2] https://www.yelp.com/dataset
[3] https://github.com/topics/corenlp

Dataset	Label Type	Classes	Labeled	Unlabeled	Dev	Test
CNN/DailyMail	Extractive Sentences	2	10/100/1,000	287,227 -10/-100/-1,000	13,368	11,490
AG News	News Topic	4	$\times 10/\times 30/\times 200$	20,000	8,000	7,600
Yahoo!Answer	Q&A Topic	10	$\times 10/\times 30/\times 200$	50,000	20,000	59,727
DBpedia	Wikipedia Topic	14	$\times 10/\times 30/\times 200$	70,000	28,000	70,000
Amzn Review	Product Review Topic	5	$\times 50/\times 200$	249,000	25,000	65,000
Yelp Review	Business Review Topic	5	$\times 50/\times 200$	249,000	25,000	50,000

Table 1: Dataset statistics and dataset split for semi-supervised text classification and semi-supervised extractive summarization tasks, in which '×' means the number of data per class. '−' means to subtract the quantity of data.

512 for extractive summarization and 256 for text classification. In the summarization task, we opt for the top 3 sentences from CNN/DailyMail based on the average length of the ORACLE human-written summaries. Fine-tuning on two tasks utilize the Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$. The supervised learning rates are configured within the intervals [1e-4, 3e-4] and [2e-5, 5e-5] for students and teachers, respectively. The distillation learning and mutual learning rates both range from [5e-4, 7e-4], with the balancing hyperparameter λ set to 0.1. The total number of training steps is 50,000. The warm-up period t for unsupervised learning is 5,000 steps. For the Curriculum Adversarial Noise Function `ANF`, the curriculum step period λ_k is 10,000, the curriculum step factor γ is 1, the variance σ^2 for initialization is 1e-5, the noise boundary ϵ is 1e-6, and the adversarial gradient ascent learning rate η is 1e-3. The supervised batch size is configured as 4, and the unsupervised batch size is set to 16 for classification (32 for summarization) in the majority of our experiments. We optimize hyperparameters through grid search on the development set, selecting the configuration yielding the best validation performance within the initial 10,000 training steps. These optimized hyperparameters are then applied to the complete training process. Model evaluation is performed every 100 training steps for both summarization and classification tasks.

4.3 Evaluation Methodology

We evaluate summarization quality using ROUGE F1 (Lin and Hovy, 2003). We report the full-length F1-based ROUGE-1, ROUGE-2, and ROUGE-L (R-1, R-2 and R-L) on the CNN/DailyMail. These ROUGE scores are computed using `ROUGE-1.5.5.pl` script[4]. We report the accuracy (Acc) results for the text classification tasks.

4.4 Implementation Details

For the extractive summarization, we can formulate it as a sequence labelling task as (Liu and Lapata, 2019). The extractive goal is to predict a sequence of labels $s_1, ..., s_n$ ($s_i \in 0, 1$) for sentences in a document, where $s_i = 1$ represents the i-th sentence should be included in the summaries. For the text classification, we feed the hidden state corresponding to each instance's [CLS] token to a softmax classification layer. We used the BERT[5] for text tokenization. For the mapping function $g(m)$, in the case of two student models, PS-NET students (S^{AK} and S^{BK}) are distilled from the first and last K layers of the teacher model. For scenarios involving multiple student models, additional PS-NET students (S^{CK}, S^{DK}, S^{EK}, and S^{FK}) are distilled from the intermediate K layers of the teacher model.

Each setting runs 3 different random seeds and computes the average performance. The standard deviation of all experimental results falls within the range of [0.3-0.7], which is not displayed to align with the baselines. All our experiments are conducted on a single NVIDIA Tesla V100 32GB GPU with PyTorch. Notably, our framework generates *two or more* students, from which *one* is selected as the inference model. To provide a clearer illustration, we present the performance of each individual student model. In practical applications, the selection of a single student model for inference relies on validation set results, ensuring computational efficiency. This practice of choosing a single model based on validation performance is standard in methodologies utilizing multiple models, such as mutual learning (Laine and Aila, 2017; Zhang et al., 2018; Qiao et al., 2018; Ke et al., 2019).

4.5 Baseline Methods

For text classification, we compare with: (*i*) supervised baselines, $\text{BERT}_{\text{BASE}}$ and default Tiny-

[4] https://github.com/andersjo/pyrouge

[5] https://github.com/google-research/bert

Models	$\mathcal{P}$(M)	AG News			Yahoo!Answer			DBpedia			Avg
		10	30	200	10	30	200	10	30	200	
$\text{BERT}_{\text{BASE}}$	109.48	81.00	84.32	87.24	60.10	64.13	69.28	96.59	98.21	98.79	82.18
UDA	109.48	84.70	86.89	88.56	64.28	67.70	69.71	98.13	98.67	98.85	84.17
TinyBERT^6	66.96	71.45	82.46	87.59	52.84	60.59	68.71	96.89	98.16	98.65	79.70
$\text{UDA}_{\text{TinyBERT}^6}$	66.96	73.90	85.16	87.54	57.14	62.86	67.93	97.41	97.87	98.26	81.79
DisCo (S^{A6})	66.96	77.45	86.93	88.82	59.10	66.58	69.75	98.57	98.61	98.73	82.73
DisCo (S^{B6})	66.96	74.38	86.39	88.70	57.62	64.04	69.57	98.50	98.45	98.57	82.02
TinyBERT^4	14.35	69.67	78.35	85.12	42.66	53.63	61.89	89.65	96.88	97.58	75.05
$\text{UDA}_{\text{TinyBERT}^4}$	14.35	69.60	77.56	83.60	40.69	55.43	63.34	88.50	93.63	95.98	74.26
DisCo (S^{A4})	14.35	77.36	85.55	87.95	51.31	62.93	68.24	94.79	98.14	98.63	80.54
DisCo (S^{B4})	14.35	76.90	85.39	87.82	51.48	62.36	68.10	94.02	98.13	98.56	80.31
PS-NET (S^{A4})	14.35	**81.03**	**87.32**	**89.04**	**62.33**	**68.10**	**71.35**	**97.19**	**98.70**	**98.90**	**83.77**
PS-NET (S^{B4})	14.35	**82.06**	**87.38**	**89.77**	**65.21**	**68.02**	**71.08**	**98.44**	**98.71**	**98.82**	**84.39**
FLiText	9.60	67.14	77.12	82.12	48.30	57.01	63.09	89.26	94.04	97.01	75.01
DisCo (S^{A2})	8.90	75.05	82.16	86.38	51.05	58.83	65.63	89.55	96.14	97.70	78.05
DisCo (S^{B2})	8.90	70.61	81.87	86.08	48.41	57.84	64.04	89.67	96.06	97.58	76.90
PS-NET (S^{A2})	8.90	**81.14**	**85.35**	**87.10**	**61.12**	**64.40**	**66.33**	**96.61**	**98.24**	**98.33**	**82.07**
PS-NET (S^{B2})	8.90	**81.89**	**87.69**	**89.11**	**64.16**	**66.88**	**69.57**	**98.05**	**98.77**	**98.57**	**83.85**

Table 2: Test accuracy (Acc (%)) for semi-supervised text classification tasks and the baseline results are derived from DisCo. $\mathcal{P}$(M) is the number of model parameters in millions. The $\text{BERT}_{\text{BASE}}$ and TinyBERT are supervised frameworks, UDA and $\text{UDA}_{\text{TinyBERT}}$, DisCo, and FLiText are semi-supervised frameworks using the same amount of unlabeled data as used by our PS-NET. The best results are in-bold, and the best average results are in-blue.

BERT (Jiao et al., 2020), (*ii*) semi-supervised UDA (Xie et al., 2020), and we introduce two noteworthy lightweight semi-supervised baseline models: FLiText (Liu et al., 2021) and DisCo (Jiang et al., 2023). FLiText is a lightweight and fast semi-supervised learning framework and consists of a two-stage training process where it initially trains a large inspirer model (BERT) and then optimizes a target network (TextCNN). DisCo is the state-of-the-art faster and lighter SSL framework which employs a co-training technique to optimize multiple small student models, promoting knowledge sharing among students through diverse data and model views. We also compare with other prominent SSL text classification methods and report their results on the Unified SSL Benchmark (USB) (Wang et al., 2022a). Most of these SSL methods work well on CV tasks, and Wang et al. (2022a) generalize them to NLP tasks by integrating a 12-layer BERT.

For extractive summarization, we use the open-source releases: (*i*) supervised baseline, BERTSUM (Liu and Lapata, 2019), (*ii*) unsupervised techniques, LEAD-3, TextRank (Mihalcea and Tarau, 2004) and LexRank (Erkan and Radev, 2004) and (*iii*) two state-of-the-art semi-supervised extractive summarization methods, UDASUM and CPSUM (Wang et al., 2022b) for comparison.

Table 3: ROUGE F1 performance of the extractive summarization. L_d=100 refers to the labelled samples. SSL baselines (CPSUM, UDASUM, DisCo) use the same unlabeled data as our PS-NET.

Models	$\mathcal{P}$(M)	L_d	CNN/DailyMail		
			R-1	R-2	R-L
ORACLE		100	*48.35*	*26.28*	*44.61*
LEAD-3		100	40.04	17.21	36.14
TextRank		100	33.84	13.11	23.98
LexRank		100	34.63	12.72	21.25
BERTSUM	109.48	100	38.58	15.97	34.79
CPSUM	109.48	100	38.10	15.90	34.39
UDASUM	109.48	100	38.58	15.87	34.78
TinyBERTSUM^4	14.35	100	39.83	17.24	35.98
$\text{UDASUM}_{\text{TinyBERT}^4}$	14.35	100	40.11	17.43	36.23
DisCo (S^{A4})	14.35	100	40.39	17.55	36.47
DisCo (S^{B4})	14.35	100	40.40	17.57	36.48
PS-NET (S^{A4})	14.35	100	**41.29**	**17.78**	**37.11**
PS-NET (S^{B4})	14.35	100	**40.74**	**17.86**	**37.16**

5 Experimental Results

5.1 Evaluation on Text Classification

Upon comparing the 2-layer students of PS-NET with the 12-layer BERT, our method demonstrates a notable performance enhancement across various text classification tasks, despite a 12.3× re-

Table 4: Test accuracy (Acc (%)) of other prominent SSL text classification models in Amazon Review and Yelp Review datasets. Results of baselines are sourced from the most up-to-date results on GitHub of USB Benchmark (Wang et al., 2022a). Results underlined indicate performance inferior to PS-NET. Svp refers to supervised methods.

Models	$\mathcal{P}$(M)	Amzn-50	Amzn-200	Yelp-50	Yelp-200
Fully-Svp	109.48	63.60	63.60	67.96	67.96
Svp	109.48	48.69	52.47	49.78	53.29
P-Labeling	109.48	46.55	52.34	49.40	52.79
∏-model	109.48	22.78	46.83	24.27	40.18
MeanTch	109.48	47.86	52.34	49.40	52.79
VAT	109.48	50.17	53.46	47.03	54.70
UDA	109.48	39.24	31.62	30.67	33.05
FixMatch	109.48	52.39	56.95	53.48	59.35
Flexmatch	109.48	54.27	57.75	56.65	59.49
AdaMatch	109.48	53.28	57.73	54.60	59.84
SimMatch	109.48	54.09	57.79	53.88	59.74
FreeMatch	109.48	53.59	57.36	52.05	59.63
SoftMatch	109.48	54.71	57.79	55.91	60.24
DisCo (S^{A4})	14.35	46.28	48.64	45.42	50.87
DisCo (S^{B4})	14.35	36.51	46.41	38.47	49.25
PS-NET (S^{A4})	14.35	46.22	53.23	49.68	55.32
PS-NET (S^{B4})	14.35	**47.56**	**54.77**	**50.12**	**57.65**

Table 5: Validation accuracy (Acc (%)) of PS-NET with multiple student peers. The labelled data consists of 10 samples per class. The students (S^{A2}, S^{B2}, S^{C2}, S^{D2}, S^{E2}, S^{F2}) are distilled from layers $\{1, 2\}$, $\{11, 12\}$, $\{3, 4\}$, $\{5, 6\}$, $\{7, 8\}$, $\{9, 10\}$ of the teacher $\text{BERT}_{\text{BASE}}$. The students ($S^{A6}$, S^{B6}) are distilled from layers $\{1, 2, 3, 4, 5, 6\}$, $\{7, 8, 9, 10, 11, 12\}$ of $\text{BERT}_{\text{BASE}}$.

Models	AG News	Yahoo!Answer	DBpedia
PS-NET (S^{A2})	81.14	61.12	96.61
PS-NET (S^{B2})	81.89	63.91	98.05
PS-NET (S^{A2})	81.20	60.22	95.70
PS-NET (S^{B2})	**83.50**	64.14	98.43
PS-NET (S^{C2})	81.33	61.05	96.43
PS-NET (S^{D2})	81.45	61.11	96.87
PS-NET (S^{E2})	82.41	61.28	97.55
PS-NET (S^{F2})	81.34	62.19	98.21
PS-NET (S^{A6})	82.39	64.12	98.51
PS-NET (S^{B6})	82.61	**64.33**	**98.49**

duction in model size. The 2-layer students of PS-NET notably surpass the 4-layer and 6-layer semi-supervised $\text{UDA}_{\text{TinyBERT}}$ and DisCo by a margin in semi-supervised text classification. With minimal labelled data, our PS-NET (the inferior student) featuring a 4-layer distilled BERT, outperforms the 4-layer $\text{UDA}_{\text{TinyBERT}}$ by an average margin of 9.51% across three datasets. Notably, PS-NET demonstrates robust performance, even with a minimal annotated data size of 10 per class. The superior student in PS-NET, equipped with a 2-layer distilled BERT, exhibits a substantial average performance improvement of 8.84% over FLiText and outperforms the best student of 2-layer DisCo by 5.80% across three datasets.

Table 4 provides a comparative analysis of PS-NET with other notable SSL methods equipped with a 12-layer BERT, utilizing results from the Unified SSL Benchmark (USB) (Wang et al., 2022a). Remarkably, PS-NET's 4-layer BERT-based students outperform most of these methods. These findings underscore the superiority of our model in scenarios involving lightweight model architecture and limited labelled data across various text classification tasks.

5.2 Evaluation on Extractive Summarization

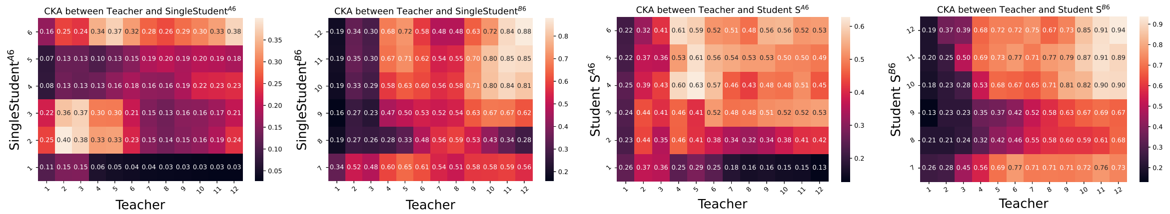
Results of the low-resource performance of the extractive summarization on CNN/DailyMail are shown in Table 3. Our approach is obviously superior to all supervised and SSL baselines, with only 100 labelled samples available.

In the summarization task, marginal performance differences exist among the SSL models. These discrepancies can be attributed partially to the inherent difficulty of 2-class sentence classification and the collapse problem (Yan et al., 2021; Chen and He, 2021; Gao et al., 2021) associated with BERT sentence representation. These difficulties are worsened by the constraint of having only 100 labelled summaries for training extractive models. Despite these factors, our method outperforms all existing SSL models in extractive summarization tasks, indicating its suitability for scenarios characterized by severe data scarcity issues.

5.3 Qualitative Analysis

In this section, we investigate the impacts of DML & CAT, and the scaling of students and layers within our PS-NET framework. Model efficiency is detailed in Appendix A.2.

Benefits of Mutual-learning (Peer Collaboration). To compare the advantages of mutual learning within our framework, we establish a SingleStudent setup with one student, employing the distillation loss as a regularization term alongside the supervised loss. It is crucial to note that the Sin-



(a) SingleStudent^{A6} (b) SingleStudent^{B6} (c) PS-NET only w/ DML (S^{A6}) (d) PS-NET only w/ DML (S^{B6})

Figure 2: The visualization of the Center Kernel Alignment (CKA (Zhu and Wang, 2021)) scores of PS-NET in Subfigures (c) and (d), along with its ablation variant, SingleStudent, shown in Subfigures (a) and (b). All models utilize a 6-layer BERT on the AG News dataset for evaluation, with 10 labelled examples per class.

Table 6: Validation accuracy (Acc (%)) comparison between PS-NET's students and the SingleStudent model under limited 10 labelled data per class. The SingleStudent undergoes a two-stage learning process with only one teacher and one student.

Models	AG News	Yahoo!Answer	DBpedia
SingleStudent^{A2}	80.29	61.46	96.13
SingleStudent^{B2}	80.66	63.10	97.48
PS-NET (S^{A2})	**81.14**	61.12	96.61
PS-NET (S^{B2})	**81.89**	**63.91**	**98.05**

gleStudent model differs from TinyBERT[6].

As shown in Table 6, under the two-stage framework of supervised knowledge optimization and unsupervised model compression, the performance of the SingleStudent is notably weaker compared to that of two students engaged in mutual learning. Figure 3 further supports this, showing that mutual learning improves the Center Kernel Alignment (CKA) score between students and the teacher, effectively narrowing the performance gap. This improvement is due to mutual learning in PS-NET, where students indirectly extract regularization from each other's predictive capabilities, thereby enhancing their individual generalization abilities.

Benefits of Curriculum Adversarial Learning (Self Transcendence). The ablation experimental results from Table 7 underscore the substantial performance boost of PS-NET when Curriculum Adversarial Training (CAT) is integrated into its training regimen. Within the FS-NET framework, guided by the CAT, one student consistently emerges in optimal performance. This improvement indicates that CAT effectively enhances the model's generalization capability.

[6]First, the SingleStudent uses online distillation with both labelled and unlabeled data in our two-stage framework, while TinyBERT begins with unlabeled data for general distillation and fine-tunes with labelled data. Second, TinyBERT separates labelled and unlabeled training, where the teacher optimizes labels while distilling knowledge.

Table 7: Validation accuracy (Acc (%)) of PS-NET using the minimum labelled data, comparing with and without curriculum adversarial training (CAT).

Models	AG News	Yaho..	Amzn-50	DBpedia
FS-NET (S^{A2}).w/o CAT	80.37	62.29	44.15	97.65
FS-NET (S^{B2}).w/o CAT	78.29	61.16	44.74	97.50
FS-NET (S^{A2}).w CAT	**81.14**	61.12	43.18	96.61
FS-NET (S^{B2}).w CAT	**81.89**	**63.91**	**45.31**	**98.05**

Effect of More Student Peers. The prior experiments study the dual-student cohort. We next investigate how PS-NET scales with more students in the cohort. In PS-NET, each student learns from all other students individually, regardless of how many students are in the cohort. As shown in Table 5, expanding to a four-student cohort in PS-NET enhances individual student performance, showcasing improved generalization as peer numbers increase. These results demonstrate that more student perturbations complement each other and are important to obtain superior performance for PS-NET with a compressed model and few labelled data.

Effect of More Student Layers. As shown in Table 5, two students with multiple layers exhibited performance advantages on the three datasets. It indicates that more student feature encoding diversifies the cohort and then encourages the individual models to teach each other in a complementary manner underlying multiple views to improve the learning performances. In practical scenarios, multiple lighter-weight students remain the preferred option, as smaller individual learning parameters result in faster inference speeds.

6 Conclusion and Future Work

We present a novel framework, PS-NET, designed to address both label scarcity and model size reduction in pre-trained language models (PLMs) such as BERT. PS-NET incorporates lightweight student cohorts to facilitate mutual learning and adversarial training, thereby enhancing generalization. In the future, we aim to extend PS-NET to other NLP benchmarks, including language understanding, machine reading comprehension, and text generation tasks. Additionally, leveraging other language models, such as RoBERTa (Liu et al., 2019) and GPT (Radford et al., 2018, 2019; Brown et al., 2020), will further contribute to our work.

7 Broader Impacts & Limitations

PS-NET furnishes robust technical solutions for faster and lighter semi-supervised learning, providing an effective way to deploy it on resource-limited devices and industrial applications.

A substantial distinction of PS-NET exists between the compression challenge encountered in large generative models, such as ChatGPT, and the standard knowledge distillation paradigm explored in PS-NET. Notably, Zhu et al. (2023) categorize standard knowledge distillation as falling within the White-box KD category, where the teacher's parameters are available for use. Conversely, the compressed Black-box KD applied to ChatGPT requires students to grasp both the teachers' knowledge and their emergent abilities (Wei et al., 2022). These abilities should be distilled by the teacher and be imitated by the students, including In-Context Learning (ICL) (Dong et al., 2023; Wang et al., 2023b), Chain of Thought (CoT) (Shi et al., 2023), and Instruction Following (IF) (Ouyang et al., 2022; Brooks et al., 2023). While various technological paradigms and explicit workloads prompt discussions on generative large language models for future research, BERT proves more applicable and sufficient for the discriminative tasks addressed in our works.

Besides, Few-shot LLMs emphasize task-agnostic pretraining LLM and its transferability with limited samples, while we focus on small, randomly initialized models tailored for limited labeled data. Recent few-shot methods like ICL (Dong et al., 2023; Wang et al., 2023b), PEFT (Zheng et al., 2024), and prompt-fre SetFit achieve strong results with minimal labeled data and without relying on unlabeled data, yet rely heavily on the pretrained performance of large models. Applying such methods to our PS-NET, which relies on arbitrarily small, randomly initialized models, poses significant challenges and expected workload.

Furthermore, distillation with LLMs is a meaningful aspect of evaluating model PS-NET scalability. However, LLMs encounter challenges in serving as teachers to train significantly smaller student models. Previous studies have discussed the inferior performance of knowledge distillation when there is a significant size disparity between the teacher and student models (Cho and Hariharan, 2019; Mirzadeh et al., 2020; Li et al., 2022; Huang et al., 2022). Existing research on LLM-based knowledge distillation primarily focuses on extracting and transferring the rich, nuanced understanding developed by these models rather than merely reducing the model size. Additionally, while ensembling models (Chen and Guestrin, 2016; Ke et al., 2017) boosts performance (as demonstrated by Jiang et al. (2023)) but contradicts our goal of a singular model for faster inference. We focus on training a lightweight model with limited labels for efficient inference.

Therefore, several promising avenues for further exploration exist within the PS-NET framework, with numerous optimization opportunities.

8 Ethical Statement

Data Availability and Safety. The experimental data analyzed in this paper are primarily publicly accessible; otherwise, we will provide links upon request for access.

Usage of Large PLM. The paper does not engage in additional utilization of the GPT-3.5 model or its derivatives, such as generating content for manuscripts or coding for models.

Authors Conflicts. The authors assert that they have no conflicts of interest. Informed consent has been obtained from all individual participants involved in the study. There are no financial or non-financial relationships or interests that could be perceived as potentially influencing the publication.

References

David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. 2019. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, pages 5050–5060.

David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alexey Kurakin. 2022. Adamatch: A unified approach to semi-supervised learning and domain adaptation. In *ICLR*. OpenReview.net.

Raymond A. Board and Leonard Pitt. 1989. Semi-supervised learning. *Mach. Learn.*, 4:41–65.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, pages 18392–18402. IEEE.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *NeurIPS*.

Ming-Wei Chang, Lev-Arie Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: Dataless classification. In *AAAI*, pages 830–835. AAAI Press.

Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. 2023. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. *CoRR*, abs/2301.10921.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. Mixtext: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *ACL*, pages 2147–2157. Association for Computational Linguistics.

Junfan Chen, Richong Zhang, Junchi Chen, and Chunming Hu. 2024. Open-set semi-supervised text classification via adversarial disagreement maximization. In *ACL*, pages 2170–2180. Association for Computational Linguistics.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *SIGKDD*, pages 785–794. ACM.

Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *CVPR*, pages 15750–15758. Computer Vision Foundation / IEEE.

Jang Hyun Cho and Bharath Hariharan. 2019. On the efficacy of knowledge distillation. In *ICCV*, pages 4793–4801. IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186. Association for Computational Linguistics.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey for in-context learning. *CoRR*, abs/2301.00234.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.*, 22:457–479.

Yue Fan, Anna Kukleva, Dengxin Dai, and Bernt Schiele. 2023. Revisiting consistency regularization for semi-supervised learning. *Int. J. Comput. Vis.*, 131(3):626–643.

Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born-again neural networks. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 1602–1611. PMLR.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *EMNLP*, pages 6894–6910. Association for Computational Linguistics.

Yuxin Guo, Shijie Ma, Hu Su, Zhiqing Wang, Yuhao Zhao, Wei Zou, Siyang Sun, and Yun Zheng. 2023. Dual mean-teacher: An unbiased semi-supervised framework for audio-visual source localization. In *NeurIPS*.

Jianfeng He, Hang Su, Jason Cai, Igor Shalyminov, Hwanjun Song, and Saab Mansour. 2024. Semi-supervised dialogue abstractive summarization via high-quality pseudolabel selection. In *NAACL*, pages 5976–5996. Association for Computational Linguistics.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *NeurIPS*, pages 1693–1701.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. 2022. Knowledge distillation from A stronger teacher. In *NeurIPS*.

Weifeng Jiang, Qianren Mao, Chenghua Lin, Jianxin Li, Ting Deng, Weiyi Yang, and Zheng Wang. 2023. Disco: Distilled student models co-training for semi-supervised text mining. In *EMNLP*, pages 4015–4030. Association for Computational Linguistics.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling BERT for natural language understanding. In *EMNLP*, volume EMNLP 2020 of *Findings of ACL*, pages 4163–4174. Association for Computational Linguistics.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. Lightgbm: A highly efficient gradient boosting decision tree. In *NeuralPS*, pages 3146–3154.

Zhanghan Ke, Daoye Wang, Qiong Yan, Jimmy S. J. Ren, and Rynson W. H. Lau. 2019. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *ICCV*, pages 6727–6735. IEEE.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. 2017. Adversarial examples in the physical world. In *ICLR*. OpenReview.net.

Samuli Laine and Timo Aila. 2017. Temporal ensembling for semi-supervised learning. In *ICLR*. OpenReview.net.

Dong-Hyun Lee et al. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML*, volume 3, page 896.

Haejun Lee, Drew A. Hudson, Kangwook Lee, and Christopher D. Manning. 2020. SLM: learning a discourse language representation with sentence unshuffling. In *EMNLP*, pages 1551–1562. Association for Computational Linguistics.

Xin-Chun Li, Wen-Shu Fan, Shaoming Song, Yinchuan Li, Bingshuai Li, Yunfeng Shao, and De-Chuan Zhan. 2022. Asymmetric temperature scaling makes larger networks teach well again. In *NeurIPS*.

Chin-Yew Lin and Eduard H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *HLT-NAACL*.

Chen Liu, Mengchao Zhang, Zhibing Fu, Panpan Hou, and Yu Li. 2021. Flitext: A faster and lighter semi-supervised text classification with convolution networks. In *EMNLP*, pages 2481–2491. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *EMNLP-IJCNLP*, pages 3728–3738. Association for Computational Linguistics.

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

Julian J. McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *RecSys*, pages 165–172. ACM.

Pablo N Mendes, Max Jakob, and Christian Bizer. 2012. *DBpedia: A multilingual cross-domain knowledge base*. European Language Resources Association (ELRA).

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *EMNLP*, pages 404–411. Association for Computational Linguistics.

Seyed-Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. 2020. Improved knowledge distillation via teacher assistant. In *AAAI*, pages 5191–5198. AAAI Press.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2019. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(8):1979–1993.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

Wonpyo Park, Wonjae Kim, Kihyun You, and Minsu Cho. 2020. Diversified mutual learning for deep metric learning. In *ECCV*, volume 12535 of *Lecture Notes in Computer Science*, pages 709–725. Springer.

Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. In *ICLR*. OpenReview.net.

Siyuan Qiao, Wei Shen, Zhishuai Zhang, Bo Wang, and Alan L. Yuille. 2018. Deep co-training for semi-supervised image recognition. In *ECCV*, volume 11219 of *Lecture Notes in Computer Science*, pages 142–159. Springer.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training. *OpenAI blog*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. 2015. Semi-supervised learning with ladder networks. In *NeuralPS*, pages 3546–3554.

Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. 2016. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *NeurIPS*, pages 1163–1171.

Dinghan Shen, Mingzhi Zheng, Yelong Shen, Yanru Qu, and Weizhu Chen. 2020. A simple but tough-to-beat data augmentation approach for natural language understanding and generation. *CoRR*, abs/2009.13818.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2023. Language models are multilingual chain-of-thought reasoners. In *ICLR*. OpenReview.net.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *NeurIPS*.

Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *ICLR*. OpenReview.net.

Chenyang Wang, Junjun Jiang, Zhiwei Zhong, and Xianming Liu. 2023a. Spatial-frequency mutual learning for face super-resolution. In *CVPR*, pages 22356–22366. IEEE.

Xinyi Wang, Wanrong Zhu, and William Yang Wang. 2023b. Large language models are implicitly topic models: Explaining and finding good demonstrations for in-context learning. *CoRR*, abs/2301.11916.

Yidong Wang, Hao Chen, Yue Fan, Wang Sun, Ran Tao, Wenxin Hou, Renjie Wang, Linyi Yang, Zhi Zhou, Lan-Zhe Guo, Heli Qi, Zhen Wu, Yu-Feng Li, Satoshi Nakamura, Wei Ye, Marios Savvides, Bhiksha Raj, Takahiro Shinozaki, Bernt Schiele, Jindong Wang, Xing Xie, and Yue Zhang. 2022a. USB: A unified semi-supervised learning benchmark for classification. In *NeurIPS*.

Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, Bernt Schiele, and Xing Xie. 2023c. Freematch: Self-adaptive thresholding for semi-supervised learning. In *ICLR*. OpenReview.net.

Yiming Wang, Qianren Mao, Junnan Liu, Weifeng Jiang, Hongdong Zhu, and Jianxin Li. 2022b. Noise-injected consistency training and entropy-constrained pseudo labeling for semi-supervised extractive summarization. In *COLING*, pages 6447–6456. International Committee on Computational Linguistics.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *CoRR*, abs/2206.07682.

Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. In *NeurIPS*.

Hai-Ming Xu, Lingqiao Liu, and Ehsan Abbasnejad. 2022. Progressive class semantic matching for semi-supervised text classification. In *NAACL*, pages 3003–3013. Association for Computational Linguistics.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. Consert: A contrastive framework for self-supervised sentence representation transfer. In *ACL/IJCNLP*, pages 5065–5075. Association for Computational Linguistics.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. 2021. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In *NeurIPS*, pages 18408–18419.

Miao Zhang, Li Wang, David Campos, Wei Huang, Chenjuan Guo, and Bin Yang. 2022a. Weighted mutual learning with diversity-driven model compression. In *NeurIPS*.

Minjia Zhang, Uma-Naresh Niranjan, and Yuxiong He. 2022b. Adversarial data augmentation for task-specific knowledge distillation of pre-trained transformers. In *AAAI*, pages 11685–11693. AAAI Press.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NeurIPS*, pages 649–657.

Ying Zhang, Tao Xiang, Timothy M. Hospedales, and Huchuan Lu. 2018. Deep mutual learning. In *CVPR*, pages 4320–4328. Computer Vision Foundation / IEEE Computer Society.

Mingkai Zheng, Shan You, Lang Huang, Fei Wang, Chen Qian, and Chang Xu. 2022. Simmatch: Semi-supervised learning with similarity matching. In *CVPR*, pages 14451–14461. IEEE.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *CoRR*, abs/2403.13372.

Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. 2023. A survey on model compression for large language models. *CoRR*, abs/2308.07633.

Yichen Zhu and Yi Wang. 2021. Student customized knowledge distillation: Bridging the gap between student and teacher. In *ICCV*, pages 5037–5046. IEEE.

A Appendix

A.1 Baselines Details

For the text classification task, TinyBERT (Jiao et al., 2020) is a compressed model implemented by 6-layer or 4-layer $\text{BERT}_{\text{BASE}}$. For semi-supervised methods, we use the released code to train the UDA, which includes ready-made 12-layer $\text{BERT}_{\text{BASE}}$, 6-layer, or 4-layer TinyBERT. In extractive summarization, the ORACLE system serves as an upper bound in the domain of extractive summarization.

Other SSL algorithms integrated with BERT are implemented in a unified semi-supervised learning benchmark (USB) (Wang et al., 2022a), including Mean Teacher (Tarvainen and Valpola, 2017), VAT (Miyato et al., 2019), FixMatch (Sohn et al., 2020), AdaMatch (Berthelot et al., 2022), and SimMatch (Zheng et al., 2022). These methods boost model robustness by ensuring consistent predictions on perturbed unlabeled samples, or uses pseudo labels for training enhancement. PCM (Xu et al., 2022) is a complex multi-submodule combination SSL model with a 12-layer BERT backbone. Recently, FreeMatch (Wang et al., 2023c) dynamically adjusts the confidence threshold based on the model's learning status. SoftMatch (Chen et al., 2023) employs a unified sample weighting formulation for pseudo-labeling. The majority of models were originally introduced in the realm of computer vision, and we present their text classification outcomes in the USB benchmark evaluation. Fully-Svp merges unsupervised data labels with labeled data for BERT training, while Svp is solely on BERT's classification results from labeled data.

A.2 Model Efficiency Analysis

DisCo and our PS-NET employ student models, such as 4-layer or 2-layer BERT. In other words, during the model deployment phase, DisCo and PS-NET demonstrate identical inference speeds of 4-layer or 2-layer BERT. As shown in Table 8, compared with the teacher $\text{BERT}_{\text{BASE}}$, the 2-layer small models are 12.30× smaller and 7.52× inference speedup in the model efficiency. FLiText is slightly faster than the smaller model generated DisCo and PS-NET. This is because FLiText uses a convolutional network while our student models use BERT with multi-head self-attention. The lower computational complexity of convolutional networks. The 1D-CNN requires $O(k \times n \times d)$ operations[7] used by FLiText. In contrast, the multi-head self-attention mechanism of BERT requires $O(n^2 \times d + n \times d^2)$ operations. However, despite the FLiText-model having more parameters, it gives a worse performance compared to the smaller student model generated by DisCo and PS-NET in Table 2. Our PS-NET achieves optimal performance and maintains comparable inference speed.

Table 8: Model efficiency about inference speedup on a single NVIDIA Tesla V100 32GB GPU. $\mathcal{T}_{\text{TS}}$(ms) refers to the speedup of extractive summarization models trained with 100 labelled data. $\mathcal{T}_{\text{TC}}$(ms) illustrates the speedup of text classification models trained with AG News 200 labelled data per class.

Models	$\mathcal{T}_{\text{TS}}$(ms)	Models	$\mathcal{T}_{\text{TC}}$(ms)
BERTSUM	12.66	$\text{BERT}_{\text{BASE}}$	12.94
CPSUM	12.66	TinyBERT^4	2.86
TinyBERTSUM^4	2.64	$\text{UDA}_{\text{TinyBERT}^4}$	2.86
$\text{UDASUM}_{\text{TinyBERT}^4}$	2.64	FLiText	1.56
DisCo(S^{A2} or S^{B2})	1.72	DisCo(S^{A2} or S^{B2})	1.72
PS-NET (S^{A2} or S^{B2})	1.72	PS-NET (S^{A2} or S^{B2})	1.72

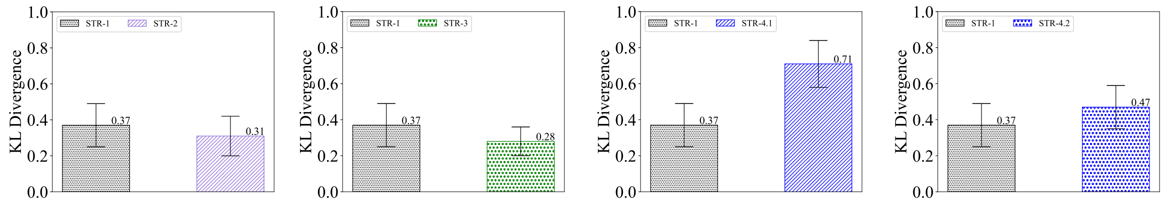
Table 9: Validation accuracy (Acc (%)) comparison between PS-NET, DisCo students and DML students (training from scratch) in SSL text classifications only using a limited 10 labelled data per class.

Models	AG News	Yahoo!Answer	DBpedia
DML(Net A2)	34.92	17.24	54.86
DML(Net B2)	34.82	17.46	61.28
DML(Net A4)	37.51	21.23	65.28
DML(Net B4)	36.76	21.62	64.80
DisCo (S^{A2})	70.47	48.10	89.78
DisCo (S^{B2})	74.30	50.95	89.80
PS-NET (S^{A2})	81.14	61.12	96.61
PS-NET (S^{B2})	**81.89**	**63.91**	**98.05**
DisCo (S^{A6})	74.69	57.42	98.06
DisCo (S^{B6})	77.40	58.48	98.03
PS-NET (S^{A6})	82.39	64.12	98.51
PS-NET (S^{B6})	**82.61**	**64.33**	**98.49**

A.3 Match Manner Analysis

We computed the match manner between the teacher network and student network using the average KL divergence ($\tau = 1$) on the predicted probabilities. Figure 3 (a) demonstrates that smaller representational capacity hinders student performance. A comparison between Figure 3 (a) and Figure 3 (b) reveals that increasing the number of

[7] n is the sequence length, d is the representation dimension, k is the kernel size of convolutions.



(a) STR-1 vs STR-2 (b) STR-1 vs STR-3 (c) STR-1 vs STR-4.1 (d) STR-1 vs STR-4.2

Figure 3: Match manner of KL divergence on PS-NET teacher and students. A smaller KL divergence value indicates less mismatch. Strategy 1 (STR-1) employs two 2-layer students in PS-NET, utilizing the first 2 layers and the last 2 layers of BERT teacher, respectively. Strategy 2 (STR-2) involves two 6-layer students in PS-NET. Strategy 3 (STR-3) utilizes six 2-layer students in PS-NET. Strategy 4.1 (STR-4.1) and Strategy 4.2 (STR-4.2) incorporate two SingleStudent models and BERT teacher, with two students corresponding to the first 2 layers (STR-4.1) and the last 2 layers (STR-4.2) of BERT teacher.

Table 10: Test accuracy (Acc (%)) of other prominent SSL models and our PS-NET. All results are reported by the Unified SSL Benchmark (USB) (Wang et al., 2022a). L_m is the number of the BERT layers.

$\mathcal{D}$	**Models**	$\mathbf{L}_m$	$\mathbf{L}_d$	Acc
AG News	∏-model (Rasmus et al., 2015)	12	50	86.56
	P-Labeling (Lee et al., 2013)		50	87.01
	MeanTeacher (Tarvainen and Valpola, 2017)		50	86.77
	PCM (Xu et al., 2022)		50	**88.85**
	MixText (Chen et al., 2020)		30	87.40
	DisCo	6	30	86.93
	PS-NET (ours)	2	30	**87.53**
Yahoo!Answer	P-Labeling (Lee et al., 2013)	12	200	66.56
	MeanTeacher (Tarvainen and Valpola, 2017)		200	66.57
	∏-model (Rasmus et al., 2015)		200	67.04
	VAT (Miyato et al., 2019)		200	68.47
	FlexMatch (Zhang et al., 2021)		200	68.58
	AdaMatch (Berthelot et al., 2022)		200	69.18
	FixMatch (Sohn et al., 2020)		200	69.24
	SimMatch (Zheng et al., 2022)		200	69.36
	CRMactch (Fan et al., 2023)		200	69.38
	SoftMactch (Chen et al., 2023)		200	69.56
	FreeMactch (Wang et al., 2023c)		200	69.68
	SoftMatch (Chen et al., 2023)		200	69.56
	DisCo	6	200	69.75
	PS-NET (ours)	4	200	**71.24**

students further diminishes the gap with the teacher network. In the context of a single student and a single teacher, shallow networks (such as STR-1) may amplify the discrepancy of predictions to the teacher and tend to be fairly severe.

A.4 Performance Superiority of PS-NET

Regarding the validation experiment with a limited 10 labelled data per class, as depicted in Table 9. PS-NET demonstrates superior performance compared to DisCo using the same training data. Besides, to further clarify the differences between our method and DML training from scratch, we compared the effectiveness of PS-NET and pure DML under an extreme SSL setting (4-layer and 2-layer BERT, 10 labelled data per class). It can be seen that in Table 9, with 10 labelled data per class, DML's performance barely exceeds that of a single model with random initialization and lags significantly behind PS-NET and other SSL baselines. The experimental results indicate that DML performs poorly when directly applied to SSL scenarios. In contrast, our method is specifically designed for SSL, emphasizing that a more powerful teacher network is essential to guide student models in scenarios with sparsely labelled data.

Besides, Table 10 provides more baseline SSL methods from the Unified SSL Benchmark (USB) (Wang et al., 2022a) in AG News, Yahoo!Answer datasets. With 200 labelled examples per class, PS-NET with 4 BERT layers achieves 71.24% accuracy, surpassing methods like VAT, FlexMatch, AdaMatch, and FixMatch, which use 12 BERT layers. With only 2 BERT layers and a relatively low labelled data per class (30), PS-NET also outperforms models like 6 BERT layers of DisCo. All supplementary experiments confirm the superior performance of the lightweight models achieved by PS-NET.

Finally, we sum up PS-NET integrates knowledge distillation and knowledge optimization within a unified framework, facilitating direct optimization of transferred knowledge by supervised signals. Moreover, in PS-NET, DML with interaction behaviour and CAT with adversarial perturbations enable each network to acquire distinct knowledge, bolstering the generalization and robustness of lightweight models.

A.5 Framework Variations from DisCo

Both PS-NET and DisCo represent technical solutions for faster and lighter semi-supervised learning. Additionally, similarities with DisCo are apparent in the use of a semi-supervised learning framework, with multiple networks learning logits estimates collaboratively. Regarding semi-supervised optimization, the objectives of both models remain consistent. They aim to augment generalization ability by employing complementary learning from multiple peer networks to elevate the posterior entropy (Pereyra et al., 2017) of each student network, thereby fostering shared learning experiences.

The semi-supervised learning framework in DisCo originates from Deep Co-Training, DCO (Qiao et al., 2018) which emphasizes multi-view learning. However, PS-NET's framework is derived from Deep Mutual Learning, DML (Zhang et al., 2018) which focuses on knowledge distillation variants. More information is provided in the original paper by Qiao et al. (2018) and Zhang et al. (2018). The proposed PS-NET differs from DisCo in the following aspects:

- **Distillation Methods**: DisCo uses offline distillation, necessitating pre-training of potent teacher networks and distillation of multiple students in advance. In contrast, PS-NET employs online distillation, where supervised learning and unsupervised distillation for both the teacher and student occur simultaneously. This design facilitates a more seamless emulation of the teacher's behavior. Specifically, the general knowledge is acquired during the initial training phases, while task-specific knowledge is learned in subsequent stages. This consistent emulation allows the teacher to guide the optimization paths of all students, rather than relying solely on peer learning without external guidance. By emphasizing external guidance, our framework enhances mutual learning between two divergent students, ultimately strengthening their collaborative learning process.

- **Learning Procedures**: DisCo pre-distils multiple students and then conducts co-training, where labelled and unlabeled data are input together. In contrast, PS-NET performs supervised learning ① sequentially followed by unsupervised knowledge distillation. This approach encourages the teacher model to actively participate in every single optimization step, thereby mitigating the impact of the scale gap between the teacher and student models on distillation performance. Similar discussions can be found in methodologies such as TAKD (Mirzadeh et al., 2020) and BANs (Furlanello et al., 2018). Furthermore, PS-NET incorporates curriculum adversarial training (CAT) (shown in Algorithm 1) to ② progressively increase learning complexity. These procedures enables PS-NET to implement an iterative learning approach, facilitating continuous self-improvement of the lightweight model.