Calibrating Pseudo-Labeling with Class Distribution for Semi-supervised Text Classification

Weiyi Yang[1], **Richong Zhang**[1,2*], **Junfan Chen**[1,3], **Jiawei Sheng**[4]
[1]CCSE, School of Computer Science and Engineering, Beihang University, Beijing, China
[2]Zhongguancun Laboratory, Beijing, China
[3]School of Software, Beihang University, Beijing, China
[4]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China

Abstract

Semi-supervised text classification (SSTC) aims to train text classification models with few labeled data and massive unlabeled data. Existing studies develop effective pseudo-labeling methods, but they can struggle with unlabeled data that have imbalanced classes mismatched with the labeled data, making the pseudo-labeling biased towards majority classes, resulting in catastrophic error propagation. We believe it is crucial to explicitly estimate the overall class distribution, and use it to calibrate pseudo-labeling to constrain majority classes. To this end, we formulate the pseudo-labeling as an optimal transport (OT) problem, which transports the unlabeled sample distribution to the class distribution. With a memory bank, we dynamically collect both the high-confidence pseudo-labeled data and true labeled data, thus deriving reliable (pseudo-) labels for class distribution estimation. Empirical results on 3 commonly used benchmarks demonstrate that our model is effective and outperforms previous state-of-the-art methods.

1 Introduction

Semi-supervised text classification (SSTC), which aims to achieve text classification with a few labeled data and massive unlabeled data, has become an appealing technique. It can facilitate many real-world applications, where the acquisition of numerous labeled data is expensive (Lee et al., 2021; Wang et al., 2022b; Zou and Caragea, 2023; Zou et al., 2023). In the context of large language models, SSTC research still has important practical significance in the utilization of unlabeled data for better efficiency and generalization (Xiao et al., 2023; Mishra et al., 2023; Chen et al., 2024).

Among various SSTC approaches, *pseudo-labeling* method (Xie et al., 2020; Chen et al., 2020; Lee et al., 2021; Xiao et al., 2023; Yang et al., 2023) can be one of the most common and effective semi-supervised paradigms, which assigns pseudo-labels for unlabeled data and takes into training. In this sense, unlabeled data expand the data for text classification through pseudo-labeling. However, the large volume of initially unlabeled data makes it impossible to know in advance the true data distribution and to proactively handle potential imbalance issues. Consequently, the model can be biased towards the majority classes, making some minority class samples incorrectly pseudo-labeled as the majority classes (shown in Figure 1(a) and (b)) with error propagation. This increases the difficulty of SSTC and becomes a bottleneck problem for further improvements.

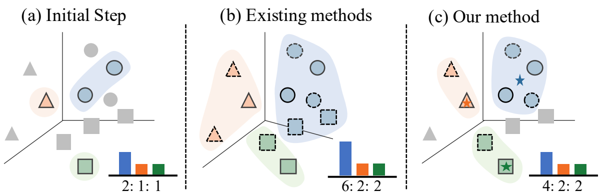


Figure 1: Illustration of the bias of imbalanced class data in SSTC. The symbols indicate classes of labeled (bordered), unlabeled data (unbordered) and pseudo-labeled data (dashed). ⋆ denotes the class prototypes. The bars in the legend indicate the class distribution.

To alleviate this bias, existing studies have attempted re-sampling or re-weighting methods. The re-sampling methods (Peng et al., 2025; Wei et al., 2021; Yang et al., 2023) select pseudo-labels with high probabilities to reduce majority class data, and the re-weighting methods (Guo and Li, 2022; Lai et al., 2022; Kim et al., 2020) diminish the weights of majority classes in training. However, they do not explicitly constrain the *class distribution* in pseudo-labeling, i.e., the relative sample frequencies from different classes. Without this key constraint, the class frequencies of pseudo-labels cannot be ensured, and the model can still assign exces-

[*] Corresponding author: zhangrc@act.buaa.edu.cn

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 13015–13028
November 4-9, 2025 ©2025 Association for Computational Linguistics

sive pseudo-labels of majority classes. To this end, we propose two key principles for reliable pseudo-labeling in SSTC: First, *the pseudo-labeled sample ought to have a high semantic similarity to its class*, ensuring high confidence in pseudo-labeling. Second, *the pseudo-labels ought to satisfy the class distribution of all data*, protecting class frequencies from excessive pseudo-labels of majority classes.

However, determining a reliable class distribution is still difficult. Intuitively, the class distribution can be determined by counting the class frequencies from labeled data (Kim et al., 2020). Unfortunately, since the class distribution of the labeled data depends on the annotators, e.g., annotated evenly, *the class distributions of the labeled data are hardly matched with the unlabeled data*. Therefore, directly estimating class distribution from labeled data may be unreliable. To this end, we further design a fixed-size memory bank (MB) to dynamically estimate the class distribution with unlabeled data, which is updated by the newly pseudo-labeled data of high confidence and the batched true labeled data. In this way, the estimation acquires more precise global class distribution, which can further be used to calibrate the pseudo-labeling to generate reliable pseudo-labels with limited majority classes.

Based on the successful estimation of the precise class distribution, we propose to constrain the pseudo-labeling with the class distribution. Particularly, we formulate pseudo-labeling as an *optimal transport (OT)* problem (Villani et al., 2009; Peyré et al., 2019). In general, the OT problem aims to derive a transport plan from a source distribution to a target distribution at the minimal total cost, based on the given coupling costs of all sample pairs. In the context of pseudo-labeling, assigning classes to unlabeled samples can be seen as a transport between their distributions, whose coupling costs can be achieved by the semantic similarity between the unlabeled samples and classes. Then, by solving OT, each unlabeled sample can be assigned with the most similar class as its pseudo-label, while ensuring the class distribution with a previously determined distribution (in Figure 1(c)).

Finally, we propose the overall Prototypical OT guided Pseudo-Labeling framework for SSTC, termed as PL-POT. We first design a text classifier as the backbone model. Inspired by previous studies (Cao et al., 2022), to treat each unlabeled sample equally, we specify a uniform distribution as the source sample distribution. The calibrated class distribution estimation serves as the target class distribution. To measure the coupling cost of OT, we obtain class prototypes using the samples in MB, and calculate the semantic similarity between the unlabeled samples and class prototypes. By solving the OT problem, the model achieves the aforementioned two principles, thus alleviating the critical bias of imbalanced class data. Our contributions can be summarized into three-folds:

- We propose a novel PL-POT framework for SSTC with OT for calibration. It alleviates the critical bias of imbalanced class data and is able to handle mismatched class distributions.
- To estimate the class distribution, we design an MB storing true labeled and reliable pseudo-labeled data, allowing the reliability with mismatched class distributions.
- Empirical studies indicate that our proposed model outperforms previous state-of-the-art methods on 3 commonly-used benchmarks, and comprehensive analysis confirms the effectiveness of PL-POT.

2 Related Work

2.1 Semi-supervised Text Classification

Semi-supervised text classification aims at utilizing the unlabeled data to improve the model performance. Existing SSTC methods can be divided into three groups: Generative methods (Gururangan et al., 2019; Croce et al., 2020), Graph-based methods (Li et al., 2021; Cui et al., 2022) and Pseudo-Labeling methods (Chen et al., 2024; Yang et al., 2023; Wang et al., 2021; Xiao et al., 2023; Tsai et al., 2022). Among existing SSTC frameworks, including LLM-based methods (Xiao et al., 2023; Zhang et al., 2023), pseudo-labeling methods have gained interest due to the remarkable performance. Most of these models rely on the principled assumption that the unlabeled and labeled data ought to have the same class distributions, which can struggle when facing balanced labeled data with imbalanced unlabeled data. Besides, SSTC with unlabeled data that have imbalanced classes mismatched with the labeled data can be a practical problem, yet remains under-researched.

2.2 Imbalanced Semi-supervised Text Classification

Existing supervised text classification studies on imbalanced classes propose re-sampling (Wei et al., 2021; Zhu et al., 2022; Freire et al., 2023) and re-weighting (Guo and Li, 2022; Lai et al., 2022) methods according to the class distribution. However, in semi-supervised text classification, the unlabeled data can be mismatched with the labeled data, making the class distribution unknown for existing methods. Existing imbalanced SSTC methods (Yang et al., 2023; Zou et al., 2023) employ conduct adaptive re-sampling to enforce balanced class distributions. Notably, PGPL (Yang et al., 2023) synergizes class prototypes and DeCrisisMB (Zou et al., 2023) implements a memory bank (MB) for balanced pseudo-label sampling during training iterations, these approaches risk prediction miscalibration when confronting mismatched class distributions between labeled and unlabeled data. Distinct from such balanced sampling paradigms, our PL-POT model achieves pseudo-label distribution alignment through global class distribution estimation across all data. Using the estimation, we formulate the pseudo-labeling as an optimal transport (OT) problem to ensure the predictions with an appropriate class distribution, addressing the imbalanced and mismatched class problem.

3 Preliminaries

3.1 Task Formulation

In SSTC, we are given a small set of labeled text data and a large set of unlabeled text data. Formally, let $\mathcal{C}$ be the class set of labels. Let $\mathcal{D}_l = \{\ (x_1^l, y_1^l), \cdots, (x_m^l, y_m^l)\ \}$ be the set of pre-annotated m labeled text samples, where each x_i^l and $y_i^l \in \mathcal{C}$ denote a text sample and its corresponding label, respectively. Let $\mathcal{D}_u = \{\ x_1^u, \cdots, x_n^u\ \}$ be the set of n unlabeled text data, where $x_i^u \in \mathcal{D}_u$ denotes an unlabeled text sample. We leverage the limited labeled data and the massive unlabeled data with unknown distribution for pseudo-labeling and include them as additional training data.

3.2 Optimal Transport

Optimal transport (OT) (Villani et al., 2009; Peyré et al., 2019) aims to find a solution for transferring mass from one distribution to another with minimal cost. Formally, suppose we are given two set of points $\mathcal{X} = \{x_1, x_2, ..., x_n\}$ and $\mathcal{Y} = \{y_1, y_2, ..., y_m\}$ with their empirical distributions as $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, respectively. We define $\boldsymbol{C}_{ij}$ as the cost between x_i and y_j. For simplicity, the OT problem can be denoted as $\mathrm{OT}(\boldsymbol{\mu}, \boldsymbol{\nu}, \boldsymbol{C}) := \min_{\boldsymbol{\Pi}} < \boldsymbol{C}, \boldsymbol{\Pi} >$, where $< \cdot, \cdot >$ denotes Frobenius inner product of matrices, and there are $\{\boldsymbol{\Pi} \in \mathbb{R}_+^{n\times m} | \boldsymbol{\Pi}\mathbf{1}_n = \boldsymbol{\mu}, \boldsymbol{\Pi}^\top\mathbf{1}_m = \boldsymbol{\nu}\}$. Note that the OT problem can be solved using the Sinkhorn-Knopp algorithm (Cuturi, 2013) efficiently, and we provide the details in **Appendix** A.

The OT solutions can be used in many studies (Ho et al., 2017; Courty et al., 2017; Ge et al., 2021; Yang et al., 2023). In particular, Chang et al. (2022); Tian et al. (2023); Li et al. (2024a) have achieved significant advancements in unsupervised representation learning through OT-driven matching of samples to prototypes. However, in SSTC, the use of distributional information within a unified OT framework to calibrate pseudo-labeling decision remains unexplored.

4 Method

The model structure of PL-POT is illustrated in Figure 2. The labeled and unlabeled text samples are first input into the backbone classifier. Then, we estimate the class distribution with MB to calibrate the unlabeled samples. We also measure the coupling cost using the similarity between unlabeled samples and class prototypes from MB. Finally, by formulating pseudo-labeling as a transport from unlabeled samples to classes, we ensure the pseudo-labeled samples having high similarity to their classes, and satisfying the calibrated class distribution. The pseudo-labeled data is used to augment the training set and retrain the model.

4.1 Backbone Text Classifier

To utilize the power of pre-trained language models in SSTC, we follow the previous practices (Chen et al., 2020) that use BERT (Devlin et al., 2019) as the text encoder. Specifically, let x be a labeled sample $x^l \in \mathcal{D}_l$ or an unlabeled sample $x^u \in \mathcal{D}_u$. The function $f(x;\theta)$ denotes the BERT encoding process with mean pooling operation over the token representations. Afterward, a two-layer MLP with $\tanh(\cdot)$ activation is adopted to derive the relevance score of x corresponding to each class:

$$g(x;\phi_c,\theta) = \mathrm{MLP}(f(x;\theta);\phi_c), \qquad (1)$$

where ϕ_c is the MLP parameter corresponding to class $c \in \mathcal{C}$. To train the backbone text classifier with labeled data, we adopt cross-entropy loss on batched labeled samples:

$$\mathcal{L}_s = -\frac{1}{|\mathcal{B}^l|}\sum_{x_i^l\in\mathcal{B}^l}\log\frac{\exp(g(x_i^l,\phi_{y_i^l},\theta))}{\sum\limits_{c\in\mathcal{C}}\exp(g(x_i^l,\phi_c,\theta))}, \qquad (2)$$

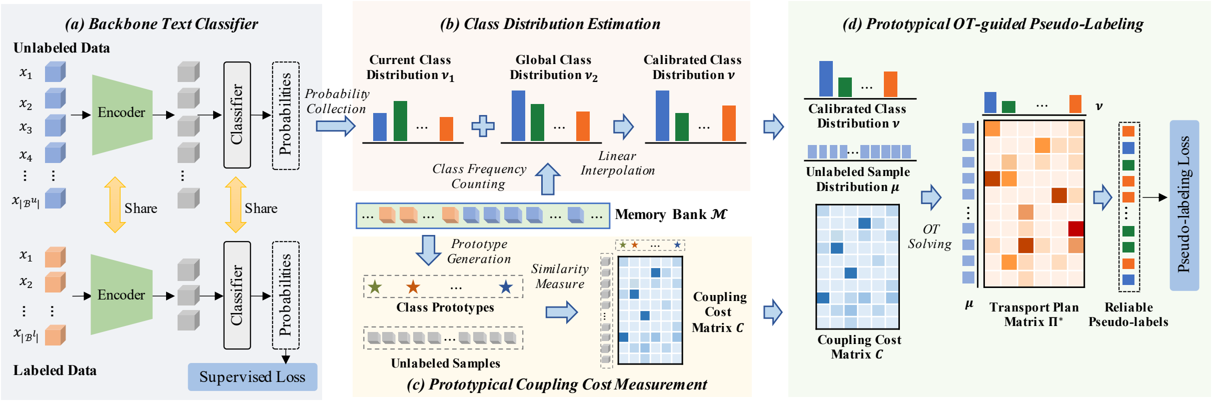


Figure 2: The overview of our proposed model, PL-POT.

To improve the stability in training, we adopt the warm-up strategy, training with labeled data in initial training iterations (w.r.t., labeled batches). The warmed classifier is further used to assign pseudo-labels for unlabeled samples, and then adopt these pseudo-labeled samples for training.

4.2 Prototypical OT-guided Pseudo-Labeling

In general, the classifier trained on the labeled data can be further enhanced by pseudo-labeled data. However, certain majority classes tend to have more and more pseudo-labels in the self-training with error propagation. In light of the aforementioned considerations, we formulate pseudo-labeling as an OT problem, for that OT takes into account the distribution of the data $\boldsymbol{\nu}$ when making predictions, so that it is not affected too much by bias towards majority classes. Specifically, suppose we have obtained the distribution $\boldsymbol{\mu}$ of the batched unlabeled samples $\mathcal{B}^u$ and distribution $\boldsymbol{\nu}$ of the pre-defined classes $\mathcal{C}$, and the coupling costs from samples to classes as $\boldsymbol{C} \in \mathbb{R}^{|\mathcal{B}^u| \times |\mathcal{C}|}$ (detailed later in Eq. (9)). The pseudo-labeling (PL) procedure can be formulated as:

$$\begin{aligned} \mathrm{PL}(x^u \in \mathcal{B}^u, \mathcal{C}; g) &:= \mathrm{OT}(\boldsymbol{\mu}, \boldsymbol{\nu}, \boldsymbol{C}) \\ &:= \min\nolimits_{\boldsymbol{\Pi}} < \boldsymbol{C}, \boldsymbol{\Pi} >, \\ \text{s.t. } \boldsymbol{\Pi}\mathbf{1}_{|\mathcal{B}^u|} = \boldsymbol{\mu},&\ \boldsymbol{\Pi}^\top\mathbf{1}_{|\mathcal{C}|} = \boldsymbol{\nu}. \end{aligned} \tag{3}$$

As widely utilized in previous studies (Cao et al., 2022; Zhang et al., 2024), $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ can be assumed as uniform distributions without prior knowledge, i.e., $\boldsymbol{\mu} := \{\mu_i | \mu_i = 1/|\mathcal{B}^u|\}_i^{|\mathcal{B}^u|}$ and $\boldsymbol{\nu} := \{\nu_i | \nu_i = 1/|\mathcal{C}|\}_i^{|\mathcal{C}|}$, respectively. However, this assumption treats all samples and classes equally, neglecting the fact that different classes may have different frequencies in practice. Therefore, we would like to estimate the practical class distribution and detail the coupling cost measurement in the following sections.

4.2.1 Class Distribution Estimation

To estimate the class distribution for unlabeled samples, we employ the classifier initially trained on the labeled data as the estimator. Specifically, for current unlabeled sample batch $\mathcal{B}^u$, the unlabeled sample $x \in \mathcal{B}^u$ are first input into the classifier:

$$p(c|x) = \frac{1}{Z}\exp(g(x, \phi_c, \theta)), \tag{4}$$

where the factor $Z := \sum_{c \in \mathcal{C}} \exp(g(x, \phi_c, \theta))$. Intuitively, we can decide the class of x with $\hat{y} = \arg\max_c p(c|x)$, and simply count the relative frequencies of the decided classes in $\mathcal{B}^u$ as current class distribution. Here, we collect the predicted probabilities of classes in $\mathcal{B}^u$ to estimate an enriched class distribution:

$$\boldsymbol{\nu}_1(c) = \frac{1}{|\mathcal{B}^u|}\sum_{x \in \mathcal{B}^u} p(c|x), c \in \mathcal{C}. \tag{5}$$

In this way, it enriches the underlying knowledge of classes, and avoid the potential incorrectness of the pseudo-labels.

However, predictions can still be biased towards majority classes and uncalibrated (Yang et al., 2023; Wang et al., 2022a). Therefore, we introduce a memory bank (MB) $\mathcal{M}$ to achieve global estimation of the class distribution with all reliable (pseudo-) labels. Specifically, the MB stores true labeled samples at the beginning, and then stores both batched true labeled and selected pseudo-labeled samples in recent training iterations:

$$\begin{aligned} \mathcal{M} = \{&\cdots, (x_1^l, y_1^l), \cdots, (x_{|\mathcal{B}^l|}^l, y_{|\mathcal{B}^l|}^l), \\ &(x_1^u, \hat{y}_1^u), \cdots, (x_{N_t}^u, \hat{y}_{N_t}^u), \cdots\}, \end{aligned} \tag{6}$$

where N_t is the selected reliable pseudo-labeled samples at t step. $\mathcal{M}$ is achieved by a queue with a fixed maximum length $|\mathcal{M}| \equiv M$. In this way, we combine the true labels of the labeled data and the reliable pseudo-labels of unlabeled data, allowing global estimation and alleviating potential mismatch between labeled and unlabeled data. Based upon, we count the relative frequencies of classes in $\mathcal{M}$ as the global class distribution:

$$\boldsymbol{\nu}_2(c) = \frac{1}{|\mathcal{M}|} \sum_{y_i \in \mathcal{M}} \mathbb{I}(y_i = c), c \in \mathcal{C}. \tag{7}$$

Finally, we can calibrate the current estimation of class distribution with the global estimation, which is used to alleviate the classifier bias of majority classes. Specifically, we derive the calibrated class distribution for the batched unlabeled samples $\mathcal{B}^u$ by linear interpolation as:

$$\boldsymbol{\nu} = \lambda \boldsymbol{\nu}_1 + (1 - \lambda) \boldsymbol{\nu}_2, \tag{8}$$

where λ is a correlation coefficient that controls the ratio of the current estimation to the global estimation. Note that $\boldsymbol{\nu}$ is dynamically updated with respect to the unlabeled batch $\mathcal{B}^u$ and the status of $\mathcal{M}$ during self-training iterations.

4.2.2 Prototypical Coupling Cost Measurement

To derive the coupling cost between unlabeled samples and classes, we devise a prototypical measurement. Instead of obtaining prototypes with unlabeled samples in $\mathcal{B}^u$ only, we acquire more reliable and informative samples of the classes in $\mathcal{M}$. For class $c \in \mathcal{C}$, the prototype $\boldsymbol{z}_c$ is computed by:

$$\boldsymbol{z}_c := \frac{1}{N_c} \sum_{x_i \in \mathcal{M}} f(x_i; \theta) \mathbb{I}(y_i = c), \tag{9}$$

where $N_c := \sum_{x_i \in \mathcal{M}} \mathbb{I}(y_i = c)$ is the total number of the samples with class c. Here, x_i includes both the labeled sample x^l and pseudo-labeled sample x^u in $\mathcal{M}$, which ensures semantic correctness by the true labeled data and comprises enriched semantics by the reliable pseudo-labeled data.

Afterward, we define a measurement function ψ to derive the coupling cost between $x^u \in \mathcal{B}^u$ and $c \in \mathcal{C}$. In general, it can be achieved by any semantic similarity measurement (with $\psi \in \mathbb{R}_+$ to meet the OT requirement), and here we adopt cosine similarity for its simplicity and effectiveness:

$$\psi(f(x^u), \boldsymbol{z}_c) := 1 - \frac{f(x^u; \theta)^\top \boldsymbol{z}_c}{\|f(x^u; \theta)\|_2 \|\boldsymbol{z}_c\|_2}. \tag{10}$$

Here, higher cosine similarity implies lower coupling cost. The obtained scores constitute entire coupling cost matrix $\boldsymbol{C} \in \mathbb{R}^{|\mathcal{B}^u| \times |\mathcal{C}|}$, which is used for pseudo-labeling in Eq. (3).

4.2.3 Pseudo-label Decision & Memory Bank Update

After obtaining the class distribution $\boldsymbol{\nu}$ and coupling cost $\boldsymbol{C}$, we solve the optimal transport plan $\boldsymbol{\Pi}^*$ by Sinkhorn-Knopp algorithm (Cuturi, 2013). Note that we adopt a uniform distribution to estimate unlabeled sample distribution $\boldsymbol{\mu}$ to treat all samples equally. Afterward, we decide the pseudo-labels by confirming the decision with $\boldsymbol{\Pi}^*$:

$$\mathbb{I}(x_i^u = \hat{y}_i^u) := \mathbb{I}(\arg\max_c \boldsymbol{\Pi}_{ic}^* = \hat{y}_i^u), \tag{11}$$

where $\hat{y}_i^u$ is the predicted class of the classifier. This decision further confirms the correctness of the pseudo-labels, improving the reliability. For each class, half of the pseudo-labeled data with higher probabilities in $\mathcal{B}^u$ and all labeled samples in $\mathcal{B}^l$ are inserted into the queue of $\mathcal{M}$, while the data that exceeds the length leaves the queue.

4.3 Overall Training Objective

To enrich the supervision with unlabeled data, we apply cross-entropy loss on the reliable pseudo-labeled samples:

$$\mathcal{L}_u = -\frac{1}{|\mathcal{B}^u|} \sum_{x_i^u \in \mathcal{B}^u} \delta(x_i^u) \log \frac{\exp(g(\tilde{x_i^u}, \phi_{y_i^u}, \theta))}{\sum_{c \in \mathcal{C}} \exp(g(\tilde{x_i^u}, \phi_c, \theta))},$$
$$\delta(x_i^u) = \begin{cases} 1, & \arg\max_k \boldsymbol{\Pi}_{ik}^* = \hat{y}_i^u \\ \alpha, & \arg\max_k \boldsymbol{\Pi}_{ik}^* \neq \hat{y}_i^u \end{cases}, \tag{12}$$

where we set weights δ for different samples. Inspired by Ishida et al. (2020), we let α be a negative scalar that indicates the mistrust degree of $\hat{y}_i^u$ in training. As previous SSTC studies (Yang et al., 2023; Chen et al., 2024), we also leverage the augmented sample $\tilde{x_i^u}$ of x_i^u with pseudo-labels for training. The overall objective $\mathcal{L}$ is combined with the two objectives during self-training iterations:

$$\mathcal{L} = \mathcal{L}_s + \beta \mathcal{L}_u, \tag{13}$$

where β is the harmonious factor to balance the two losses.

5 Experiment Setup

5.1 Datasets

Following the previous works in SSTC (Chen et al., 2020; Lee et al., 2021; Yang et al., 2023), we conduct intensive experiments on three widely used text classification benchmark datasets:

AGNews (Zhang et al., 2015) is extrated from the AG news corpus and is processed by compiling the titles and descriptions of articles from the 4 classes.

Yahoo (Chang et al., 2008) is a widely used question classification dataset with 10 classes, which is the question and answer pairs extracted from the Yahoo! Answers website.

DBPedia (Lehmann et al., 2015) is extracted from Wikipedia and is commonly used in query understanding. This dataset contains 14 classes.

Dataset	$\|\mathcal{C}\|$	Lab	Unlab	Dev	Test
AG News	4	10	20000	8000	7600
DBpedia	14	10	70000	28000	70000
Yahoo	10	10	50000	50000	60000

Table 1: Statistics of the datasets. $|\mathcal{C}|$ denotes the class numbers, 'Lab' denote labeled texts for each class, and 'Unlab', 'Dev' and 'Test' denote total samples for unlabeled, validation and test set, respectively.

5.2 Experimental Settings

We evaluate PL-POT on 3 commonly-used datasets. In the training set, the value assigned to each class of labeled data is 10. The largest class of unlabeled data is set to 5000, which is the number of unlabeled data points typically observed in a standard SSTC task. The remaining unlabeled data are randomly sampled based on the predefined imbalance ratios, denoted as τ_u. This ratio represents the number of samples of the most numerous class divided by the least numerous class. The number of texts in the other classes is calculated as a proportion of these two values. The hold-out test set remains untouched and balanced. The statistics of the datasets are shown in Table 1.

5.3 Implementation Details

We used PyTorch[1] for implementation. For all compared models, the maximum sentence length is set to 256. For bert, the initial learning rate is tuned in [$1e^{-6}$, $1e^{-5}$] for BERT parameters and [$1e^{-4}$, $1e^{-3}$] for other parameters. We set batch size of 16 for unlabeled data and 4 for labeled data. The threshold λ that controls the ratio of current estimation is turned in [0, 1]. The weight β to balance the loss is fixed to 1 and the loss weight α is turned in [-0.5, 0]. We turn the length of memory bank M in [16, 32, 64, 128]. For parameters of the OT part, the cost epsilon is turned in [0.1, 0.5] and the iteritions of sinkhorn algorithms are turned in [10, 100]. We run each experiment setting 3 times on the validation data and report the average performance and standard deviation. For all baseline models, we report our detailed hyper-parameter settings of all models in **Appendix** B.

5.4 Baselines

We evaluate the models on the three benchmarks and compare them with the following baselines:

BERT (Devlin et al., 2019) adopts a pre-trained model for text classification, trained with only labeled data.

UDA (Xie et al., 2020) makes soft predictions with original data to train the augmented data.

MixText (Chen et al., 2020) makes predictions by mixing the original data with the augmented data.

SALNet (Lee et al., 2021) establishes attention-driven lexicons that synergize pseudo-label refinement with semi-supervised co-training through dynamic lexicon-model interdependency.

PGPL (Yang et al., 2023) synergizes prototype-guided predictions with balanced pseudo-label resampling through iterative refinement cycles.

DeCrisisMB (Zou et al., 2023) uses a memory bank to store and equally sample generated pseudo-labels from each class at each training iteration.

SPPW (Chen et al., 2024) filters mislabeled texts with both label confidence and text hardness synergistically to replace the overfitting-prone learning.

To make fair comparisons with baselines, we use the released code in their original papers on our split data, with the same BERT encoder and data augmentations as MixText. We run each model 3 times and report the mean and standard deviation. Under balanced experimental settings, we use 3 different labeled seeds and run 3 experiments under each labeled seed. For SPPW, we report the reported results in the paper.

6 Overall Results

6.1 Imbalanced Model Performance Evaluation

In real-world scenarios, it is natural that unlabeled data have different distributions, i.e., $\tau_u \neq 1$. In Table 2, we compare the existing SOTA methods with the evaluation criterion (Accuracy/Macro-F1). The hold-out test set remains untouched and balanced.

[1] https://pytorch.org/

		$\tau_u = 5$		$\tau_u = 10$		$\tau_u = 20$	
Dataset	Method	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
AGNews	UDA	81.88±2.52	82.06±2.57	79.87±3.06	79.91±3.15	72.43±3.18	71.21±4.75
	MixText	85.64±0.96	84.33±0.96	84.13±1.27	83.43±1.34	82.93±1.57	82.75±2.39
	PGPL	85.53±0.84	85.62±0.60	84.03±0.78	83.97±0.92	82.16±1.02	81.58±1.36
	DeCrisisMB	85.11±0.75	84.66±0.73	83.03±1.04	82.97±0.54	81.95±1.15	80.73±1.19
	PL-POT	**86.98±0.68**	**85.71±0.91**	**84.17±1.29**	**84.20±1.26**	**84.18±1.93**	**84.08±2.75**
Yahoo	UDA	61.49±1.27	60.78±1.02	58.71±0.01	57.87±1.06	58.01±1.67	56.82±1.90
	Mixtext	65.16±0.66	63.63±0.47	64.25±0.41	62.94±0.25	64.03±0.53	62.30±0.78
	PGPL	64.64±0.35	64.67±0.43	62.78±0.88	62.69±0.76	58.22±0.99	58.23±1.02
	DeCrisisMB	64.25±0.43	64.04±0.58	62.65±0.63	62.52±0.61	59.63±0.97	59.04±1.10
	PL-POT	**66.11±0.81**	**65.34±0.75**	**64.47±0.43**	**63.29±0.26**	**64.34±1.02**	**63.60±1.27**
DBpedia	UDA	97.22±0.58	97.20±0.60	97.12±0.63	97.12±0.66	94.50±3.10	94.34±3.10
	MixText	97.92±0.76	96.49±1.03	96.83±1.24	96.64±1.35	95.34±1.03	94.68±1.44
	PGPL	98.01±0.19	98.07±0.27	97.54±0.40	97.02±0.39	94.65±0.89	94.13±0.68
	DeCrisisMB	98.14±0.22	97.82±0.43	96.00±0.54	95.63±0.55	94.23±0.74	94.16±0.78
	PL-POT	**98.29±0.77**	**98.28±0.52**	**97.82±0.46**	**97.82±0.46**	**97.77±0.68**	**97.73±0.74**

Table 2: Comparison of classification performance on all datasets under max/min = τ_u (hold-out test set is balanced). The evaluation criterion is Accuracy/Macro-F1. The best results are in bold.

	AGNews		Yahoo		DBpedia		Average	
Method	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
BERT	74.53 ±3.42	72.39±2.03	57.40 ±0.95	56.87±0.88	96.58±0.56	96.11±0.57	76.17	75.12
UDA	84.15±1.10	84.10±1.05	62.83±0.88	59.91±1.36	98.28±0.21	98.37±0.21	81.75	80.79
MixText	86.57±0.56	86.28±0.61	65.40±1.72	64.25±1.46	97.67±0.09	97.67±0.10	83.21	82.73
SALNet	77.61±3.17	77.61±3.17	52.43±0.27	52.30±0.16	95.39±0.17	95.39±0.15	75.14	75.10
PGPL	87.86±0.41	87.53±0.32	67.43±0.67	67.02±0.35	98.57±0.20	98.03±0.22	84.62	84.19
DeCrisisMB	86.75 ±0.53	86.65±0.62	65.23±1.20	64.15±1.27	98.43±0.12	98.29±0.13	83.47	83.03
SPPW	88.59 ±0.44	88.54±0.43	64.86±1.48	63.76±1.75	98.43±0.25	98.43±0.27	83.96	83.58
PL-POT	88.41±0.26	88.37±0.27	67.41±0.59	66.37±0.83	98.58±0.22	98.58±0.23	**84.80**	**84.44**

Table 3: The semi-supervised text classification results on three datasets under balanced experimental settings.

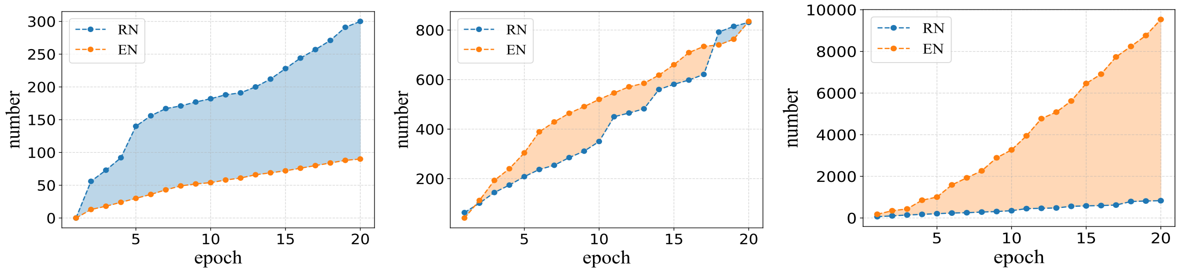
The superiority of PL-POT is evident when unlabeled data exhibit disparate imbalanced ratios relative to labeled data, particularly when unlabeled data also display significant imbalance. Similarly, PGPL, which employs the use of prototypes, demonstrates a proclivity towards convergence when 'τ_u' is relatively low. In the case of 'τ_u = 20', PL-POT outperforms the existing state-of-the-art by 1.33% in terms of F1 on AGNews and 3.39% on DBpedia. PL-POT outperforms DeCrisisMB even when both leverage memory banks to mitigate class imbalance, demonstrating its superiority in dynamically calibrating pseudo-label. This phenomenon can be attributed to the utilization of a pseudo-labeling framework, which is effective when calibrated with estimated class distributions. Consequently, PL-POT achieves superior results, even in the context of severe data imbalances.

6.2 Balanced Model Performance Evaluation

Considering that most existing methods are task settings for balanced datasets, for fairness of comparison we used balanced unlabeled data for all datasets, i.e. 5000 unlabeled data were sampled for each class and tested the validity of our model on the balanced dataset. The SSTC results of the compared models are shown in Table 3. It illustrates that PL-POT outperforms baseline models on Average results (Acc/F1). This observation manifests that our PL-POT model achieves improvements in SSTC, even the balance experiment settings.

6.3 Ablation Study

To systematically evaluate component contributions, we conduct an ablation study on AGNews via individually removing each module. As shown in Table 4 (averaged over three validation runs with τ_u=20), the full model achieves optimal validation performance, confirming that each module con-

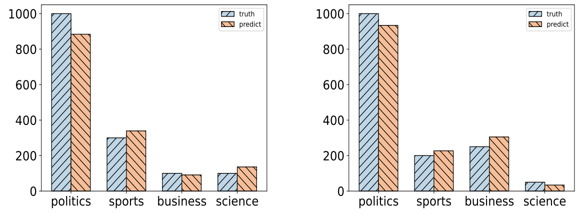


(a) PL-POT

(b) Similarity

(c) Balance OT

Figure 3: Calibration results via different measures on AGNews ($\tau_u = 20$).



(a) τ_u=10

(b) τ_u = 20

Figure 4: Calibrated class distribution on AGNews.

tributes positively to overall effectiveness. It can be observed that after removing the Distribution Estimation (DE) module, the Macro-F1 result is reduced by 23.14%. The reason for this is that when the DE module is removed, the prior distribution aligns with the distribution of labeled data, which is balanced. This is an extreme mismatch with the distribution of classes for unlabeled data, resulting in the forced assignment of many majority classes into majority classes in order to achieve balance. This confirms the effectiveness of our approach.

Datasets	Ablation	Acc	F1	ΔF1
AGNews	PL-POT	**86.36**	**85.79**	-
	w/o Warm	85.84	84.19	↓ 1.60
	w/o MB	85.14	84.10	↓ 1.69
	w/o DE	78.28	62.65	↓ 23.14
	w/o MD	84.03	84.09	↓ 1.70

Table 4: Ablation results (%) on AGNews, MD means the module which employs mistrust degree adjust in the training objective and MB means memory bank module.

6.4 Class Distribution Evaluation

In order to evaluate our results more accurately, we compare the predicted number with the truth distribution on all individual classes on AGNews with different ratios in Figure 4. It is observed that PL-POT with POT calibration improves the performance on all individual classes. The class distribution is close to the true distribution after calibration, especially for the minority classes, and there is no large predicted-true class mismatch due to the dominant class bias, illustrating the validity of our POT calibration and distribution estimation.

6.5 Calibration Results Evaluation

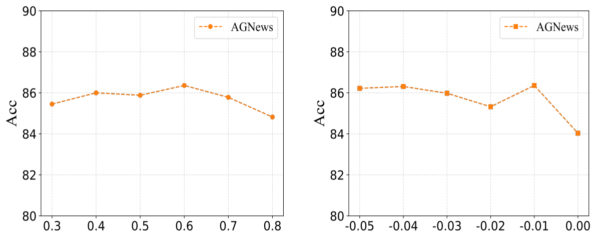
To validate PL-POT's pseudo-label calibration efficacy, we analyze epoch-wise correction results. Figure 3 shows that the number of right pseudo-labels (RN) are blue, while the error pseudo-labels (EN) are yellow. As the training progresses, the distinction between the two classes widens, although both blue and yellow show an upward trend, indicating that the accuracy of the pseudo-labeling calibration is improving. Figure 4(b) shows the calibration results based on the similarity between prototypes and text embeddings, and Figure 4(c) shows the calibration results based on the optimal transport method with balanced distribution. Both methods correct more incorrect than correct results. It illustrates the accuracy of our OT correction method based on estimated distributions.

7 Hyper-parameter Analysis

In this section, we perform parametric sensitivity analyses at the experimental setup of τ_u = 20 on AGNews. We report all the results (Acc) after 3 trials on validation set.

7.1 Impact of Correlation Coefficient λ

In our method, we use the correlation coefficient λ to control the ratio of the current estimation to the historical estimation. From Figure 5 (a), when λ is set to 0.6, PL-POT can get the best accuracy of 86.36%. It shows that both historical and current results are important for estimating the distribution of current unlabeled data.



(a) parameter 'λ'

(b) parameter 'α'

Figure 5: Hyper-parameter analysis on AGNews.

7.2 Impact of correlation coefficient α

In our method, we use the hyper-parameter α as correlation coefficient to control the flooding ratio on the calibrated results. From Figure 5 (b), the configuration of α is not sensitive to the model performance because it is automatically updated based on the training status. Nevertheless, it is important since when the α is set to 0, the result receives the largest decrease.

7.3 Impact of length of memory bank M

We analyze the Memory Bank (MB) length sensitivity across datasets with varying class counts and annotation sizes. Experimental results (Accuracy/Macro-F1) on AGNews and Yahoo reveal performance degradation when MB length exceeds 128. This decline suggests error accumulation from outdated pseudo-labels historical predictions generated in earlier training iterations that lose reliability over time. Notably, overextended MBs amplify noise by retaining suboptimal pseudo-labels beyond their temporal validity window.

M	AGNews	Yahoo
16	85.14/84.10	62.63/60.74
32	84.62/83.66	63.94/62.31
64	85.83/84.00	64.01/64.24
128	86.36/85.79	65.84/65.34
256	84.01/83.54	62.00/61.55
512	82.64/80.85	57.75/55.74

Table 5: The impact of memory bank length M.

8 Conclusion

In this paper, we propose a novel PL-POT model for SSTC task, which formulates the pseudo-labeling problem as an OT problem to alleviate the bias of imbalanced class data. Particularly, we calibrate the class distribution of unlabeled samples with reliable (pseudo-) labeled samples in MB. Then, we derive the semantic similarity between unlabeled samples and class prototypes as the coupling cost. By solving OT, each unlabeled sample is assigned with the most plausible class, and ensures the class distribution of pseudo-labels. Experimental results demonstrate PL-POT can effectively calibrate the pseudo-labels with the class distribution, and outperforms previous state-of-the-art methods.

9 Baselines of Semi-supervised Image Classification

In imbalanced semi-supervised learning, there are also methods in image classification. We also compared these methods with our method:

CDMAD (Lee and Kim, 2024) adopts a "white" embedding to to measure the degree of deviation of the pseudo-labeled data.

TCBC (Li et al., 2024b) makes twice class bias correction for imbalanced semi-supervised learning.

From the table 6, we can find that not all semi-supervised image classification methods can be applied to text classification problems, such as CDMAD with "white" images. On the other hand, the results of another image classification method TCBC is also worse than ours. TCBC also utilizes an estimate of the class distribution from the participating training samples to correct the model.

	$\tau_u = 5$		$\tau_u = 10$	
Method	Acc	F1	Acc	F1
CDMAD	-	-	-	-
TCBC	84.75	80.19	76.45	75.37
Ours	86.98	85.71	84.17	84.20

Table 6: Comparison with image classification tasks.

Limitations

Although PL-POT has been proven to be effective according to our extensive experiments, the current design of our method may not be optimal and could be improved in the future. Specifically, on the one hand, it becomes increasingly difficult to accurately estimate the estimated distribution as the number of pseudo-labels increases, which is an aspect that needs improvement in the future. On the other hand, we focus on the semi-supervised text classification, and it is also possible for our method to be applied to the field of images. In the future, we would investigate the applicability of our PL-POT model on the open-set semi-supervised image classification tasks.

10 Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. U2433212 and No. 62306026), in part by the National Science and Technology Major Project under Grant 2022ZD0120205, in part by the Fundamental Research Funds for the Central Universities, and in part by the State Key Laboratory of Complex & Critical Software Environment.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Yuan He, Xiaochun Cao, and Qingming Huang. 2022. Otkge: Multi-modal knowledge graph embeddings via optimal transport. *Advances in Neural Information Processing Systems*, 35:39090–39102.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45.

Ming-Wei Chang, Lev-Arie Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: Dataless classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 2, pages 830–835.

Wanxing Chang, Ye Shi, Hoang Tuan, and Jingya Wang. 2022. Unified optimal transport framework for universal domain adaptation. *Advances in Neural Information Processing Systems*, 35:29512–29524.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. Mixtext: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157.

Junfan Chen, Richong Zhang, Jiarui Wang, Chunming Hu, and Yongyi Mao. 2024. Self-paced pairwise representation learning for semi-supervised text classification. In *Proceedings of the ACM on Web Conference 2024*, pages 4352–4361.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. 2017. Joint distribution optimal transportation for domain adaptation. *Le Centre pour la Communication Scientifique Directe - HAL - UJM - Université Jean Monnet,Le Centre pour la Communication Scientifique Directe - HAL - UJM - Université Jean Monnet*.

Danilo Croce, Giuseppe Castellucci, and Roberto Basili. 2020. GAN-BERT: Generative adversarial learning for robust text classification with a bunch of labeled examples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2114–2119, Online. Association for Computational Linguistics.

Hongyan Cui, Gangkun Wang, Yuanxin Li, and Roy E Welsch. 2022. Self-training method based on gcn for semi-supervised short text classification. *Information Sciences*, 611:18–29.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26.

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

Daniela L Freire, Alex MG de Almeida, Márcio de S. Dias, Adriano Rivolli, Fabíola SF Pereira, Giliard A de Godoi, and Andre CPLF de Carvalho. 2023. Exploratory study of data sampling methods for imbalanced legal text classification. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 108–120. Springer.

Zheng Ge, Songtao Liu, Zeming Li, Osamu Yoshie, and Jian Sun. 2021. Ota: Optimal transport assignment for object detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lan-Zhe Guo and Yu-Feng Li. 2022. Class-imbalanced semi-supervised learning with adaptive thresholding. In *International conference on machine learning*, pages 8082–8094. PMLR.

Suchin Gururangan, Tam Dang, Dallas Card, and Noah A. Smith. 2019. Variational pretraining for semi-supervised text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5880–5894, Florence, Italy. Association for Computational Linguistics.

Nhat Ho, XuanLong Nguyen, Mikhail Yurochkin, Hung Hai Bui, Viet Huynh, and Dinh Phung. 2017. Multilevel clustering via wasserstein means. In *International conference on machine learning*, pages 1501–1509. PMLR.

Takashi Ishida, Ikko Yamane, Tomoya Sakai, Gang Niu, and Masashi Sugiyama. 2020. Do we need zero training loss after achieving zero training error? In *International Conference on Machine Learning*, pages 4604–4614. PMLR.

Jaehyung Kim, Youngbum Hur, Sejun Park, Eunho Yang, Sung Ju Hwang, and Jinwoo Shin. 2020. Distribution aligning refinery of pseudo-label for imbalanced semi-supervised learning. *Advances in neural information processing systems*, 33:14567–14579.

Zhengfeng Lai, Chao Wang, Henrry Gunawan, Sen-Ching S Cheung, and Chen-Nee Chuah. 2022. Smoothed adaptive weighting for imbalanced semi-supervised learning: Improve reliability against unknown distribution data. In *International Conference on Machine Learning*, pages 11828–11843. PMLR.

Hyuck Lee and Heeyoung Kim. 2024. Cdmad: Class-distribution-mismatch-aware debiasing for class-imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23891–23900.

Ju-Hyoung Lee, Sang-Ki Ko, and Yo-Sub Han. 2021. Salnet: Semi-supervised few-shot text classification with attention-based lexicon construction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13189–13197.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195.

Bin Li, Ye Shi, Qian Yu, and Jingya Wang. 2024a. Unsupervised cross-domain image retrieval via prototypical optimal transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 3009–3017.

Chen Li, Xutan Peng, Hao Peng, Jianxin Li, and Lihong Wang. 2021. Textgtl: Graph-based transductive learning for semi-supervised text classification via structure-sensitive interpolation. In *IJCAI*, pages 2680–2686.

Lan Li, Bowen Tao, Lu Han, De-chuan Zhan, and Han-jia Ye. 2024b. Twice class bias correction for imbalanced semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 13563–13571.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Nishant Mishra, Gaurav Sahu, Iacer Calixto, Ameen Abu-Hanna, and Issam Laradji. 2023. Llm aided semi-supervision for efficient extractive dialog summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10002–10009.

Wujian Peng, Zejia Weng, Hengduo Li, Zuxuan Wu, and Yu-Gang Jiang. 2025. Bmb: Balanced memory bank for long-tailed semi-supervised learning. *IEEE Transactions on Multimedia*.

Gabriel Peyré, Marco Cuturi, et al. 2019. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7.

Long Tian, Jingyi Feng, Xiaoqiang Chai, Wenchao Chen, Liming Wang, Xiyang Liu, and Bo Chen. 2023. Prototypes-oriented transductive few-shot learning with conditional transport. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16317–16326.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Austin Cheng-Yun Tsai, Sheng-Ya Lin, and Li-Chen Fu. 2022. Contrast-enhanced semi-supervised text classification with few labels.

Cédric Villani et al. 2009. *Optimal transport: old and new*, volume 338. Springer.

Cheng Wang, Jorge Balazs, György Szarvas, Patrick Ernst, Lahari Poddar, and Pavel Danchenko. 2022a. Calibrating imbalanced classifiers with focal loss: An empirical study. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 145–153.

Ximei Wang, Jinghan Gao, Mingsheng Long, and Jianmin Wang. 2021. Self-tuning for data-efficient deep learning. In *International Conference on Machine Learning*, pages 10738–10748. PMLR.

Yidong Wang, Hao Chen, Yue Fan, Wang Sun, Ran Tao, Wenxin Hou, Renjie Wang, Linyi Yang, Zhi Zhou, Lan-Zhe Guo, et al. 2022b. Usb: A unified semi-supervised learning benchmark for classification. *Advances in Neural Information Processing Systems*, 35:3938–3961.

Chen Wei, Kihyuk Sohn, Clayton Mellina, Alan Yuille, and Fan Yang. 2021. Crest: A class-rebalancing self-training framework for imbalanced semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10857–10866.

Ruixuan Xiao, Yiwen Dong, Junbo Zhao, Runze Wu, Minmin Lin, Gang Chen, and Haobo Wang. 2023. Freeal: Towards human-free active learning in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14520–14535.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33.

Weiyi Yang, Richong Zhang, Junfan Chen, Lihong Wang, and Jaein Kim. 2023. Prototype-guided pseudo labeling for semi-supervised text classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16369–16382.

Ruoyu Zhang, Yanzeng Li, Yongliang Ma, Ming Zhou, and Lei Zou. 2023. Llmaaa: Making large language models as active annotators. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13088–13103.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28:649–657.

Zefeng Zhang, Jiawei Sheng, Zhang Chuang, Liangyunzhi Liangyunzhi, Wenyuan Zhang, Siqi Wang, and Tingwen Liu. 2024. Optimal transport guided correlation assignment for multimodal entity linking. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 4103–4117. Association for Computational Linguistics.

Lvxing Zhu, Hao Chen, Chao Wei, and Weiru Zhang. 2022. Enhanced representation with contrastive loss for long-tail query classification in e-commerce. In *Proceedings of the Fifth Workshop on e-Commerce and NLP (ECNLP 5)*, pages 141–150.

Henry Zou and Cornelia Caragea. 2023. Jointmatch: A unified approach for diverse and collaborative pseudo-labeling to semi-supervised text classification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7290–7301.

Henry Zou, Yue Zhou, Weizhi Zhang, and Cornelia Caragea. 2023. Decrisismb: Debiased semi-supervised learning for crisis tweet classification via memory bank. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6104–6115.

A Optimal Transport

Optimal Transport (OT) is a mathematical framework that addresses the problem of finding the most cost-effective method for transferring mass from one probability distribution to another, with its conceptual origins tracing back to the pioneering work of Gaspard Monge (Villani et al., 2009) in the late 18th century. Let $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ be two probability measures defined on the same space $\mathbb{R}^n$, denoting the source and target distributions of resources, respectively. The *Monge problem* aims to find a mapping $\boldsymbol{\Pi} : \mathbb{R}^n \to \mathbb{R}^n$ that transforms a distribution $\boldsymbol{\mu}$ into a distribution $\boldsymbol{\nu}$ while minimizing the total transfer cost. This cost is usually measured by a cost function $c(x, \boldsymbol{\Pi}(x))$, where x is the original location of the resource and $\boldsymbol{\Pi}(x)$ is the new location to which it is transferred. It can be formulated as the following optimization problem:

$$\boldsymbol{\Pi}^* = \arg\min_{\boldsymbol{\Pi}} \int_{\mathbb{R}^n} c(x, \boldsymbol{\Pi}(x))\, d\mu(x), \tag{14}$$

where $\boldsymbol{\Pi}$ is the transport plan, $c(x, \boldsymbol{\Pi}(x))$ the cost function measuring from the x to the $\boldsymbol{\Pi}(x)$ transfer cost, and the integral the total cost over all x.

However, the Monge formulation of optimal transport imposes a restrictive assumption: the mass must be indivisible, meaning it requires a deterministic transport map $\boldsymbol{\Pi}$ that assigns each point in the source distribution to a single point in the target. This approach becomes infeasible when the two distributions have different total masses, or when the transport naturally requires a many-to-many correspondence between source and target points. To overcome these limitations, the *Kantorovich problem* generalizes the framework by seeking a probabilistic transport plan. In this problem, we consider two given probability distributions $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, defined on two spaces X and Y, respectively. The Kantorovich problem aims to find a joint probability distribution (i.e., transport plan) $\boldsymbol{\Pi}$ over $X \times Y$ whose marginal distributions are $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$, respectively, while minimizing the overall transport cost. Mathematically, this problem can be expressed as follows:

$$\begin{aligned} \boldsymbol{\Pi}^* &= \arg\min_{\boldsymbol{\Pi} \in \mathbb{R}_+^{n \times m}} \sum_{i=1}^{m} \sum_{j=1}^{n} \boldsymbol{C}_{ij} \boldsymbol{\Pi}_{ij}, \\ \text{s.t. } \boldsymbol{\Pi} \mathbf{1}_n &= \boldsymbol{\mu}, \quad \boldsymbol{\Pi}^\top \mathbf{1}_m = \boldsymbol{\nu}, \end{aligned} \tag{15}$$

where $\boldsymbol{\Pi}$ is a joint probability measure on $X \times Y$, representing the transition plan, and $c(x, y)$ is a cost function for moving from $x \in X$ to $y \in Y$. Notably, the Kantorovich problem can be solved by Sinkhorn-Knopp algorithm (Cuturi, 2013) efficiently. which substitutes the linear program problem by a smooth problem with an entropy regularization term as follows: The algorithm 1 shows the flow of the Sinkhorn-Knopp algorithm, where $\oslash$ denotes element-wise division.

Algorithm 1 Sinkhorn-Knopp Algorithm

```
Require: Cost matrix $\mathbf{C} \in \mathbb{R}^{n \times m}$, marginals $\boldsymbol{\mu} \in \mathbb{R}^n$, $\boldsymbol{\nu} \in \mathbb{R}^m$, regularization parameter $\lambda > 0$
Ensure: Approximate transport matrix $\mathbf{P}$
  Initialize $\mathbf{u} \leftarrow \mathbf{1}_n$, $\mathbf{v} \leftarrow \mathbf{1}_m$
  Compute $K \leftarrow \exp(-\lambda \mathbf{C})$
  while not converged do
    Update $\mathbf{u} \leftarrow \boldsymbol{\mu} \oslash (K\mathbf{v})$
    Update $\mathbf{v} \leftarrow \boldsymbol{\nu} \oslash (K^T \mathbf{u})$
  end while
  Compute $\mathbf{P} \leftarrow \text{diag}(\mathbf{u}) K \text{diag}(\mathbf{v})$
  return $\mathbf{P}$
```

B Hyper-parameter Settings

For reproduction, we report our hyper-parameter settings in Table 8 and Table 9. For a fair comparison, we implement PL-POT the same hyper-prameters of the encoder modules and same optimizer Adamw (Loshchilov and Hutter, 2018) as MixText. As shown in Table 8, the initial learning rate is tuned in [$1e^{-6}$, $1e^{-5}$] for BERT parameters and [$1e^{-4}$, $1e^{-3}$] for other parameters.

For PL-POT (shown in Table 9), the correlation coefficient λ that controls control the ratio of the current estimation to the historical estimation is set to 0.6 for AGNews and 0.5 for DBpedia and Yahoo for datasets with more classes correspond to a higher proportion of negative samples. For parameters of the OT part, the cost epsilon is set to 0.2 and the iteritions of sinkhorn algorithms is set to 20. The length of memory bank is set to 128. Note that the hyper-parameter settings are tuned on the valid data by grid search with 3 trials. We use the same experimental settings for all baseline models. We run all experiments on an NVIDIA Tesla P100 GPU with 16GB memory.

C Comparison with LLM

The emergence of the Large Language Model (LLM) era has ushered in a plethora of solutions for low - resource scenarios. Nevertheless,

MODEL	Alpaca-7B	LLaMA2	ChatGPT4	PL-POT($'\tau' = 5$)	PL-POT($'\tau' = 20$)
Acc	77.34	79.40	84.62	86.98	84.18
F1	78.27	80.67	84.50	85.71	84.08

Table 7: Comparison with LLM and PL-POT.

Hyper-parameter	Value
type embedding dimension d	768
Bert attention dropout	0.1
Bert hidden dropout	0.1
MLP hidden dimension	128
Sequence Length	256
Optimizer	Adamw
Learning rate of BERT	$1e^{-5}$
Learning rate of MLP	$1e^{-3}$
Training epoch	100
Steps/epoch	100
batch size on labeled data	4
batch size on unlabeled data	16

Table 8: Common hyper-parameter settings.

Hyper-parameter	Value
Warm step	1
Epsilon	0.2
Iteritions of sinkhorn	20
Correlation coefficient λ (AG News)	0.6
Correlation coefficient λ (DBpedia,Yahoo!Answer)	0.5
Flooding coefficient α	0.01
Threshold ξ	0.95
Length of memory bank M	128

Table 9: Hyper-parameter settings of PL-POT.

leveraging the Application Programming Interfaces (APIs) of large models comes with extra expenses, and directly applying open-source large models often fails to meet expectations. As depicted in the figure 7, in contrast to established large-model approaches like Alpaca-7B (Taori et al., 2023), LLaMA2 (Touvron et al., 2023), ChatGPT4 (Achiam et al., 2023), our proposed method not only demands significantly lower training costs but also delivers superior outcomes.

D Finer-grained Experiments

To demonstrate the generalizability of our method, we evaluate PL-POT on Banking (Casanueva et al., 2020), a financial intent detection dataset characterized by fine-grained class granularity with 77 categories, 9003 training samples, 3080 test samples, and an inherent class imbalance ratio of 4.5. Under a semi-supervised setup allocating 10 labeled samples per class alongside unlabeled training data, PL-POT achieves 82.95% accuracy and 82.04% F1-score averaged over three runs, outperforming UDA and PGPL. We set the confidence threshold to 0.5, and optimal transport cost ϵ=0.1. These results demonstrate PL-POT's adaptability to high-class-density scenarios, where its dynamic pseudo-label calibration via temporal reliability weighting and distribution alignment mitigates error propagation from historical predictions, contrasting with static threshold-based baselines.

Model	Accuracy	Macro-F1
Supervised	72.29	69.81
UDA	78.83	77.60
PGPL	81.77	80.25
PL-POT	82.95	82.04

Table 10: The results on Banking dataset.