LOPS: Learning Order Inspired Pseudo-Label Selection for Weakly Supervised Text Classification

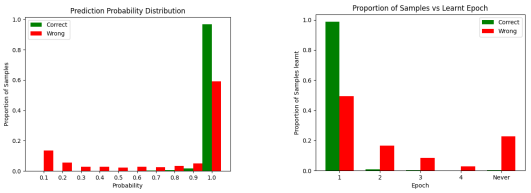
Anonymous ACL submission

Abstract

Iterative self-training is a popular framework in weakly supervised text classification that involves bootstrapping a deep neural classifier from heuristic pseudo-labels. The quality of pseudo-labels, especially the initial ones, is crucial to final performance but they are inevitably noisy due to their heuristic nature, so selecting the correct ones has a huge potential for performance boost. One straightforward solution is to select samples based on the softmax probability scores corresponding to their pseudo-labels. However, we show through our experiments that such methods are ineffective and unstable due to the erroneously high-confidence predictions from poorly calibrated models. Recent studies on the memorization effects of deep neural models suggest that these models first memorize training samples with clean labels and then those with noisy labels. Inspired by this observation, we propose a novel pseudo-label selection method LOPS that takes learning order of samples into consideration. We hypothesize that the learning order reflects the probability of wrong annotation in terms of ranking, and therefore, select the top samples that are learnt earlier. LOPS can be viewed as a strong performance-boost plug-in to most of existing weakly-supervised text classification methods, as confirmed in extensive experiments on six real-world datasets.

1 Introduction

Weakly supervised text classification has recently attracted much attention from researchers and the main-stream methods (Agichtein and Gravano, 2000; Riloff et al., 2003; Tao et al., 2015; Meng et al., 2018; Mekala and Shang, 2020; Mekala et al., 2020, 2021) follow an iterative self-training framework. As shown in Figure 2, these methods start with generating pseudo-labels, train a deep neural classifier to learn the mapping between documents and classes, and then bootstrap on unlabeled data.



(a) (b)

Figure 1: Distributions of correctly and wrongly labeled pseudo-labels using different selection strategies on the NYT coarse-grained dataset for its initial pseudo-labels. The base classifier is BERT. (a) is based on the softmax probability of samples' pseudo-labels and (b) is based on the earliest epochs at which samples are learnt.

The quality of the pseudo-labels, especially the initial ones, plays a crucial role in the final performance of these self-training-based methods. In weak supervision, people typically generate initial pseudo-labels by some heuristic, for example, through string matching between the documents and user-provided seed words. Therefore, pseudo-labels are inevitably noisy. A classifier trained on such noisy labels has a high risk of making erroneous predictions, worsening the quality of pseudo-labels in next iterations and upon bootstrapping significantly hurting the final performance.

A straightforward solution to address this problem is to threshold samples by the softmax probability scores corresponding to their pseudo-labels. However, deep neural networks (DNNs) usually have a poor calibration and generate overconfident predicted probability scores (Guo et al., 2017). For example, as shown in Figure 1(a), 60% of wrongly-labeled samples in noisy New York Times (NYT) coarse-grained dataset have a predicted probability by BERT greater than 0.9 for their pseudo-labels, and 0.9 is generally considered to be high probability. Although there are recent works that use uncertainty to fix calibration (Rizve et al., 2021), they require a validation set, which is unavailable under the weakly supervised setting. As a result,

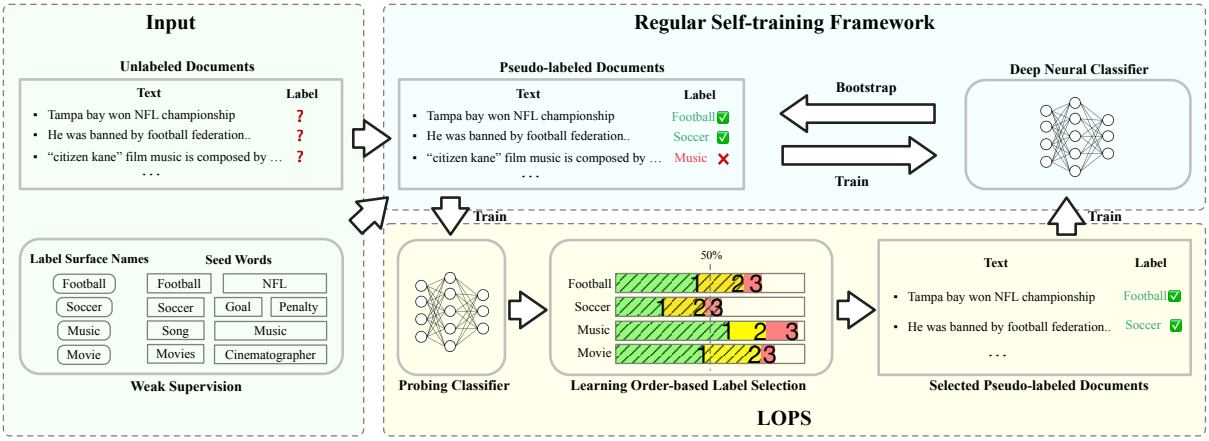


Figure 2: Usually self-training frameworks follow the path in the top block, starting from generating noisy pseudo-labeled documents, training the text classifier, and bootstrapping by adding high confidence predictions. We propose to add a step "Label Selection" (shown in below block) to select the correctly labeled documents. LOPS trains a classifier to obtain the learning order of samples and we stop the training when at least 50% of samples corresponding to each class are learnt. The numbers shown are learnt epochs and the samples lying in the shaded part are selected.

probability score-based selection is not appropriate here. There are other lines of work focusing on label selection from noisy data using co-teaching, curriculum learning (Jiang et al., 2018; Han et al., 2018), and weighting the instances for selective training (Ren et al., 2018; Fang et al., 2020). However, all these methods require clean validation sets to infer the parameters, whereas in our problem, we have no clean annotated data.

Recent studies on the memorization effects of DNNs show that they memorize easy and clean instances first, and gradually learn hard instances and eventually memorize the wrong annotations (Arpit et al., 2017; Geifman et al., 2018; Zhang et al., 2021). In our experiments on text classification tasks, we observe the same pattern for different classifiers. For example, as shown in Figure 1(b), BERT classifier learns most of the clean instances in the first epoch and learns wrong instances across all epochs. Although it also learns good number of wrong instances in the first epoch, it is significantly less than the probability-based selection in Figure 1(a). Since the correct samples are learnt first, we hypothesize that learning order-based selection will be able to filter out the wrongly labeled samples.

Illuminated by our observation, we propose a novel learning order inspired pseudo-label selection method LOPS. As shown in Figure 2, we propose to add a "Label Selection" step after generating pseudo-labels and train the classifier on selected pseudo-labeled documents. LOPS involves training a classifier and tracking the learning order of samples and we stop the training when at least 50% of samples corresponding to each class are learnt. Specifically, we define a sample is learnt if and only if the classifier trained on pseudo-labels gives the same argmax prediction as its pseudo-label at the end of an epoch. We empirically show that LOPS improves the vanilla self-training methods and it is much more effective and stable than probability score-based selections.

Our contributions are summarized as follows:

- We propose a novel pseudo-label selection method LOPS that takes learning order of samples into consideration.
- We show that selection based on learning order is much stable and effective than selection based on probability scores.
- Extensive experiments and case studies on real-world datasets with different classifiers and weakly supervised text classification methods demonstrate significant performance gains upon using LOPS. It can be viewed as a solid performance-boost plug-in for weak supervision.

Reproducibility. We will release the code and datasets on Github[1].

2 Related Work

We review the literature about (1) pseudo-labeling in weakly supervised text classification, (2) label

[1] https://github.com/anonymous

selection methods, and (3) learning dynamics.

2.1 Pseudo-Labeling in Weakly Supervised Text Classification

Since the weakly supervised text classification methods lack gold annotations, pseudo-labeling has been a common phenomenon to generate initial supervision. Pseudo-labeling procedure depends on the type of weak supervision. Mekala and Shang (2020) and Mekala et al. (2020) have a few label-indicative seed words as supervision and they generate pseudo-labels using string-matching where a document is assigned a label whose aggregated term frequency of seed words is maximum. (Meng et al., 2018) generates pseudo-documents using the seed information corresponding to a label. (Wang et al., 2020) takes only label names as supervision and generates class-oriented document representations, and cluster them to create a pseudo-training set. Under the same scenario, (Mekala et al., 2021) consider samples that exclusively contain the label surface name as its respective weak supervision. In (Karamanolakis et al., 2021b), pseudo-labels are created from the predictions of a trained neural network. All the above mentioned methods involve learning from noisy data and our label selection method substantially reduces the noise and improves their performance.

2.2 Label Selection

There are different lines of work aiming to select true-labeled examples from a noisy training set. One line of work involves training multiple networks to guide the learning process. Along this direction, (Malach and Shalev-Shwartz, 2017) maintains two DNNs and update them based on their disagreement. (Jiang et al., 2018) learns another neural network that provides data-driven curriculum. (Han et al., 2018; Yu et al., 2019) use co-training where they select instances based on small loss criteria and cross-train two networks simultaneously. Another line of work learns weights for the training data. Along this line, (Ren et al., 2018) propose a meta-learning algorithm that learns weights corresponding to training examples based on their gradient directions. (Fang et al., 2020) learns dynamic importance weighting that iterates between weight estimation and weighted classification. weighting the instances for selective training (Ren et al., 2018; Fang et al., 2020). Recently, (Rizve et al., 2021) propose utilizing prediction uncertainty to perform label selection. All the above-mentioned methods require clean validation sets to infer parameters, whereas our method needs no clean annotated data. Inspired from the recent studies on memorization effects of DNNs that they learn clean data earlier than noisy data, we use learning order to select the samples.

2.3 Learning dynamics

In deep learning regime, models with large capacity are typically more robust to outliers. Nevertheless, data examples can still exhibit diverse levels of difficulties. Arpit et al. (2017) finds that data examples are not learned equally when injecting noisy data into training. Easy examples are often learned first. Hacohen et al. (2019) furthers shows such order of learning examples is shared by different random initializations and neural architectures. Toneva et al. (2019) shows that certain examples are forgotten frequently during training, which means that they can be first classified correctly then incorrectly. Model performance can be largely maintained when removing those least forgettable examples from training.

3 Problem Formulation

The input of our problem contains: (1) n unlabeled text documents $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_n\}$, (2) m target classes $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m\}$ and (3) a source of weak supervision $\mathcal{W}$. Using the weak supervision $\mathcal{W}$, a subset of unlabeled text documents are pseudo-labeled to generate noisy training data $\hat{\mathcal{D}} = \{\hat{\mathcal{D}}_{\mathcal{C}_1} \cup \hat{\mathcal{D}}_{\mathcal{C}_2} \cup \ldots \hat{\mathcal{D}}_{\mathcal{C}_m}\}$ where $\hat{\mathcal{D}}_{\mathcal{C}_i}$ denotes the samples that are pseudo-labeled as $\mathcal{C}_i$. $\hat{\mathcal{D}}$ can be partitioned as correctly labeled samples $\hat{\mathcal{D}}_{\checkmark}$ and wrongly labeled samples $\hat{\mathcal{D}}_{\times}$ based on the underlying groundtruth labels. We aim to select $\hat{\mathcal{D}}_{\checkmark}$ from noisy training data $\hat{\mathcal{D}}$ and filter out wrongly annotated samples $\hat{\mathcal{D}}_{\times}$.

Note that, we have no clean annotated data. Also, there is no restriction on source of supervision $\mathcal{W}$. It can be just the label surface names (Wang et al., 2020), label-indicative seed words (Mekala and Shang, 2020), or rules (Karamanolakis et al., 2021a).

4 Pseudo-Labels are Noisy

Pseudo-labeling is the process of generating labels for unlabeled samples to guide the learning process. In the context of weakly supervised learning, where we don't have any annotated samples, initial pseudo-labels are generated using some heuris-

Table 1: Dataset statistics.

Dataset	# Docs	# labels	Avg Len
NYT-Coarse	13081	5	778
NYT-Fine	13081	26	778
20News-Coarse	17871	5	400
20News-Fine	17871	17	400
AGNews	120000	4	426
Books	33594	8	620

tics like counting and string-matching utilizing the weak supervision. Since this process is heuristic, the initial pseudo-labels are noisy.

We consider New York Times fine-grained dataset and generate pseudo-labels using different heuristics from (Mekala and Shang, 2020; Mekala et al., 2020, 2021; Wang et al., 2020) and compute noise ratio shown in Table 2. The first document is incorrectly pseudo-labeled as *football* by all string-match based strategies. Football and soccer are used interchangeably and the string-match strategies assign *football* for the second document whereas the contextualization helps by identifying the interpretation and assigns correct pseudo-label. We can observe that no strategy is perfect and all of them generate noisy labels with significantly high noise ratio.

When a classifier is trained on such noisy training data, it can make some high confident erroneous predictions. And, upon bootstrapping the classifier on unlabeled data, it has a snowball effect where such high confident erroneous predictions are added to the training data, and thus corrupting it more. As this process repeats for a few iterations, it adds more noise and significantly effects the final performance. The number of iterations of self-training is a key hyper-parameter to tune and we cannot apply self-training for too many iterations as the performance typically improves in the beginning but later, drops down significantly. For example, the macro f1-score of ConWea (Mekala and Shang, 2020) on 20Newsgroup coarse-grained dataset with 30% noise, increases for the first two iterations to 75% and drops down to 56% by sixth iteration.

As shown in Table- 2, pseudo-label heuristics generate significantly noisy training data. Therefore, identifying and selecting the correctly labeled samples is necessary and has a huge potential for a boost in performance. Note that, if the labels are not selected carefully, it could instead hurt the performance.

5 Probability-based Pseudo-label Selection: An Intuitive Baseline

One intuitive way is to select the samples based on model's prediction probability scores. Specifically, we train a classifier on pseudo-labeled data and predict on the same data and using the prediction probabilities corresponding to pseudo-labels, filter out the low confidence samples based on a threshold.

However, many of these selected predictions are usually incorrect due to the poor calibration of neural networks (Guo et al., 2017). For example, in NYT coarse-grained dataset, the average confidence score of correctly-labeled samples is 0.98 and wrongly-labeled samples is 0.72 and as shown in Figure 1(a), 60% incorrectly labeled samples have probability more than 0.9.

As a result of poor calibration in DNNs, the prediction probability scores are densely distributed and very close, due to which, choosing a threshold is difficult. Moreover, selection based on an absolute, fixed threshold for all datasets is not feasible as the distribution of prediction probability varies across different datasets. And, selection based on quantile suffers from poor calibration that causes a low-entropy probability distribution. Therefore, filtering based on such sensitive, poorly calibrated probability-based threshold is unstable and has high variance across multiple runs, as confirmed in our experiments.

6 LOPS: Our Pseudo-label Selection

In this section, we describe LOPS, our label selection method.

Our selection method takes learning order into consideration. It is based on the recent studies that a DNN learns clean instances first and gradually memorizes the wrongly annotated samples. We call a sample in training data being learnt, when the model's prediction of that sample matches the training label. We define learning order of the training data as a collection of epochs at which each sample is learnt, sorted in ascending manner. We calculate the learning order at the granularity of epoch because the model would have seen all the training data by the end of an epoch, and hence, the learning order computed would be fair for all samples. And, if needed, it's easy to extend it to the more fine-grained granularity such as batches.

As the model is known to learn clean instances first, we hypothesize that this learning order reflects

Table 2: Pseudo-labels generated using different heuristics on NYT-Fine dataset and their respective noise ratios. Incorrect pseudo-labels are highlighted in *red* and correct pseudo-labels are highlighted in *green*. N/A denotes no pseudo-label assigned.

Input Docs (unlabeled)	**"Class": [Seed Words]**
1. Tom aikens, a michelin-starred chef, says running a restaurant is same as managing a football team. 2. Genoa defender giovanni marchese was handed bans by italian football federation 3. orson welles made his debut in "citizen kane". It's music was composed by paul bowles..	"soccer": ["soccer"], "football": ["football"], "music": ["music"], "movies": ["movies"]

Pseudo-label Heuristic	**Generated Initial Pseudo-labels**	**# of Pseudo-labels**	**Noise Ratio**
String-Match (Mekala et al., 2020)	1. football, 2. football, 3. music	8229	31.80%
Contextualized String-Match (Mekala and Shang, 2020)	1. football, 2. soccer, 3. music	8411	31.24%
Exclusive String-Match (Mekala et al., 2021)	1. football, 2. football, 3. N/A	3512	52.13%
Clustering (Wang et al., 2020)	1. business, 2. football, 3. movies	5865	15.64%

the probability of wrong annotation in terms of ranking. In a preliminary experiment on noisy New York Times coarse-grained dataset with BERT (Devlin et al., 2018) as classifier, we plot the distribution of epochs at which each sample is learnt for correctly labeled and wrongly labeled samples shown in Figure 1(b). We can observe that there is a clear demarcation between the epochs at which correctly-labeled and wrongly-labeled samples are learnt. Almost all the correctly labeled samples are learnt in first epoch where as the wrongly labeled samples are learnt across all epochs. Moreover, there is a significant proportion of wrongly labeled samples that are never learnt.

Motivated by this observation, for every label, we select its corresponding training samples that are learnt early. Moreover, following (Mekala and Shang, 2020), we assume that weak supervision $\mathcal{W}$ is of reasonable quality i.e. majority of pseudo-labels are good. Therefore, we select top-50% of samples for each label based on their learning order. Specifically, we train a classifier and obtain predictions of all samples in training data at the end of each epoch and track their learning order. If our selected bucket doesn't contain 50% or more samples corresponding to a label yet, we add the new learned ones belonging to that label. Finally, we stop the training when at least 50% of samples corresponding to every label are learnt.

We summarize our method using pseudo-code in Algorithm 1. LOPS can be viewed as a performance-boost plug-in for weakly supervised text classification.

Algorithm 1: LOPS Method

Input: Noisy training data $\hat{\mathcal{D}}$, Classifier C.
Output: Selected samples $\mathcal{D}_{sel}$
for *epoch* $e \in \{1, 2, 3, \ldots n_{ep}\}$ **do**
  Train C on $\hat{\mathcal{D}}$
  Obtain predictions on $\hat{\mathcal{D}}$ using C
  for each label l **do**
    if $\mathcal{D}_{sel}$ *contains* $< 50\%$ *of* $\hat{\mathcal{D}}_l$ **then**
      $\mathcal{D}_{sel}(l)$ = samples with label l learnt in epoch e
      $\mathcal{D}_{sel} = \mathcal{D}_{sel} \cup \mathcal{D}_{sel}(l)$
  if $\%$ *of learnt samples* > 50 *for all labels* **then**
    Break
  $e = e + 1$
Return $\mathcal{D}_{sel}$

7 Experiments

In this section, we evaluate our label selection method on different state-of-the-art classifiers and weakly supervised text classification frameworks.

7.1 Datasets

We experiment on three datasets. The dataset statistics are provided in Table 1. The details of datasets are provided below:

- **The New York Times (NYT):** The NYT dataset is a collection of news articles published by The New York Times. They are classified into 5 coarse-grained genres (e.g., science, sports) and 25 fine-grained categories (e.g., music, football, dance, basketball).
- **The 20 Newsgroups (20News):** The 20News dataset[2] is a collection of newsgroup documents partitioned widely into 6 groups (e.g., recreation, computers) and 20 fine-grained classes (e.g., graphics, windows, baseball, hockey). Following (Wang et al., 2020), coarse- and fine-grained miscellaneous labels are ignored.
- **AGNews** (Zhang et al., 2015) is a huge collection of news articles categorized into four coarse-grained topics such as business, politics, sports, and technology.
- **Books** (Wan and McAuley, 2018; Wan et al., 2019) is a dataset containing description of books, user-book interactions, and users' book reviews

[2] http://qwone.com/~jason/20Newsgroups/

Table 3: Evaluation results on six datasets using different combinations of classifiers and selection methods. Initial pseudo-labels are generated using String-Match. Micro and Macro f1 scores and their respective standard deviations are presented in percentages. Abnormally high standard deviations are highlighted in *blue* and low performances are highlighted in *red*.

Classifier	Method	NYT-Coarse		NYT-Fine		20News-Coarse		20News-Fine		AGNews		Books	
		mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1
BERT	No-Filter	90(0.17)	80(0.91)	77(0.36)	71(0.43)	77(0.27)	76(0.76)	70(0.30)	69(0.25)	75(0.64)	75(0.47)	55(0.54)	57(0.82)
	Random	90(0.47)	80(0.47)	78(0.94)	71(0.47)	79(1)	76(1.5)	71(0.5)	70(1)	76(0.35)	76(0.65)	56(0.18)	58(0.35)
	Probability	92(1.5)	85(2)	46(2.5)	22(0.5)	78(2.5)	77(3)	47(23.5)	47(23.5)	77(1.25)	77(1.34)	54(1.12)	56(1.43)
	Stability	93(0.5)	86(0.5)	48(29.5)	35(33.5)	76(5)	75(5)	**73(0.5)**	**72(1)**	**79(0.75)**	**79(0.35)**	55(0.43)	57(0.19)
	LOPS	**94(0.36)**	**88(0.5)**	**84(0.54)**	**81(0.34)**	**81(1)**	**80(0.43)**	**73(0.61)**	**72(1)**	**79(0.86)**	**79(0.58)**	**57(0.87)**	**59(0.46)**
	Upperbound	98(0.27)	96(0.37)	97(0.71)	92(0.62)	94(0.37)	94(0.61)	87(0.37)	86(0.36)	89(0.46)	89(0.76)	76(0.21)	76(0.19)
RoBERTa	No-Filter	90(0.41)	82(0.24)	79(0.65)	76(0.54)	76(0.41)	75(0.58)	67(0.67)	67(0.87)	74(0.44)	74(0.71)	57(0.29)	58(0.53)
	Random	92.33(0.21)	84(0.82)	76(1.25)	74(0.34)	76(1)	74(1)	68(0.23)	68(0.23)	74(0.32)	74(0.27)	56(0.57)	58(0.32)
	Probability	**93(0.48)**	**87(1)**	26(23)	14(11.5)	76(0.5)	75(1)	46(23)	45(23.5)	76(0.89)	76(1.12)	56(1.28)	57(1.85)
	Stability	90(1.09)	83(0.5)	21.5(12.5)	9(5)	**78(1)**	**76(1.5)**	70(1)	70(1)	**76(0.48)**	**76(0.64)**	58(1.18)	59(1.06)
	LOPS	92(2.99)	85(3)	**81(0.9)**	**80(0.5)**	77(2)	75(2)	**70(0.68)**	**70(0.34)**	75(0.22)	75(0.27)	**59(0.41)**	**60(0.45)**
	Upperbound	98(0.17)	96(0.16)	97(0.34)	92(0.26)	94(0.74)	94(0.35)	85(0.32)	85(0.65)	89(0.17)	89(0.28)	76(0.29)	77(0.22)
XLNet	No-Filter	89(0.74)	80(0.64)	77(0.34)	71(0.75)	77(0.39)	75(0.68)	60(0.74)	66(0.61)	72(0.97)	72(0.53)	57(0.31)	58(0.46)
	Random	90(0.03)	80(0.51)	76(0.94)	72(0.7)	78(0.5)	75(1)	67(0.49)	67(0.32)	67(0.22)	67(0.63)	57(0.43)	58(0.45)
	Probability	91(0.29)	**83(0.5)**	38(6.5)	36(1)	77(1)	75(0.3)	69(0.82)	69(0.12)	70(1.09)	70(1.14)	54(1.42)	56(1.26)
	Stability	**91(1)**	82(1.5)	79(0.5)	76(1.1)	79(1.5)	77(1.5)	68(0.49)	68(1)	74(1.1)	74(0.87)	56(0.88)	58(0.97)
	LOPS	89(0.17)	81(0.9)	**80(0.22)**	**77(0.83)**	**82(0.5)**	**81(0.2)**	**70(0.31)**	**70(0.27)**	**77(0.57)**	**77(0.54)**	**58(0.65)**	**59(0.67)**
	Upperbound	98(0.12)	96(0.21)	97(0.32)	93(0.38)	94(0.23)	94(0.29)	86(0.43)	86(0.35)	89(0.28)	89(0.39)	76(0.44)	76(0.43)
GPT-2	No-Filter	91(0.24)	82(0.28)	76(0.41)	69(0.38)	78(0.26)	76(0.38)	70(0.46)	70(0.38)	61(0.28)	61(0.43)	51(0.41)	53(0.37)
	Random	90(0.42)	80(0.56)	77(0.52)	70(1.02)	79(0.46)	78(0.32)	69(0.21)	69(0.29)	68(0.18)	68(0.19)	53(0.46)	55(0.42)
	Probability	93(1.04)	85(1.13)	76(0.57)	71(0.69)	80(1.49)	78(1.50)	69(1.21)	69(1.18)	66(0.69)	66(0.89)	51(1.11)	54(1.09)
	Stability	94(0.56)	88(0.59)	79(0.62)	75(0.65)	81(1.02)	78(1.50)	**70(0.68)**	**70(0.63)**	72(0.58)	72(0.53)	53(1.02)	55(1.13)
	LOPS	**95(0.49)**	**89(0.51)**	**80(0.09)**	**76(0.21)**	**82(0.57)**	**80(0.63)**	**70(0.76)**	**70(0.48)**	**75(0.52)**	**75(0.31)**	**56(0.89)**	**58(0.63)**
	Upperbound	98(0.24)	96(0.21)	97(0.18)	92(0.19)	94(0.23)	93(0.27)	86(0.35)	85(0.38)	88(0.26)	88(0.28)	72(0.19)	73(0.22)

collected from a popular online book review website Goodreads[3]. Following (Mekala et al., 2020), we select books belonging to eight popular genres. Using the title and description as text, we aim to predict the genre of a book.

7.2 Compared Label Selection Methods

We compare with several metrics used for label selection mentioned below:

- **Probability:** We sort the prediction probabilities corresponding to pseudo-labels in descending order and select the same number of samples as LOPS in each iteration of bootstrapping.
- **Random:** We randomly select the same number of samples as LOPS in each iteration of bootstrapping. To avoid the label imbalance after selection, we sample in a stratified fashion based on class labels.
- **Learning Stability (stability):** (Dong et al., 2021) introduced a metric to measure the data quality based on the frequency of events that an example is predicted correctly throughout the training. We sort the samples based on learning stability in descending order i.e. most stable to least stable and select the same number of samples as LOPS in each iteration of bootstrapping.

We consider the same number of samples as LOPS in each iteration for all above baselines because we cannot tune individual thresholds for each dataset since there is no clean data under the weakly supervised setting and one fixed threshold for all datasets doesn't work as the distribution of prediction probability varies across datasets. So, to perform controlled experiments with a fair comparison, we consider the same number of samples as LOPS in each iteration.

We also present experimental results without any label selection (denoted by ***No-Filter***) as lower bound and with all the wrongly annotated samples removed as ***Upperbound***.

7.3 Experimental Settings

Seed Words. For all our experiments, we consider label-indicative seed words used in (Mekala and Shang, 2020; Wang et al., 2020) as weak supervision and generate initial pseudo-labels using **String-Match** (Mekala et al., 2020) unless specified.

Text Classifiers. We experiment on four state-of-the-art text classifiers: (1) **BERT** (`bert-base-uncased`) (Devlin et al., 2018), (2) **RoBERTa** (`roberta-base`) (Liu et al., 2019), (3) **XLNet** (`xlnet-base-cased`) (Yang et al., 2019), and (4) **GPT-2** (Radford et al., 2019).

We follow the same self-training method for all these classifiers that starts with generating pseudo-labels, training a classifier on pseudo-labeled data, and bootstrap it on unlabelled data by adding samples whose prediction probabilities are greater than

[3] https://www.goodreads.com/

Table 4: Evaluation results of weakly supervised text classification frameworks with LOPS label selection method. This demonstrates that LOPS can be easily plugged in and improves the performance.

Framework	Method	NYT-Coarse		NYT-Fine		20News-Coarse		20News-Fine		AGNews		Books	
		mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1	mi-f1	ma-f1
ConWea	No-Filter	93	87	**87**	77	74	74	68	68	73	73	52	52
	LOPS	**94**	**90**	**87**	**78**	**79**	**78**	**70**	**70**	**79**	**79**	**57**	**58**
X-Class	No-Filter	**96**	**93**	**86**	**74**	58	61	70	70	82	**82**	53	54
	LOPS	**96**	**93**	**86**	**74**	**60**	**62**	**71**	**71**	**83**	**82**	**54**	**56**

δ. The pseudo-code of a self-training weakly supervised text classification framework with label selection is shown in Algorithm 2 in Appendix A.

While training the classifiers, we fine-tuned RoBERTa for 3 epochs, BERT, XLNet, GPT-2 for 4 epochs. We bootstrapped all the classifiers for 5 iterations and while bootstrapping, we set the probability threshold δ as 0.6 to select the confident predictions.

Weakly Supervised Text Classification Frameworks. We also experiment on state-of-the-art weakly supervised text classification methods described below.[4]

- **ConWea** (Mekala and Shang, 2020) is a seed-word driven iterative framework that uses pre-trained language models to contextualize the weak supervision.
- **X-Class** (Wang et al., 2020) takes only label surface names as supervision and learns class-oriented document representations. These document representations are aligned to classes, computing pseudo labels for training a classifier.

We use the public implementations of ConWea[5] and X-Class[6] and modify them to plug-in our filter. Specifically, in ConWea, we add our filter before training the text classifier and for X-Class, we plug-in our filter after learning the document-class alignment.

7.4 Quantitative Results

We discuss the effectiveness of LOPS with different classifiers and weakly supervised text classification frameworks.

7.4.1 Evaluation results with different classifiers

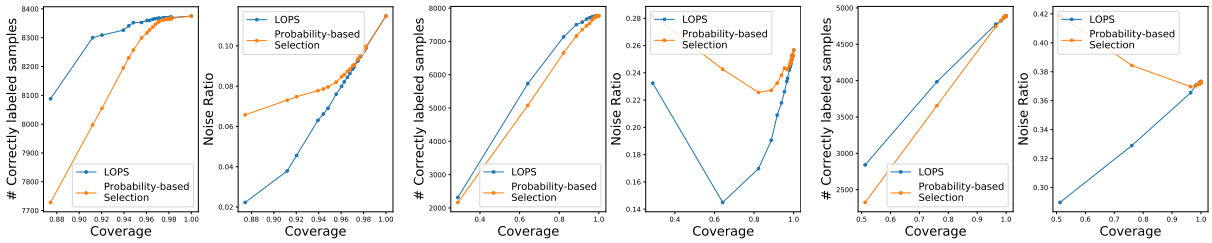
We summarize the evaluation results with different combinations of classifiers and selection methods in Table 3. All experiments are run on three random seeds and mean, standard deviations are reported in percentages. We discuss the effectiveness of LOPS as follows:

- As shown in table 3, upon plugging our proposed method LOPS, we observe a significant boost in performance over No-Filter with all the classifiers. In some cases like BERT on NYT-Fine, the improvement is as high as 7 points on micro-f1 and 10 points on macro-f1.
- We observe that LOPS always outperforms random selection which shows that the selection in LOPS is strategic and principled.
- LOPS performs better than probability and stability based selection methods in most of the cases. This shows that LOPS is very effective in removing the wrongly labeled samples and preserving the correctly labeled samples.
- We observe abnormally low performances of probability and stability based selection methods in some scenarios (highlighted in *red* in Table 3). This is because the probability and stability scores are so densely distributed that many wrongly labeled samples are selected that significantly effected the performance, which got worsened with iterative self-training.
- We have to note unusually high standard deviation for probability and problematic score based selection methods in some cases(highlighted in *blue* in Table 3). This demonstrates that these selection methods are unstable. LOPS is comparatively more stable and its effectiveness is largely due to its invariance.
- Although probability and stability based selection methods outperform LOPS in a few cases, their unstable nature makes them unreliable. Therefore, we believe LOPS is a superior method than other compared selection methods.
- We observe that LOPS uplifts the performance quite close to supervised methods. This demonstrates that LOPS acts as an effective plugin and helps in closing the performance gap between the weakly supervised and supervised settings.

[4] We also considered experimenting on ASTRA, however the instructions to run on custom datasets were not made public yet.

[5] https://github.com/dheeraj7596/ConWea

[6] https://github.com/ZihanWangKi/XClass



(a) NYT-Coarse (b) 20News-Fine (c) Books

Figure 3: Risk vs Coverage Analysis: we plot # correctly labeled samples and noise ratio in selected subset by LOPS and probability-based method on NYT-Coarse, 20News-Fine, and Books datasets using BERT classifier.

Table 5: Incorrectly pseudo-labeled samples selected by probability-based selection are shown below. These samples are learnt at later epochs, thus LOPS avoids selecting them.

Document	Pseudo-label
Corinthians have received offer from tottenham hotspur for brazil's paulinho although the midfielder said on saturday he would not decide his future until after the confederations cup ."there is an official offer from tottenham to corinthians but, as i did when there was an inter milan offer, i'll sit and decide with my family before i make any decision," paulinho told reporters.	Football Softmax Prob: 0.96 Learnt Epoch: 2
Brittney griner and elena delle donne were poised to make history as the first pair of rookies from same class to start wnba all-star game. Now, neither will be playing as both are sidelined with injuries. It's a tough blow for the league, which has been marketing the two budding stars.	Baseball Softmax Prob: 0.96 Learnt Epoch: 2
Denmark central defender simon kjaer has joined french side lille from vfl wolfsburg on a four-year deal. Lille paid two million euros. 72 million pounds for the 24-year-old kjaer, who has won 35 caps for his country. He joined wolfsburg from palermo for 12 million euros.	Intl. Business Softmax Prob: 0.94 Learnt Epoch: 2
Fiorentina striker giuseppe rossi is quickly making up for lost time after suffering successive knee ligament injuries which kept him out of action for the best part of two years.	Football Softmax Prob: 0.95 Learnt Epoch: 2

7.4.2 Evaluation results with different weakly supervised text classification methods

We summarize the evaluation results with different weakly supervised methods in Table 4. The results demonstrate that LOPS improves the performance of ConWea significantly and X-Class sometimes. Note that, X-Class sets a confidence threshold and selects only top-50% instances. So, this selection already provides a hidden advantage and LOPS improves the performance on top of it.

7.5 Risk-Coverage Analysis

We perform risk-coverage analysis by plotting the number of correctly labeled samples selected and noise ratio vs coverage for both LOPS and probability-based selection methods on three datasets NYT-Coarse, 20News-Fine, Books using BERT classifier shown in Figure 3. Coverage is defined as the proportion of samples selected after executing the selection method and noise ratio is the proportion of wrongly annotated documents in the selected documents. We can observe that the number of correctly labeled samples selected is higher for LOPS than probability-based selection for all datasets. And also, noise ratio is lower for LOPS than probability-based selection method on all datasets. This plot clearly shows that LOPS is much effective than probability-based selection.

7.6 Example samples

A few incorrectly pseudo-labeled samples from NYT-Fine dataset that are selected by probability-based selection with RoBERTa as classifier are shown in Table 5. We observe a high probability assigned to each incorrect pseudo-label whereas these are learnt by the classifier at later epochs. These wrongly annotated samples induce error that gets propagated and amplified over the iterations. By not selecting these wrong instances, LOPS curbs this and boosts the performance.

8 Conclusion and Future Work

In this paper, we proposed LOPS, a novel learning order inspired pseudo-label selection method. Our method is inspired from recent studies on memorization effects that showed that clean samples are learnt first and then wrong samples are memorized. Experimental results demonstrate that our method is effective, stable and can act as a performance boost plugin on many text classifiers and weakly supervised text classification methods. It outperforms several label selection methods based on probability and learning stability. In the future, we are interested in analyzing the role of noise and investigate any positive consequences of noise in text classification.

9 Ethical Consideration

This paper proposes a label selection method for weakly supervised text classification frameworks. The aim of the paper is to detect the noise caused by the heuristic pseudo-labels and we don't intend to introduce any biased selection. Based on our experiments, we manually inspected some filtered samples and we didn't find any underlying pattern. Hence, we do not anticipate any major ethical concerns.

References

Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 85–94. ACM.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. 2017. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, pages 233–242. PMLR.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Chengyu Dong, Liyuan Liu, and Jingbo Shang. 2021. Data profiling for adversarial training: On the ruin of problematic data. *CoRR*, abs/2102.07437.

Tongtong Fang, Nan Lu, Gang Niu, and Masashi Sugiyama. 2020. Rethinking importance weighting for deep learning under distribution shift. *arXiv preprint arXiv:2006.04662*.

Yonatan Geifman, Guy Uziel, and Ran El-Yaniv. 2018. Bias-reduced uncertainty estimation for deep neural classifiers. *arXiv preprint arXiv:1805.08206*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR.

Guy Hacohen, Leshem Choshen, and D. Weinshall. 2019. Let's agree to agree: Neural networks share classification order on real datasets. *arXiv: Learning*.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *arXiv preprint arXiv:1804.06872*.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. 2018. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pages 2304–2313. PMLR.

Giannis Karamanolakis, Subhabrata Mukherjee, Guoqing Zheng, and Ahmed Hassan Awadallah. 2021a. Self-training with weak supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 845–863, Online. Association for Computational Linguistics.

Giannis Karamanolakis, Subhabrata (Subho) Mukherjee, Guoqing Zheng, and Ahmed H. Awadallah. 2021b. Self-training with weak supervision. In *NAACL 2021*. NAACL 2021.

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

Eran Malach and Shai Shalev-Shwartz. 2017. Decoupling" when to update" from" how to update". *arXiv preprint arXiv:1706.02613*.

Dheeraj Mekala, Varun Gangal, and Jingbo Shang. 2021. Coarse2fine: Fine-grained text classification on coarsely-grained annotated data. *arXiv preprint arXiv:2109.10856*.

Dheeraj Mekala and Jingbo Shang. 2020. Contextualized weak supervision for text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 323–333.

Dheeraj Mekala, Xinyang Zhang, and Jingbo Shang. 2020. Meta: Metadata-empowered weak supervision for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8351–8361.

Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018. Weakly-supervised neural text classification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 983–992. ACM.

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.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. 2018. Learning to reweight examples for robust deep learning. In *International Conference on Machine Learning*, pages 4334–4343. PMLR.

Ellen Riloff, Janyce Wiebe, and Theresa Wilson. 2003. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 25–32. Association for Computational Linguistics.

Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. 2021. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv preprint arXiv:2101.06329*.

Fangbo Tao, Chao Zhang, Xiusi Chen, Meng Jiang, Tim Hanratty, Lance Kaplan, and Jiawei Han. 2015. Doc2cube: Automated document allocation to text cube via dimension-aware joint embedding. *Dimension*, 2016:2017.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and G. Gordon. 2019. An empirical study of example forgetting during deep neural network learning. *ArXiv*, abs/1812.05159.

Mengting Wan and Julian J. McAuley. 2018. Item recommendation on monotonic behavior chains. In *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*, pages 86–94. ACM.

Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian J. McAuley. 2019. Fine-grained spoiler detection from large-scale review corpora. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2605–2610. Association for Computational Linguistics.

Zihan Wang, Dheeraj Mekala, and Jingbo Shang. 2020. X-class: Text classification with extremely weak supervision. *arXiv preprint arXiv:2010.12794*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. 2019. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pages 7164–7173. PMLR.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2021. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115.

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

A Appendix

The pseudo code for self-training with LOPS is shown in Algorithm 2.

Algorithm 2: Self-training with LOPS label selection

Input: Unlabeled data $\mathcal{D}$, Classifier C, Weak Supervision $\mathcal{W}$.
Output: Prediction labels $predLabs$
$\hat{\mathcal{D}}$ = Generate Pseudo-labels from $\mathcal{D}$, $\mathcal{W}$
for *it* $\in \{1, 2, 3, \ldots, n_{its}\}$ **do**
  $\mathcal{D}_{sel}$ = LOPS ($\hat{\mathcal{D}}$, C)
  Train C on $\mathcal{D}_{sel}$
  $predLabs$, $predProbs$ = Predict(C, $\mathcal{D}$)
  $\hat{\mathcal{D}} = \hat{\mathcal{D}} \cup \{x \mid predProbs(x) > \delta\}$
  it $\leftarrow$ it + 1
Return $predLabs$