Cross-task Knowledge Transfer for Extremely Weakly Supervised Text Classification

Seongmin Park Kyungho Kim Jihwa Lee
ActionPower, Seoul, Republic of Korea
{seongmin.park, kyungho.kim, jihwa.lee}@actionpower.kr

Abstract

Text classification with extremely weak supervision (EWS) imposes stricter supervision constraints compared to regular weakly supervised classification. Absolutely no labeled training samples or hand-crafted rules specific to the evaluation data are allowed. Such restrictions limit state-of-the-art EWS classification methods to indirect weak labeling techniques that assign unnatural label uncertainty estimates. We present PLAT, a framework that creates weak labels by leveraging recent developments in zero-shot text classification. PLAT employs models trained for sub-tasks other than classification to label documents. Most importantly, PLAT refrains from assigning overly confident weak labels and improves soft-label training performance for downstream classifiers. Classifiers trained with PLAT significantly outperform those trained on weak labels generated by the previous state-of-the-art in extremely weakly supervised text classification.

1 Introduction

We undertake the low-resource task of categorizing an unlabeled set of documents using just candidate category labels. The task is a stricter subtask of weakly supervised text classification – weak labels cannot be obtained even through utilizing a small training set or hand-crafted rules based on domain knowledge. Such task formulation mimics a realistic and practical scenario where one has to classify a set of documents into a label from a pre-defined label set, using only class names. Following Wang et al. (2021) we call this task *classification with extremely weak supervision* (EWS).

Due to such additional constraints on sources of supervision, models under EWS cannot trivially adapt recent state-of-the-art approaches under regular weak supervision. Best-performing methods for classification under EWS usually involve mining a set of category-indicative keywords from pre-trained language models (Meng et al., 2018; Mekala and Shang, 2020; Türker et al., 2020; Meng et al., 2020; Zeng et al., 2022). At evaluation time, each document is compared to the keyword set of each label. The weak label for a document is the label with the keyword set most similar to words that constitute the document. This divorcement of feature extraction and label assignment introduces additional noise during weak labeling, causing unnatural assignment of label confidence and oversensitivity to training size (Wang et al., 2021).

We overcome such limitations in EWS by leveraging pre-trained language models to create weak labels in an end-to-end fashion. Most importantly, *we eliminate the keyword-collection step in currently popular EWS approaches*. We employ language models trained on non-classification tasks (textual entailment, next sentence prediction, and multiple-choice question-answering) as weak labelers for classification. Our research bridges weakly-supervised noisy-label training with recent developments in prompt-based low-shot text classification (Yin et al., 2019; Keskar et al., 2019). Our framework realizes both the robustness of noisy-label training and the label efficiency of prompting. We use publicly available, off-the-shelf models for each source task in our experiments.

Our contributions are as follows:

- We analyze the limitations of popular existing methods in EWS, especially in their unnatural assignment of pseudo-label confidence.
- We present PLAT[1], a framework that utilizes models trained in subtasks other than classification to create weak labels for classification. Downstream classifiers trained with our weak labels significantly outperform the previous state-of-the-art in difficult EWS datasets.
- We analyze how cross-task weak labels act as better pseudo-labels, with roots in existing

[1] **P**seudo-**L**abeling **A**cross **T**asks

Findings of the Association for Computational Linguistics: ACL 2023, pages 5329–5341

research in knowledge distillation.

2 Background

2.1 Weakly supervised text classification

Broadly, two lines of research exist in weakly-supervised text classification: obtaining better weak labels (Hancock et al., 2018; Chatterjee et al., 2020a; Rao et al., 2021; Zhang et al., 2021a, 2022a), and streamlining the training of downstream classifiers with the obtained noisy labels (Onoe and Durrett, 2019; Ren et al., 2020; Mekala et al., 2022; Yu et al., 2022; Kuang et al., 2022). PLAT focuses on improving the former by creating weak labels via knowledge transfer from models trained on tasks other than classification.

Weak labels were traditionally assigned using manually written rules (Cachay et al., 2021; Zhang et al., 2022a). Since rule-based labeling necessitates domain knowledge and hand-crafted rules specific to each dataset, much research efforts focused on automatic rule generation. However, even automatically generated rules cannot be used in situations that require EWS because the process either necessitates a small labeled dataset of the same classification task (Varma and Ré, 2018; Banerjee et al., 2019; Sukumaran et al., 2022), or human feedback is required in the iterative learning process (Zhang et al., 2022b). Under EWS, we require a method that fully automates the weak-labeling process, without any classification datasets or dataset-specific domain knowledge.

PLAT employs cross-task knowledge transfer to achieve completely automated weak-labeling without any labeled classification data.

2.2 Cross-task knowledge transfer

In cross-task knowledge transfer, a model trained for a specific source task solves a different target task. Cross-task knowledge transfer is useful when labeled training data is scarce for the target task but is abundant or unnecessary for the source task (Egonmwan et al., 2019; Lin et al., 2021). Such preconditions make cross-task knowledge transfer naturally suitable for pseudo-labeling in weakly supervised training. In Egonmwan et al. (2019), for instance, question-answering models are used to create weak summary labels.

Because data efficacy of weak labelers is a prerequisite under weak supervision, recently popular zero-shot classification methods based on prompting (Brown et al., 2020; Liu et al., 2021; Sanh et al., 2022) are appealing approaches to automatic label creation. In the EWS setup, only cross-task zero-shot labelers can be used, because the task prohibits any labeled data for classification. Although cross-task knowledge transfer with a classification target task is extensively researched (Hancock et al., 2018; Wang et al., 2019; Khodorchenko, 2019; Rao et al., 2021; Chatterjee et al., 2020a), we are the first to explore prompt-based, cross-task distillation for weak-labeling in text classification.

Zhang et al. (2021a) also uses language model prompting for weak label generation, but its weak-labeler is a text classification model and thus is not a cross-task setup. In concurrent work, Smith et al. (2022) also prompts language models for zero-shot weak label generation. The research leverages multi-task models trained with multiple source tasks, either with extremely large scale (GPT-3) or already on text classification source tasks (T0++). In contrast, our work focuses on cross-task knowledge distillation capabilities of data-efficient, single-task models, each trained for a different non-classification source task. Smith et al. (2022) focuses on zero-shot capabilities that emerge from extreme-scale text generation models, while our work explores methods to handle various model output types (open-ended, binary, and multiple-choice) for weak labeling. We further provide qualitative analysis on the confidence assigned to each weak label.

2.3 Common approaches to text classification under EWS

Popular methods under the EWS constraint employ a keyword-set matching scheme for weak labeling. Keywords for each label are auto-generated by mining pre-trained language models. Throughout this paper, we call these methods *keyword-based EWS*.

WeSTClass (Meng et al., 2018) augments training data by creating pseudo-documents from seed words for each class. ConWea (Mekala and Shang, 2020) uses masked language models to discern overlapping keywords with context. Context-infused keyword set for each class is matched with documents for weak labeling. WESSTEC (Türker et al., 2020) queries a knowledge base for information about each label and calculates the similarity between a document's vector representation to each label knowledge embedding. LOTClass (Meng et al., 2020) enriches each label's keyword set by collecting possible replacement words for

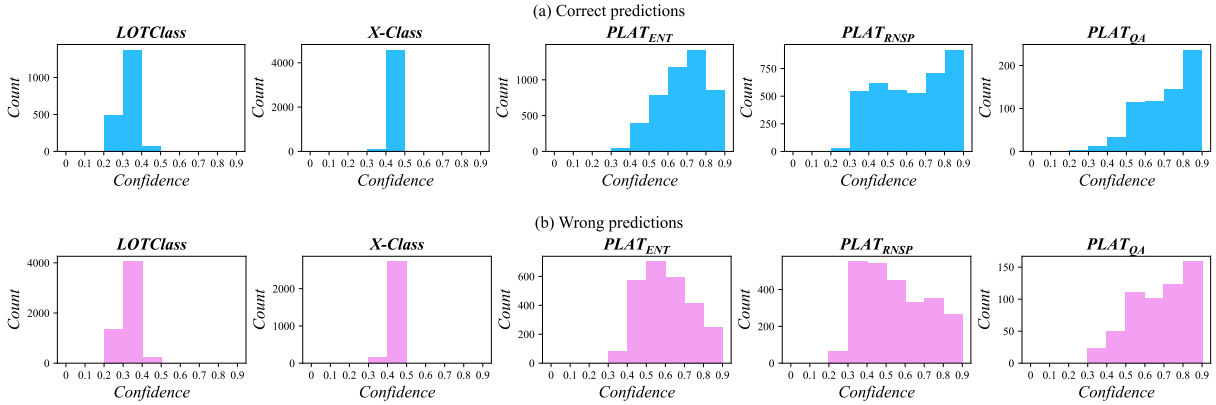

Figure 1: Weak label confidence assigned to correct and wrong predictions on AGNews. Soft label training cannot take advantage of drastic confidence distributions from keyword-based EWS (LOTClass and X-Class). Precursory analysis of weak label confidence distributions helps in avoiding overconfident weak labelers.

every label from masked language models. X-Class (Wang et al., 2021) force-aligns document representations to label embeddings for weak labeling and achieves state-of-the-art results in EWS. All aforementioned methods train a downstream classifier such as BERT (Devlin et al., 2019) or RoBERTa (Liu et al., 2019) as the second step in their pipelines. In succeeding work, ClassKG (Zhang et al., 2021b) posits EWS as a keyword-subgraph annotation task and takes keyword correlation into account.

PLAT departs from existing conventions by eliminating the keyword collection process from the EWS pipeline. We directly mine weak predictions instead of keywords from trained language models.

In a similar and concurrent work as ours, WDDC (Zeng et al., 2022) also uses zero-shot prompting in pre-trained language models to create weak labels. WDDC uses cloze prompts to extract keyword sets (to be compared against document words, as in most aforementioned works), which is significantly different from PLAT that *assigns classification labels directly with the prompts*.

We choose LOTClass and X-Class as baselines in our experiments, for their state-of-the-art results and reproducibility.

3 Problems with keyword-matching EWS

Compared to simple supervised classification, EWS methods mentioned in Section 2.3 introduce two additional steps to the training pipeline: building category-indicative keyword sets and assigning weak labels to unlabeled documents using the built keyword sets. Our investigations show that the disjoint nature of such approaches leads to unwanted side effects that degrade weak-label quality.

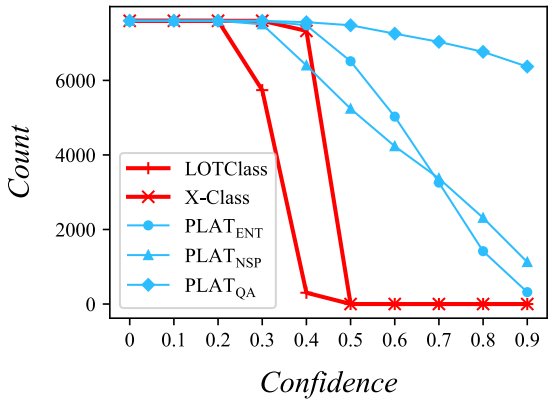

Figure 2: Number of weak labels on AGNews above each label confidence threshold. In keyword-based EWS, it is not straightforward to choose a confidence cut-off for downstream classifier training.

3.1 Uninformative label confidence

Accurate estimates of label uncertainty are important in noisy training scenarios commonly used in weakly-supervised classification (Meng et al., 2020; Yuan et al., 2020). The keyword-matching process used in state-of-the-art EWS forces weak labelers to gauge weak label confidence indirectly. We find that weak labels obtained this way are often coupled with unreliable label confidence that is sensitive to hyperparameters such as the size of the keyword set or evaluation set. In X-Class, measuring the distance of a document's embedding from its label cluster center is the only way to measure label uncertainty. In LOTClass, prediction confidence is the number of keywords a document contains in its pseudo-label keyword set.

Even though PLAT uses non-classification models for weak labeling, much more natural confidence is assigned to its weak labels. Label confidence of correct predictions are higher on average compared to those of wrong predictions (Figure 1). In contrast, LOTClass and X-Class assigns similar confidence to both. Weak labels created by LOTClass and X-Class show drastic drops in pseudo-label count as the label confidence threshold increases (Figure 2). Such overconfidence in label quality estimates can hinder downstream classifier performance (Wei et al., 2022; Jiang et al., 2021).

3.2 Inability to handle complex class names

Keyword-based EWS relies on mining words within documents in the evaluation set and extracting category-indicative words for each class. Therefore, even state-of-the-art methods require class names to be either lexically or contextually descriptive. In clickbait classification, for example, the word "clickbait" does not exist among news headlines the model has to classify. In such cases, keyword-based EWS methods have no anchor within the documents to extract category-indicative keywords for the word "clickbait". Wang et al. (2021) shows existing EWS methods falter when the label names do not appear in documents to be weakly labeled.

We observe the same phenomenon even with keyword-based EWS methods that consider language context. Robustness further deteriorates when label names are more complex, such as consisting of multiple tokens. Most keyword-based EWS use masked language models for context-aware keyword search, and it is not straightforward to consolidate sequence vectors as a single, contextual vector to be used in clustering algorithms in keyword-based EWS. Our method sidesteps such limitations by flexibly handling any label name through language model prompting.

3.3 Sensitivity to dataset size

Weak-label quality of keyword-based EWS also relies heavily on the size of the test set. While EWS methods require no labeled documents for training, existing methods still require a sizable count of unlabeled data to perform well. At their core, most keyword-based EWS methods aim to generate cluster-centers for each class by leveraging textual information in unlabeled documents. A smaller evaluation set will naturally result in lower-quality cluster boundaries. To overcome such reliance on dataset size, we propose a way to weakly label each document in a zero-shot manner. We confirm this intuition by comparing F_1 scores of weak labels created by keyword-based EWS (X-Class) and with PLAT (Figure 3) at varying confidence thresholds.

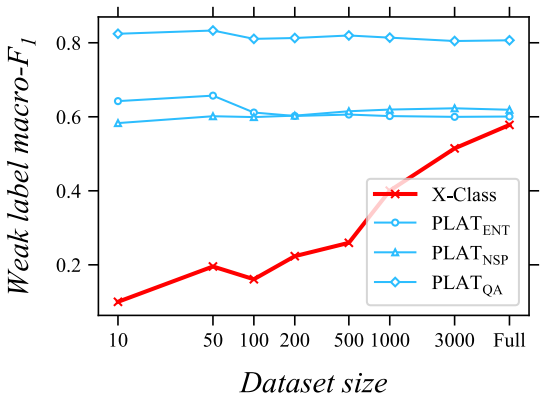

Figure 3: Macro-F_1 of weak labels on AGNews while varying dataset size. Zero-shot capabilities of PLAT make the framework robust to data count.

4 PLAT

PLAT draws inspiration from recent findings in zero-shot language model prompting (Yin et al., 2019; Keskar et al., 2019; Ma et al., 2021) to obtain weak labels for unlabeled documents. In the EWS setting, PLAT leverages source models trained on a single non-classification task to solve classification tasks.

PLAT follows the typical two-step weakly-supervised classification pipeline. In the first phase, PLAT creates weak labels for classification. In the second phase, a final classifier is trained using the obtained weak labels. The novelty of PLAT lies in improving the weak labeling phase with cross-task knowledge distillation. We aim to keep the PLAT framework source-model-agnostic and make the weak labelers hot-swappable, taking advantage of parallel advances in zero-shot NLP.

4.1 Cross-task weak labeling

We test three different models for weak labeling, each trained on a single task: entailment (PLAT$_{ENT}$), next sentence prediction (PLAT$_{NSP}$), and multiple-choice question answering (PLAT$_{QA}$). Although each task and corresponding model have different input and output formats, adding appropriate prompts can reduce all tasks into indirect classification tasks.

Dataset	Type	# of Classes	Dataset size	Average word count per sample
AGNews	News topic	4	7,600	37.72
Yahoo	News topic	10	60,000	10.70
DBpedia	Article topic	14	70,000	46.14
Clickbait	Clickbait detection	2	16,000	9.09

Table 1: Datasets to benchmark PLAT against keyword-based EWS.

Let $X = \{x_0, \ldots, x_m\}$ be a set of unlabeled documents to be classified. A weak labeler must categorize x_i as a single class from the set of all possible classes $C = \{c_0, ..., c_n\}$.

For every $x_i \in X$, PLAT's pseudo-labelers generate two kinds of labels: a hard label $H_i \in C$, which is the single most likely class that x_i belongs to, and a soft label S_i, which is a categorical distribution over C expressing the class probability of x_i. We train separate downstream classifiers using hard and soft labels, to compare how taking label uncertainty into account can help in weakly-supervised classification.

4.1.1 Entailment (PLAT_{ENT})

Yin et al. (2019) explores zero-shot text classification through entailment. Similarly, we pose classification as an entailment task by ranking entailment probabilities of x_i and each verbalized class. A verbalized class (Schick and Schütze, 2021) is the every class name in the form of a sentence, adapted to appear as an input to the entailment model.

Verbalizers can be adapted for each classification task. For topic classification, the verbalizer could be *"This text is about <class name>."* For spam detection, the verbalizer to represent the spam class could be *"This is an ad"*. The full set of verbalizers used is detailed in the Experiments section.

V_{ENT} is a set of all verbalized class names:

$$V_{ENT} = \{verbalizer(c) \mid c \in C\}. \quad (1)$$

For every $x_i \in X$, we construct a set of all pairs of x_i and each verbalized label $v \in V_{ENT}$, between all of which we calculate textual entailment:

$$Pairs^i_{ENT} = \{(x_i, v) \mid v \in V_{ENT}\}. \quad (2)$$

Entailment model M_{ENT} is a model that takes a sentence pair (s_1, s_2) and calculates the probabilities that sentence s_1 entails, contradicts, or has no relation to sentence s_2. In this work, we only use entailment probabilities.

We use M_{ENT} to calculate the entailment probability of every $(x_i, v) \in Pairs^i_{ENT}$.

$$Probs^i_{ENT} = \{M_{ENT}(x_i, v) \mid (x_i, v) \in Pairs^i_{ENT}\}. \quad (3)$$

The hard label H_i for x_i is $argmax(Probs^i_{ENT})$, and the soft label S_i is $softmax(Probs^i_{ENT})$.

4.1.2 Next sentence prediction (PLAT_{NSP})

Ma et al. (2021) finds that next sentence prediction (NSP) and reverse NSP models perform on par with entailment in zero-shot text classification. In our experiments, NSP and reverse NSP weak labelers had a negligible difference in final classifier performance. We choose reverse NSP for higher reported classification scores in Ma et al. (2021).

We use the same verbalizers ($V_{NSP} = V_{ENT}$) as in entailment-based weak labeling in the preceding section. For every $x_i \in X$, we construct a set of all pairs of x_i and each verbalized label $v \in V_{NSP}$, similar to $Pairs^i_{ENT}$ in 4.1.1:

$$Pairs^i_{NSP} = \{(x_i, v) \mid v \in V_{NSP}\}. \quad (4)$$

For all $v \in V$, we calculate probabilities that x_i appears after each v. NSP Model M_{NSP} takes a sentence pair (s_1, s_2) and calculates the probability that s_2 appears after s_1.

$$Probs^i_{NSP} = \{M_{NSP}(v, x_i) \mid (x_i, v) \in Pairs^i_{NSP}\}. \quad (5)$$

The hard label H_i for x_i is $argmax(Probs^i_{NSP})$, and the soft label S_i is $softmax(Probs^i_{NSP})$.

4.1.3 Multiple-choice question-answering (PLAT_{QA})

A multiple-choice question-answering (QA) model M_{QA} takes a context, a question, and answer choices, and returns the distribution of answer possibility over the answer choices. To pose QA as a classification task, we set the context as each x_i,

Model	AGNews	Yahoo	DBpedia	Clickbait
Supervised	93.97 / 93.97	72.11 / 72.64	99.11 / 99.11	98.57 / 98.58
LOTClass				
Hard label	25.63 / 19.47	9.93 / 5.37	6.89 / 6.29	44.17 / 43.18
Final classifier	25.00 / 10.00	10.00 / 1.82	0.80 / 0.17	50.00 / 33.33
X-Class				
Hard label	61.82 / 57.81	40.35 / 42.53	88.17 / 87.91	23.79 / 23.72
Final classifier	62.87 / 58.81	41.76 / 43.86	**88.52 / 88.21**	21.22 / 20.81
PLAT_{ENT}				
Hard label	64.88 / 60.07	54.17 / 54.91	81.78 / 80.84	51.35 / 36.48
Final classifier	64.86 / 57.73	**55.29 / 56.45**	82.62 / 81.43	50.00 / 33.33
PLAT_{NSP}				
Hard label	64.79 / 61.90	49.45 / 47.39	39.90 / 44.18	77.23 / 77.03
Final classifier	60.87 / 56.63	52.46 / 50.04	41.32 / 44.97	79.68 / 79.50
PLAT_{QA}				
Hard label	80.86 / 80.67	41.44 / 44.59	83.32 / 82.54	83.87 / 83.80
Final classifier	**81.72 / 81.57**	43.83 / 46.88	84.91 / 84.00	**87.44 / 87.40**

Table 2: Classification performance of weak labels (*"Hard label"*) and final classifier trained with the weak labels (*"Final classifier"*). All reported scores are in the form *micro-*F_1 */ macro-*F_1. PLAT outperforms baselines in all datasets except in DBpedia, a dataset in which keyword-based EWS methods have the most opportunity to mine keyword sets.

the question as a dataset-specific prompt p, and the answer choices as verbalized versions of all classes. The question forces the model to select one verbalized element of C as an answer.

The full set of prompts and verbalizers used in PLAT_{QA} is detailed in the experiments section. Even though prompt p is dataset-specific, its construction does not require domain knowledge, and instead depends on the *type* of classification (I.e. topic classification, location classification, etc.).

Formally defined,

$$Probs^i_{QA} = M_{QA}(x_i, p, V_{QA}), \tag{6}$$

where

$$V_{QA} = \{verbalizer(c) \mid c \in C\}. \tag{7}$$

The hard label H_i for x_i is $argmax(Probs^i_{QA})$ and the soft label S_i is $softmax(Probs^i_{QA})$.

4.2 Final classifier training

A separate text classifier is trained with obtained weak labels. We use BERT in all our experiments. The final classifier is the output model of PLAT.

4.2.1 Training with hard labels

Given a set of hard labels $\{H_0, ..., H_i\}$ created by a weak-label generator, we train a downstream classifier B by maximizing the likelihood of predicting the weak label given the document. The loss function is a standard cross-entropy objective:

$$\mathcal{L}_{hard} = -\sum_{i=0}^{m}\sum_{j\in C} y(H_i)\log(B(x_i)_j), \tag{8}$$

where $y(H_i)$ is 1 only if $j = H_i$ and 0 otherwise. $B(x_i)_j$ is the prediction confidence of classifier for class $c_j \in C$ on document x_i.

4.2.2 Training with soft labels

We adopt a similar objective function when training with confidence-aware soft labels. The final classifier is trained to minimize the divergence between the one-hot model prediction and the soft confidence distribution from the weak labeler over the set of all possible class names. The classifier's objective function becomes:

$$\mathcal{L}_{soft} = -\sum_{i=0}^{m}\sum_{j\in C} S^j_i\log(B(x_i)_j), \tag{9}$$

where S^j_i is the weak label confidence of specific class $j \in C$ from overall confidence distribution assigned to x_i.

5 Experiments and Results

We qualitatively analyze PLAT in two aspects: accuracy of generated weak labels, and prediction

Model	AGNews	Yahoo	DBpedia	Clickbait
Supervised	93.97 / 93.97	72.11 / 72.64	99.11 / 99.11	98.57 / 98.58
LOTClass	25.00 (+0.00) / 10.00 (+0.00)	10.00 (+0.00) / 1.82 (+0.00)	3.39 (+2.59) / 0.54 (+0.37)	50.00 (+0.00) / 33.33 (+0.00)
X-Class	60.61 (-2.26) / 52.84 (-5.96)	41.07 (-0.69) / 43.16 (-0.70)	**88.70** (+0.18) / **88.38** (+0.17)	20.38 (-0.84) / 20.20 (-0.61)
PLAT_{ENT}	66.41 (+1.55) / 60.01 (+2.28)	**55.39** (+0.10) / **56.20** (-0.24)	82.99 (+0.36) / 82.16 (+0.73)	50.59 (+0.59) / 34.77 (+1.44)
PLAT_{NSP}	67.68 (+6.82) / 64.19 (+7.56)	52.30 (-0.16) / 50.34 (+0.30)	40.20 (-1.12) / 43.23 (-1.74)	81.22 (+1.54) / 81.07 (+1.57)
PLAT_{QA}	**81.99** (+0.26) / **81.83** (+0.26)	43.32 (-0.50) / 46.53 (-0.35)	86.38 (+1.46) / 85.74 (+1.74)	**88.38** (+0.94) / **88.37** (+0.97)

Table 3: Final classifier performance on 4 classification benchmarks after training with soft labels. Numbers in parenthesis indicate absolute increase in F_1 scores compared to hard label results in Table 2.

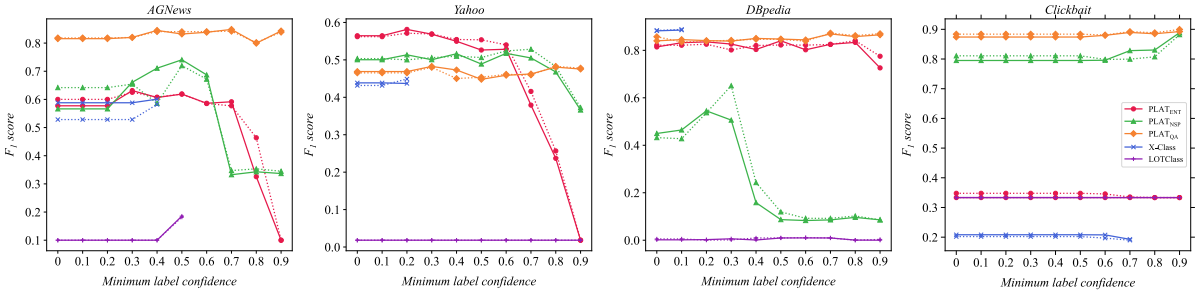

Figure 4: Final classifier performance after training with only labels above certain confidence threshold. We observe a trade-off between weak label count and minimum label confidence. Dotted lines indicate soft-label training.

classification performance of the final classifier trained with the weak labels. For the latter, we measure classifier performance in both hard- and soft-label (confidence-aware) training. The same configuration for training the final classifier is applied to all variants of PLAT and baseline weak labelers. Classifier performance is measured in macro- and micro-F_1 scores.

5.1 Datasets

We test PLAT on topic and clickbait classification datasets. For topic classification, we use AGNews (Zhang et al., 2015), Yahoo Topics (Zhang et al., 2015), and DBpedia (Zhang et al., 2015). We use Clickbait Detection (Chakraborty et al., 2016) for clickbait classification. Table 1 provides a detailed description of each benchmark dataset.

For topic classification datasets, we use the verbalizer *"This text is about <class name>"* for all models and the prompt *"What is this text about?"* for PLAT_{QA}. For clickbait classification, we use the verbalizers *"This is <news/spam>"* and the prompt *"Is this news or spam?"* for PLAT_{QA}.

5.2 Source models for weak-labeling

We use publicly available cross-task labelers in all variants of PLAT. For PLAT_{ENT}, we use BART[2] (Lewis et al., 2020) trained on MNLI (Williams et al., 2018). For PLAT_{NSP}, we use BERT[3] trained with standard token unmasking and NSP objectives. For PLAT_{QA}, we use RoBERTa[4] model trained on RACE (Lai et al., 2017).

To train the final classifier with weak labels generated by aforementioned models, we fine-tune a pre-trained BERT model[5] with a constant learning rate of $5e^{-5}$. We use an AdamW optimizer (Loshchilov and Hutter, 2018) with $\beta_1 = 0.9, \beta_2 = 0.999, eps = 1e^{-6}$, and no weight decay.

5.3 Hard label training results

Classification performance of the final classifier for baseline weak-labelers and variants of PLAT is detailed in Table 2. Weak labels generated with PLAT yield notably higher F_1 scores compared to those generated by baselines, except on DBpedia. The high performance of X-Class on DBpedia can be attributed to longer average document length and greater test set size. Compared to other datasets, DBpedia provides a greater amount of raw text from which keyword-based baselines can mine category-indicative keywords. In such settings, PLAT's zero-shot capability does not provide an advantage as great in scenarios with fewer resources.

[2]https://huggingface.co/facebook/bart-large-mnli
[3]https://huggingface.co/bert-large-cased
[4]https://huggingface.co/LIAMF-USP/roberta-large-finetuned-race
[5]https://huggingface.co/bert-base-cased

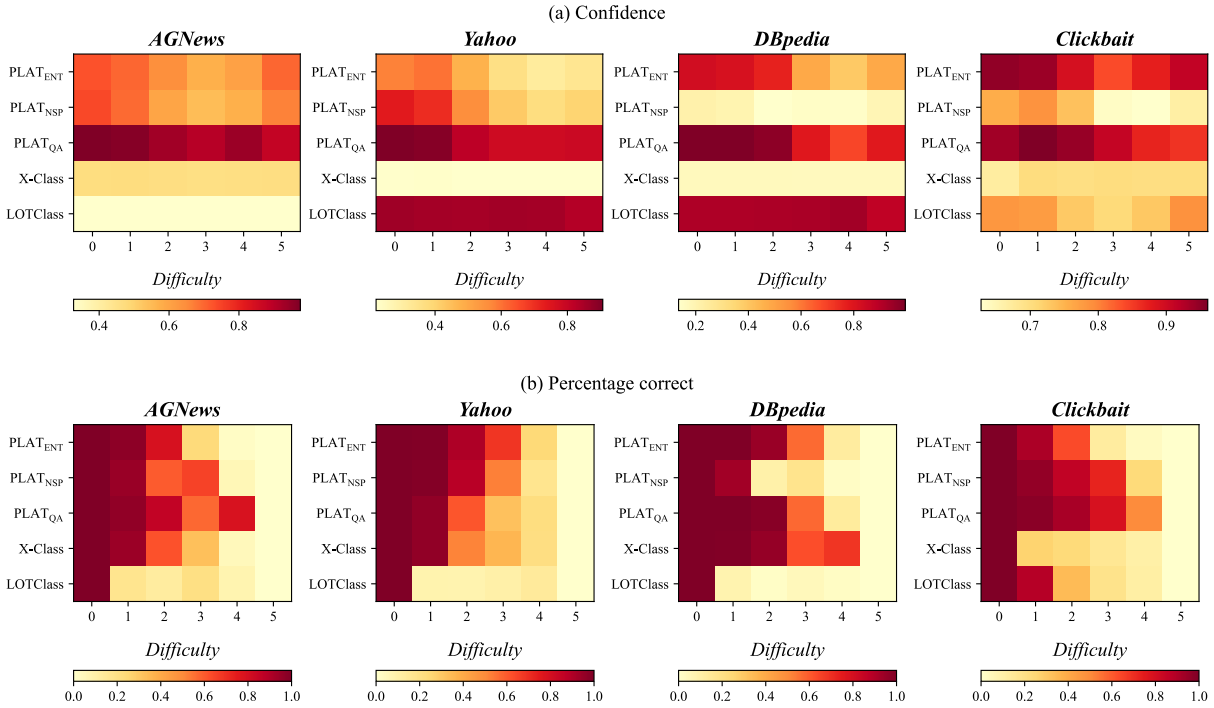

Figure 5: Confidence distribution and accuracy of pseudo labels according to question difficulty. Weak labelers with more granular label confidence tend to produce more accurate weak labels.

5.4 Confidence-aware training results

Classification performance after confidence-aware training is detailed in Table 3. PLAT also notably outperforms baselines with soft labels, taking better advantage of confidence-aware training. Our results confirm past research in knowledge distillation that accurate estimates of label uncertainty lead to better model calibration (Chatterjee et al., 2020b; Rizve et al., 2021).

Earlier works in EWS try retaining only labels with confidence over a certain threshold δ. In a noisy-training scenario, a trade-off exists between retaining a large number of training examples and average label confidence. Our work confirms findings in Wang et al. (2021) that excessively high label δ degrades final classifier performance (Figure 4). While tuning the threshold parameter results in a higher increase in F_1 scores for PLAT, we report scores at $\delta = 0$ for a fair comparison with previous work and to eliminate δ as a hyperparameter.

5.5 Classification difficulty analysis

We analyze how the performance of each weak labeler changes according to the classification difficulty of each sample. Classification difficulty of a sample is defined as the number of weak labelers that made wrong predictions. Since we compare 5 models, the maximum difficulty is 5. PLAT assigns a much more "natural" confidence distribution, where the model is confident about low-difficulty questions while comparatively uncertain about high-difficulty questions (Figure 5). Models that fail to show such graduality tend to make inaccurate predictions (LOTClass in AGNews and Yahoo, X-Class in Clickbait, and PLAT_{NSP} in DBpedia), especially on more difficult samples.

6 Conclusion

We present three variants of PLAT, a framework for text classification under extremely weak supervision. By eliminating keyword-based weak labeling, PLAT sidesteps the brittle dependence on evaluation set size and hyperparameters found in previous state-of-the art methods. PLAT is a flexible framework that leverages prompting to generate weak labels with more natural confidence estimates.

PLAT makes no assumptions about the training dynamics of its source models. Therefore, evolutions of source models are completely orthogonal to developments in PLAT. The black-box treatment of its weak labeler models enables the usage of completely unsupervised weak labelers – a potential already demonstrated by PLAT_{NSP}. We expect future developments in unsupervised solutions to enable even more resource-efficient classification under PLAT.

Limitations

We identify the following limitations of PLAT and strategies to overcome such drawbacks:

- *Performance of the final classifier is dependent on the black-box source weak labeler.* We believe this limitation can be worked around in a real-word setting by ensembling source models to vote on a likely weak label for practical accuracy gains.
- *Best-performing source models might differ for different tasks.* The dataless nature of EWS prevents precursory accuracy evaluations while choosing the source weak labeler model. However, quality of candidate weak labelers can be gauged indirectly. Users can examine confidence distributions of weak labels (as in Figure 1 and Figure 2) as an indicator of pseudo-label "naturalness". They can also perform difficulty analysis (as shown in Figure 5(a)) that does not require any labeled data. In a real-world scenario, ensemble weak labelers will be used, eliminating the need to choose a single best source model.

References

Siddhartha Banerjee, Cem Akkaya, Francisco Perez-Sorrosal, and Kostas Tsioutsiouliklis. 2019. Hierarchical transfer learning for multi-label text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6295–6300, Florence, Italy. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Salva Rühling Cachay, Benedikt Boecking, and Artur Dubrawski. 2021. End-to-end weak supervision. In *Advances in Neural Information Processing Systems*.

Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop clickbait: Detecting and preventing clickbaits in online news media. In *2016 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)*, pages 9–16. IEEE.

Oishik Chatterjee, Ganesh Ramakrishnan, and Sunita Sarawagi. 2020a. Robust data programming with precision-guided labeling functions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3397–3404.

Oishik Chatterjee, Ganesh Ramakrishnan, and Sunita Sarawagi. 2020b. Robust data programming with precision-guided labeling functions. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 3397–3404. AAAI Press.

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.

Elozino Egonmwan, Vittorio Castelli, and Md Arafat Sultan. 2019. Cross-task knowledge transfer for query-based text summarization. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 72–77, Hong Kong, China. Association for Computational Linguistics.

Braden Hancock, Paroma Varma, Stephanie Wang, Martin Bringmann, Percy Liang, and Christopher Ré. 2018. Training classifiers with natural language explanations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1884–1895, Melbourne, Australia. Association for Computational Linguistics.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Unifying question answering, text classification, and regression via span extraction. *arXiv preprint arXiv:1904.09286*.

Maria Khodorchenko. 2019. Distant supervision and knowledge transfer for domain-oriented text classification in online social networks. *Procedia Computer Science*, 156:166–175.

Zhaobin Kuang, Chidubem G. Arachie, Bangyong Liang, Pradyumna Narayana, Giulia Desalvo, Michael S. Quinn, Bert Huang, Geoffrey Downs, and Yang Yang. 2022. Firebolt: Weak supervision under weaker assumptions. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 8214–8259. PMLR.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Zhaojiang Lin, Bing Liu, Andrea Madotto, Seungwhan Moon, Zhenpeng Zhou, Paul Crook, Zhiguang Wang, Zhou Yu, Eunjoon Cho, Rajen Subba, and Pascale Fung. 2021. Zero-shot dialogue state tracking via cross-task transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7890–7900, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

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*, abs/1907.11692.

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

Tingting Ma, Jin-Ge Yao, Chin-Yew Lin, and Tiejun Zhao. 2021. Issues with entailment-based zero-shot text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 786–796, Online. Association for Computational Linguistics.

Dheeraj Mekala, Chengyu Dong, and Jingbo Shang. 2022. Lops: Learning order inspired pseudo-label selection for weakly supervised text classification. *arXiv preprint arXiv:2205.12528*.

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, Online. Association for Computational Linguistics.

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*, CIKM '18, page 983–992, New York, NY, USA. Association for Computing Machinery.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. 2020. Text classification using label names only: A language model self-training approach. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9006–9017, Online. Association for Computational Linguistics.

Yasumasa Onoe and Greg Durrett. 2019. Learning to denoise distantly-labeled data for entity typing. 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 2407–2417, Minneapolis, Minnesota. Association for Computational Linguistics.

Nikitha Rao, Chetan Bansal, and Joe Guan. 2021. Search4code: Code search intent classification using weak supervision. In *2021 IEEE/ACM 18th International Conference on Mining Software Repositories (MSR)*, pages 575–579. IEEE.

Wendi Ren, Yinghao Li, Hanting Su, David Kartchner, Cassie Mitchell, and Chao Zhang. 2020. Denoising multi-source weak supervision for neural text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3739–3754, Online. 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. In *International Conference on Learning Representations*.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Ryan Smith, Jason A Fries, Braden Hancock, and Stephen H Bach. 2022. Language models in the loop: Incorporating prompting into weak supervision. *arXiv preprint arXiv:2205.02318*.

Rohan Sukumaran, Sumanth Prabhu, and Hemant Misra. 2022. Enhanced text classification using proxy labels

and knowledge distillation. In *5th Joint International Conference on Data Science Management of Data (9th ACM IKDD CODS and 27th COMAD)*, CODS-COMAD 2022, page 227–230, New York, NY, USA. Association for Computing Machinery.

Rima Türker, Lei Zhang, Mehwish Alam, and Harald Sack. 2020. Weakly supervised short text categorization using world knowledge. In *The Semantic Web – ISWC 2020: 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part I*, page 584–600, Berlin, Heidelberg. Springer-Verlag.

Paroma Varma and Christopher Ré. 2018. Snuba: Automating weak supervision to label training data. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, volume 12, page 223. NIH Public Access.

Yanshan Wang, Sunghwan Sohn, Sijia Liu, Feichen Shen, Liwei Wang, Elizabeth J Atkinson, Shreyasee Amin, and Hongfang Liu. 2019. A clinical text classification paradigm using weak supervision and deep representation. *BMC medical informatics and decision making*, 19(1):1–13.

Zihan Wang, Dheeraj Mekala, and Jingbo Shang. 2021. X-class: Text classification with extremely weak supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3043–3053, Online. Association for Computational Linguistics.

Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. 2022. Mitigating neural network overconfidence with logit normalization.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3914–3923, Hong Kong, China. Association for Computational Linguistics.

Peilin Yu, Tiffany Ding, and Stephen H Bach. 2022. Learning from multiple noisy partial labelers. In *International Conference on Artificial Intelligence and Statistics*, pages 11072–11095. PMLR.

Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. 2020. Revisiting knowledge distillation via label smoothing regularization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3902–3910.

Ziqian Zeng, Weimin Ni, Tianqing Fang, Xiang Li, Xinran Zhao, and Yangqiu Song. 2022. Weakly supervised text classification using supervision signals from a language model. *arXiv preprint arXiv:2205.06604*.

Jieyu Zhang, Cheng-Yu Hsieh, Yue Yu, Chao Zhang, and Alexander Ratner. 2022a. A survey on programmatic weak supervision. *arXiv preprint arXiv:2202.05433*.

Jieyu Zhang, Bohan Wang, Xiangchen Song, Yujing Wang, Yaming Yang, Jing Bai, and Alexander Ratner. 2021a. Creating training sets via weak indirect supervision. *arXiv preprint arXiv:2110.03484*.

Lu Zhang, Jiandong Ding, Yi Xu, Yingyao Liu, and Shuigeng Zhou. 2021b. Weakly-supervised text classification based on keyword graph. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2803–2813, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Rongzhi Zhang, Yue Yu, Pranav Shetty, Le Song, and Chao Zhang. 2022b. Prompt-based rule discovery and boosting for interactive weakly-supervised learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 745–758, Dublin, Ireland. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Last section

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
Not applicable. Left blank.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☑ Did you run computational experiments?

5

☒ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Low resource study; computing infrastructure was almost all-encompassing.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Left blank.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Left blank.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Detailed in footnotes

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.