Partially Humanizing Weak Supervision: Towards a Better Low Resource Pipeline for Spoken Language Understanding

Ayush Kumar[†], **Rishabh Kumar Tripathi**[†][*], **Jithendra Vepa**
Observe.AI, India
{ayush,rishabh.tripathi,jithendra}@observe.ai

Abstract

Weak Supervised Learning (WSL) is a popular paradigm to develop machine learning models in absence of labeled training data. WSL involves training over noisy labels which are traditionally obtained from hand-engineered semantic rules and task-specific pre-trained models. Such rules offer limited coverage and generalization over tasks. On the other hand, pre-trained models are available only for limited tasks. Thus, obtaining weak labels is a bottleneck in weak supervised learning. In this work, we propose to utilize the prompting paradigm to generate weak labels for the underlying tasks. We show that task-agnostic prompts are generalizable and can be used to obtain noisy labels for different Spoken Language Understanding (SLU) tasks such as sentiment classification, disfluency detection and emotion classification. These prompts can additionally be updated with '*human-in-the-loop*' to add task-specific contexts. Our proposed WSL pipeline outperforms other competitive low-resource benchmarks on zero and few-shot learning by more than 4% on Macro-F1 and a conventional rule-based WSL baseline by more than 5% across all the benchmark datasets. We demonstrate that prompt-based method helps to generate more reliable labels for the above SLU tasks in less than 72% of time compared to a traditional rule-based method to obtain noisy labels and thus can be used as a universal strategy to obtain weak labels in a weak-supervised framework.

1 Introduction

Weak supervised learning (WSL) (Yu et al., 2021; Ren et al., 2020) has gained interest in the research community because of the success shown by leveraging the high availability of large volumes of unlabeled data (Zhou, 2018). In these weak supervision setups, the unlabeled samples are pseudo-annotated by noisy labels and a noise correction strategy is applied to train the model over such labels. These noisy labels are derived using source(s), commonly known as weak source(s). Most common forms of weak sources observed in the area of weak supervision are: rule-based weak sources (Hutto and Gilbert, 2014; Nielsen, 2011) and task-specific fine-tuned models (Schweter and Akbik, 2020).

Rule-based weak sources require designing the labeling functions using the heuristics, lexicons and external knowledge bases (like *SentiWordNet* (Esuli and Sebastiani, 2006)) to map an input to the class labels expected in the task. It is a challenge to extend such rules to dataset from different domains or to perform a different task. Additionally, designing rules for an individual dataset is a manually time intensive task. While, there are a few rule-based sources available for common tasks like sentiment, it is hard to find such readily available weak sources for tasks like disfluency detection (Godfrey et al., 1992).

The other type of weak-sources utilize a task specific fine-tuned language model. For example, BERT-NER, which is BERT (Devlin et al., 2019) fine-tuned for NER task, can be utilized as a weak source to identify named entities from the data. Such task-specific models cannot be used to predict class labels for a task different than what the model is trained on.

The common challenge imposed by both of these weak sources is the lack of generalizability across a wide variety of tasks. In this paper, we propose a universal weak source which not only generates better quality noisy class labels for wide range of tasks, but can also be tweaked to write '*human-in-the-loop*' task-specific details with minimal efforts. We present *prompt-based weak source*, a hybrid source which utilizes a pre-trained language model (PLM) as a knowledge base and limited human intervention as a prompter to address the labeling problems observed in traditionally used weak sources. A prompt-based weak source requires prompting a

[*]Work done during internship at Observe.AI, [†]Equal contribution

Proceedings of the Fourth Workshop on Data Science with Human-in-the-Loop (Language Advances), pages 64 - 73
December 8, 2022 ©2022 Association for Computational Linguistics

PLM (Gao et al., 2021; Schick and Schütze, 2021; Logan IV et al., 2021) to derive weak class labels. A prompt refers to a pattern string that is designed to coax the model into producing an output corresponding to a given class (Scao and Rush, 2021). We study different ways to prompt PLMs in 3.1.1 and 3.1.2. We study 3 key features of prompt-based sources as: *Generalizability* (utilize task-agnostic prompts to cater to various tasks and effectively create prompts for multiple tasks within same generic framework.), *Flexibility* (to modify the prompts to add task and class-label specific contexts in an easy manner to improve over task-agnostic prompts), *Potency* (to derive weak labels with reliable source performance).

Our major contributions in this work are:

- Instead of the classical application of prompting in a few-shot and zero-shot settings, we propose utilization of prompting paradigm to generate noisy labels needed in WSL.
- We demonstrate a generalizable, flexible and time-efficient low-resource '*human-in-the-loop*' setup to train a weak supervised model using task-agnostic and task-specific prompt-based weak sources.
- We perform extensive experiments on three benchmark SLU datasets and demonstrate the effectiveness of the proposed *'human-in-the-loop'* in reducing manual overhead along with improving the performance over the traditional rule-intensive weak sources and other competitive low resource setups.

2 Related Work

Existing works (Wang et al., 2019; Hedderich and Klakow, 2018) in WSL learns on a few gold data, while another group of work (Yu et al., 2021; Ren et al., 2020; Ratner et al., 2020) assumes that no labeled data is available. In the scope of our work, we explore approaches that do not rely on labeled data to train a weak supervised model (WSM). Ren et al. (2020) utilized BERT to learn conditional reliability scores between multiple weak sources using an attention mechanism, while Ratner et al. (2020) proposed a generative model to combine outputs from various weak sources. Yu et al. (2021) proposed a contrastive self-training strategy to learn over weak labels and outperformed prior works (Ren et al., 2020; Ratner et al., 2020). Hence, our work borrows ideas from Yu et al. (2021) to train a weak supervised model (WSM) considering its robustness towards high intensities of label noise.

Prompt-based methods utilize templates structured as *natural language inference (NLI)*-style prompts (section 3.1.1) or *cloze*-style prompts (section 3.1.2) in a zero-shot and/or few-shot setup to predict the labels for the downstream task. Works such as Logan IV et al. (2021) demonstrated few-shot training using *cloze*-style task-agnostic null-prompt, while FLAN (Wei et al., 2021) utilized NLI-style instruction templates and performed instruction tuning to improve the zeroshot performance. However, due to the large size of the model (137B parameters), we find the work unsuitable to be used in creating a low resource pipeline. On the other hand, LMBFF (Gao et al., 2021) and Pattern Exploiting Training (PET) (Schick and Schütze, 2021) utilized a relatively smaller PLM (340M parameters) on *cloze*-style prompts. LMBFF showed that a few demonstrative examples during task fine-tuning provide additional context to better learn the prompts and report improvements over PET (Schick and Schütze, 2021). Considering the benefits of LMBFF (Gao et al., 2021) over other methods in creating a low resource pipeline, we utilize this approach to perform prompt-based fine-tuning.

3 Methodology

The proposed methodology is a two-step process (Figure 1). First, we *prompt the PLMs with 'human-in-the-loop'* as a strategy to produce weak labels for the unlabeled training data. Next, we train a WSM on these weak labels. In the subsequent sections, we describe the two steps in detail.

3.1 Prompting PLMs to obtain weak labels

3.1.1 Prompting: NLI-style

In NLI-style prompts, the input utterance is transformed to a premise-hypothesis pair of an utterance and a prompt respectively. This transformed input is fed to an entailment model. For example, for input utterance *'I am happy.'* and prompt *'The sentiment of the speaker is positive'*, an entailment in this case denotes that class-label is positive. Prompt is designed to reflect the class label of utterance if prompt (hypothesis) entails the utterance (premise). For each premise, the class label associated with the prompt having highest entailment score is treated as the weak label. For prompting, we compare a couple of pre-trained models

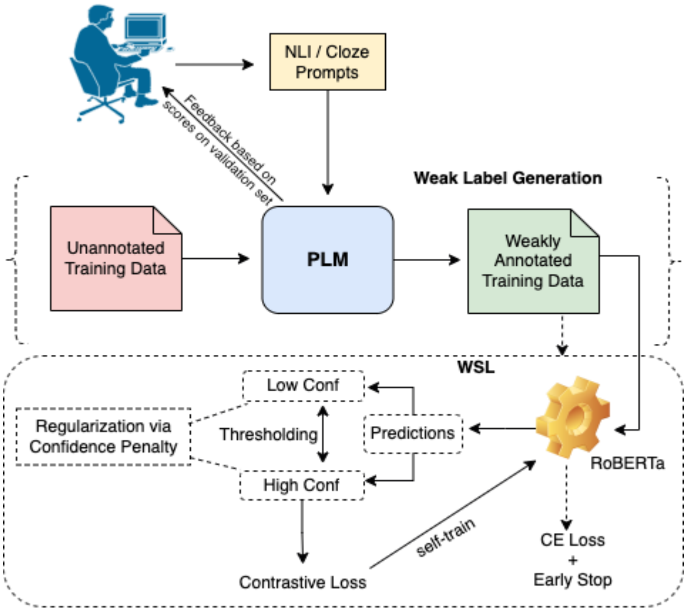

Figure 1: Proposed weak supervised framework.

namely `bart-large-mnli-yahoo-answers` and `roberta-large-mnli` available at Hugging Face library[1] (Wolf et al., 2019). Based on the results over the same set of prompts, we find that bart-based model predicts more accurate class labels, which we select for conducting all the NLI experiments.

3.1.2 Prompting: *Cloze*-style

In *cloze* prompts, a piece of text is inserted in the input examples, so that the original task can be formulated as a masked language modeling (MLM) problem. For example, considering input utterance *'I am happy.'*, the prompt based system is fed a transformed utterance containing a `[mask]` token to form utterance-instruction pair, where instruction is *'The sentiment of the speaker is* `[mask]`*'*. For the masked token, PLM generates a probability distribution over a set of verbalizers representing individual classes (Table 5 in A.1). The class corresponding to the verbalizer with highest probability is taken as the weak label. Inspired by LMBFF (Gao et al., 2021), we leverage pre-trained `roberta-large` (Liu et al., 2019) with an objective to fill the `[mask]` token in the prompt. The resulting sentence is concatenated with one demonstration per class, in the similar fashion as Gao et al. (2021). This scheme leverages additional context around the input sentence to predict appropriate class labels. However, we note *cloze* prompts to be more sensitive than NLI prompts to the changes in the demonstrations chosen. Hence, we take into account an extended version of *cloze* prompts which also requires demonstrations and performs a task fine-tuning using a few-shot setup to solve a specific task. The need of task fine-tuning with *cloze*-style prompts is studied in detail in A.3.

3.2 Weak Supervised Learning

The noisy labels obtained in the previous step are utilized to train a WSM following a certain noise-correction mechanism (Ren et al., 2020; Yu et al., 2021). We observe that Yu et al. (2021) with its contrastive self-training label correction strategy outperforms various recent WSL baselines (Ren et al., 2020; Wang et al., 2019). Directly inspired by the technique utilized in Yu et al. (2021), we adopt a noise-handling strategy which uses contrastive loss for self-training to improve over the performance of weak source(s) iteratively. This strategy is robust to the label noise. Further, we assume that there is no gold annotations available for training. The model undergoes two steps:

Initial fine-tuning: Firstly, it trains a `roberta-base` (Liu et al., 2019) encoder over noisy labels with cross-entropy objective only for fewer steps (early epochs). This prevents the model from over-fitting the label noise and simultaneously helps to generalize the learning. During training, the model encourages to utilize soft labels to reduce the aggressive gradient updates. This prevents over-fitting the label noise.

Self-training: Further, the fine-tuned encoder

[1] https://huggingface.co/models

continues to learn over its own highly confident predictions (identified by keeping a threshold on confidence scores of model predictions obtained from the first step) via optimising a contrastive loss (Huang et al., 2022; Yu et al., 2021). In particular, contrastive loss is applied to regularize the feature space by bringing samples with the same weak labels closer together while separating apart the samples with different weak labels. This makes it easier to discern between the representations for data belonging to various classes and aids the classifier in producing more accurate predictions. As noise can aggressively back-propagate through the model during training on noisy labels, we prevent such events by employing confidence-based sample re-weighting and regularisation schemes. In order to lessen the impact of incorrect predictions, the re-weighting technique promotes the inclusion of samples with high prediction confidence scores solely, anticipating that these samples are likely to be correctly classified. Additionally, we explore a couple of prior works (Yu et al., 2021; Pereyra et al., 2017) on entropy-based confidence penalty (using KL-divergence) for label smoothing and model regularization. As a result, the model smoothens the confident predictions to avoid sampling overconfident samples, hence reducing the impact of inaccurate noisy class labels. This process is continued for several iterations to help the model progressively learn over its own confident predictions and is called self-training (Yu et al., 2021; Wang et al., 2019; Ren et al., 2020). The one which we utilize in our work is a result of the combination of ideas strategized from a few strong prior works (Yu et al., 2021; Pereyra et al., 2017; Huang et al., 2022). The model utilized in our work is applicable to learn effectively and efficiently over weak annotations derived from various noisier sources and is robust to high intensity of label noise.

4 Experiments

4.1 Dataset

We consider three SLU datasets from various domains for conducting our experiments. **CMU-MOSI** is a sentiment dataset consisting of multiple modalities (Zadeh et al., 2016). In this work, we only consider text modality to perform sentiment classification. Similar to previous works (Kumar and Vepa, 2020; Tsai et al., 2019), we use train-test-valid split of 1284, 654 and 229 samples respectively. We use the **Switchboard Disfluency** (Godfrey et al., 1992) (SWBD-D) to classify the utterances into *fluent* and *disfluent* categories. We use the provided splits of train, test and validation set. For emotion task, we utilize **IEMOCAP** (Busso et al., 2008). Since distinguishing between *happy* and *excited* or *angry* and *sad* is a challenging task without audio modality, in the scope of this work, we binarize the emotion in IEMOCAP dataset to *positive* and *negative* emotion. *Positive* emotions comprise of *happy* and *excited* while *negative* emotions comprise of *sad*, *angry* and *frustrated*. We use the provided split of train, test and validation having 3270, 1207, 867 samples respectively. *It is to be noted that we use training set as unlabeled data and hence do not use ground-truth of the training splits to train the WSM.*

4.2 Weak Sources

Rule-based weak source: For sentiment and emotion classification, we use SentiWordNet (Esuli and Sebastiani, 2006) and AFINN (Nielsen, 2011) lexicons along with a rule based system called VADER (Hutto and Gilbert, 2014). We map the positive scores to positive sentiment/emotion and similarly, for negative sentiment/emotion. Empirically, we find VADER and AFINN to perform better for sentiment, while SentiWordNet and VADER perform best for emotion. For disfluency classification on SWBD-D, the labeling functions are created based on the occurrence of filler words, repetitions and soundex (Odell and Russell, 1918) codes. We create an additional rule by aggregating the weak sources via majority voting. We report the mean performance of rule-based weak sources on Macro-F1 and Coverage metrics in Table 1.

Dataset	Coverage	Macro-F1
MOSI	74.3±4.2	71.0±3.7
SWBD-D	84.4±10.5	73.6±7.4
IEMOCAP	63.9±17.2	46.6±0.3

Table 1: Rule-based weak source performance; Coverage represents %samples labeled by the rules; Macro-F1 is reported only over the samples covered by the rules

Prompt-based weak source: Since rule-based weak sources have varying coverage and require significant manual efforts and time, there is a need of a low-effort method that can generate weak labels over a given dataset with reliable performance. We propose prompting PLMs to obtain the weak labels. Specifically, we compare task-agnostic and task-specific prompts. Task-agnostic prompt represents a general instruction that can be shared across tasks while a task-specific prompt incorporates the task information in its verbiage which helps model with an additional context. To better understand what generalizability in prompts means and how are the generalized (task-agnostic) prompts different from the task-specific ones, we provide representative examples of task-specific and task-agnostic prompts in A.1 in a *cloze*-style template. Such prompts contain a `[mask]` which is expected to be replaced by an appropriate token called *verbalizer*. The *verbalizer* determines the task under the consideration. By choosing different verbalizers, we can utilize same prompt for multiple tasks in a task-agnostic setting. While, in a task-specific setting, contextual prompts (Table 5) are utilized to help predict the verbalizers more accurately. We can translate *cloze*-style prompts used in our experiments to *NLI*-style (i.e. hypotheses) by replacing the `[mask]` tokens with appropriate *verbalizers*. The hypotheses are entailed with the premises and yields entailment scores for each premise-hypothesis pair. The class which is a mapping to the verbalizer with highest entailment score is selected as the weak label for the premise.

Dataset	TSP	TAP
MOSI	83.5±3.3	82.8±1.3
SWBD-D	74.6±5.1	68.9±6.5
IEMOCAP	68.7±3.3	68.0±1.8

Table 2: Mean performance (Macro-F1) of task-specific (TSP) and task-agnostic (TAP) prompts on test set. Mean is computed across both NLI and *cloze* prompts.

Dataset	NLI	Cloze
MOSI	83.2±1.6	82.7±1.0
SWBD-D	42.0±15.6	75.2±3.9
IEMOCAP	70.3±0.4	71.4±0.6

Table 3: Mean performance (Macro-F1) of NLI and Cloze prompts on test set. We utilize both TAP and TSP for the average calculation.

With differences in prompts and verbalizers, the performance of the weak source may vary. Thus, experiments are performed on multiple prompts and/or verbalizers involving '*human-in-the-loop*'. We ask the annotators to sample 16 data points per class and annotate them with ground truth labels which we use as a validation set to evaluate the performance and fairly compare the varieties of prompts. The performance of each prompt acts as a feedback to direct the prompt curators whether or not to continue designing more variations of prompts. The process of designing more prompts terminates when no improvement in performance of prompts is observed for 5 consecutive attempts. Among the wide range of prompts designed as a consequence of the *to-and-fro* process between PLM and humans, we choose to report average across the top-3 performing prompts in Table 2. In the process, by leveraging *flexibility* as a feature of prompt-based source, we modify the prompt structure to create various task-specific prompts. On the other hand, the task-agnostic prompts can be utilized across various tasks performed in our experiments demonstrating *generalizability* of the proposed approach. Additionally, the performance scores of prompt-based weak source reported in Table 2 shows that PLMs when prompted, have potential to generate accurate weak labels (*potency*). We note that for every dataset, task-specific prompts work better than the task-agnostic prompts. This gain is significant for SWBD-D dataset denoting the need for language models to rely on task specific context unlike other tasks. Additionally, we note that NLI-style prompt produces better results for MOSI, while *cloze*-style prompt produces more reliable labels for SWBD-D and IEMOCAP dataset (Table 3). This could mean that complex tasks like disfluency and emotion classification require the model to learn on task-specific contexts which we perform with *cloze*-style prompts.

4.3 Baseline

We compare the performance of the proposed setups with a fully-supervised and other low-resource setups like few-shot learning (FSL) and zero-shot learning (ZSL): ***Oracle*** represents the performance score on test set obtained when a pre-trained RoBERTa (Liu et al., 2019) is trained on gold labels in a fully-supervised fashion. ***Meta-tuning*** is a state-of-the-art work (Zhong et al., 2021) in ZSL where authors propose utilizing question prompts for classification tasks, where zero-shot objective is directly optimized by fine-tuning on a meta-dataset. ***k-Classifier*** is a few-shot setup, a RoBERTa learns on a train-set consisting of only 16 examples per class. ***DNNC*** (Zhang et al., 2020) utilizes discriminative nearest neighbor classifier, is a state-of-the-art model for few-shot and out-of-scope intent prediction task. We use a 16-shot setup.

Further, we compare the performance of various weak sources on the test set: ***Rule*** represents the

	MOSI	SWBD-D	IEMOCAP
Oracle	*86.1±0.4*	*94.5±2.0*	*80.5±0.4*
Meta-tuning	80.3±2.0	49.1±2.7	61.6±1.8
k-Classifier	73.1±7.0	74.3±7.3	61.4±5.9
DNNC	79.9±0.8	63.9±1.7	62.3±7.4
Rule	63.0±3.2	70.9±7.1	33.8±3.1
TAP	81.5±1.6	71.1±7.5	69.0±2.8
TSP	83.5±1.2	73.2±5.2	69.8±1.7
WSL-Rule	74.6±5.4	76.4±1.5	41.0±1.8
WSL-TAP	82.8±1.8	77.5±3.9	71.5±0.3
WSL-TSP	**84.5±0.9**	**81.9±3.7**	**71.9±0.9**
Best WSL	85.26	83.86	72.47

Table 4: Comparative results of the baselines and proposed weak supervised pipeline on Macro-F1 scores.

performance of rule-based weak sources. For the calculation of recall, the samples not covered by the rules are considered to represent false negatives. This represents a heuristic-based classifier. ***Task-Agnostic-Prompt (TAP)*** and ***Task-Specific-Prompt (TSP)*** represent the performance scores obtained by prompting the PLM with a task-agnostic and task-specific prompt directly over the samples in test set, without training a WSM.

Once the weak labels are obtained on the unlabeled training data, a WSM is trained to design: ***WSL-rule*** where a WSM is trained over the weak labels derived from rule-based weak source discussed in Section 4.2. ***WSL-TAP*** and ***WSL-TSP*** are the proposed low-resource pipelines which train a WSM on weak labels obtained from task-agnostic and task-specific prompts respectively.

5 Results and Analysis

We compare our proposed framework (WSL-TAP, WSL-TSP) with other baselines in Table 4. We report the experimental results as mean performance scores with standard deviations. *Rule*, TSP and TAP represent the mean performance score across various rule-based weak sources (refer section 4.2), task-specific and task-agnostic prompts (considering various styles - NLI and *cloze* both) respectively. *WSL-Rule*, *WSL-TSP*, *WSL-TAP* represent the average scores obtained on training the WSM over various rule-based heuristics, task-specific prompts and task-agnostic prompts.

Proposed WSL vs Low-Resource Baselines: The results show that a WSM trained on prompt based weak labels (*WSL-TAP, WSL-TSP*) outperforms other baselines including state-of-the-art zero-shot (meta-tuning) and few-shot (DNNC) approaches. *WSL-TSP* outperforms the few-shot method by more than 4% on MOSI, 7% on SWBD-D and 9% on IEMOCAP dataset. Further results show that training a weak supervision model (*WSL-TAP, WSL-TSP*) over prompt-based weak labels bridges the gap with the Oracle model. Specifically, training a WSM improves the F1 scores by 1% for MOSI, 8.2% for SWBD-D and 2.1% for IEMOCAP. This shows the effectiveness of proposed pipelines for training a low-resource model against few-shot and zero-shot methods.

Proposed WSL vs Rule-based WSL: The proposed weak supervision pipeline on prompt-based weak source also outperforms a traditional rule based weak supervision pipeline (*WSL-Rule*). The distinction between the performance of rule and prompt-based WSL pipeline is more evident for MOSI and IEMOCAP datasets. The proposed method outperforms rule based pipeline by 10% in MOSI, 5% in SWBD-D and 31% in IEMOCAP dataset (*WSL-TSP* vs *WSL-Rule* in Table 4). The higher gap in performances on MOSI and IEMOCAP could be related to the worse performance of rules, where weak labels generated from semantic rules are less accurate and have lower coverage than SWBD-D dataset (Table 1). We see that *Rule* has particularly lower performance on IEMOCAP owing to limited coverage of such rules, while both *TAP* and *TSP* consistently outperform the weak labels obtained from rules. Thus, in addition to reducing the manual effort in writing rules for labeling the data, the prompt-based method generates more accurate labels to annotate the unlabeled data.

Task-agnostic vs Task-specific Prompts: We observe that weak labels obtained from task-specific prompts (*TSP*) are consistently better than task-agnostic prompts (*TAP*). Likewise, *WSL-TSP* outperforms *WSL-TAP* on all tasks. While the best results are obtained from task-specific prompts, even task-agnostic prompts outperform a rule-based pipeline. Hence, the proposed pipeline (WSL-TAP, WSL-TSP) could solve the bottleneck of labeling the data while training a WSM.

Best WSL Scores: Finally, we report the best scores obtained with proposed pipeline: MOSI = 85.26%; SWBD = 83.86%; IEMOCAP = 72.47%. We note that the best score obtained on MOSI dataset is competitive with the Oracle results, which could be explained by a highly reliable weak labels obtained from prompt-based method on MOSI. However, the performance on SWBD-D and IEMOCAP are lower than Oracle (refer to section A.4 for explanation) but are strongly better than state-of-the-art low resource methods on zero-shot and few-shot methods. Thus, we show that low resource pipeline for SLU tasks could be effectively trained via proposed pipeline.

Quantifying the importance of human efforts: We outsource the task of curating prompts to 5 proficient English speakers. We observe that it takes around 17.0±6.0 minutes on an average across the speakers to come-up with the first good prompt for the benchmark SLU tasks. While creating rules for the same under the restriction of given guidelines (section A.5), it takes *roughly around an hour* to come up with pattern-based heuristics for a certain dataset. This observation shows that we can save around 72% of annotation time by avoiding rule-based heuristics and relying only on prompt-based weak source. The reason why designing prompts takes relatively lesser time compared to designing rules is due to the fact that rule-based baselines have to be balanced between coverage and precision, along with the higher complexity of coding the rules against the designing the prompts. We consider rules (or a set of rules unioned together) to be good if it performs better than a baseline of majority class in terms of precision and cover at least 25% of the unlabeled samples. Additionally, these rules have to written in a framework which is a time-consuming step as well against simply writing a prompt which is nothing but a textual sentence. So, iterating with prompts is much faster as compared to iterating on rules. Furthermore, in many cases, a bunch of rules have to be unioned to achieve this criteria on coverage and precision and hence, it takes a longer cycle to identify the set of rules required to generate weakly labeled training data. As it is evident in Table 4 that prompt-based baselines (TAP, TSP) outperform heuristics (Rule), thus, instead of creating rules, we rather encourage human effort in curating more variations of prompts by utilizing a fraction of the expensive time we saved using the proposed weak sources.

6 Conclusion

In this work, we show effectiveness of utilizing prompt-based methods as universal weak sources to develop low-resource models for wide range of benchmark SLU tasks. We show that the proposed method outperforms traditional methods of rule based WSL as well as state-of-the-art methods on other low resource settings like ZSL and FSL. In future, we would like to study the application of automatic and soft prompts to generate the weak labels and the extension of work to multilingual tasks where limited training data is available. We would also like to explore areas around incorporating the human-in-the-loop feedback during the training process instead of only restricting it to improve on the quality of weak labels.

Limitations

The proposed work has dependency on manual prompts. Curating prompts can be subjective across different individual. This can cause variations in the results. Moreover, we also notice that the score on test set for the predictions generated by PLMs is sensitive to the change in prompt tokens. However, our work offers a future opportunity for researchers to use the proposed setup with continuous prompts to address the problem of sensitivity caused by such manually curated prompts. Moreover, the technique of prompt-based fine-tuning (Gao et al., 2021) of PLM which we utilize to infer weak label for a *cloze*-style prompt is constrained to predict class label consisting of a single token only.

References

Carlos Busso et al. 2008. IEMOCAP: interactive emotional dyadic motion capture database. *Lang. Resour. Evaluation*, 42(4):335–359.

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, NAACL-HLT 2019, Minneapolis, USA*, pages 4171–4186.

Andrea Esuli and Fabrizio Sebastiani. 2006. SENTIWORDNET: A publicly available lexical resource for opinion mining. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation, LREC, Genoa, Italy*, pages 417–422.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021*, pages 3816–3830.

John J. Godfrey, Edward Holliman, and Jane McDaniel. 1992. SWITCHBOARD: telephone speech corpus for research and development. In *1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP, California, USA*, pages 517–520.

Michael A. Hedderich and Dietrich Klakow. 2018. Training a neural network in a low-resource setting on automatically annotated noisy data. In *Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP, DeepLo@ACL 2018, Melbourne, Australia*, pages 12–18.

Bin Huang, Adi Alhudhaif, Fayadh Alenezi, Sara A. Althubiti, and Chaoyang Xu. 2022. Balance label correction using contrastive loss. *Inf. Sci.*, 607:1061–1073.

Clayton J. Hutto and Eric Gilbert. 2014. VADER: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM, Michigan, USA*.

Ayush Kumar and Jithendra Vepa. 2020. Gated mechanism for attention based multi modal sentiment analysis. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain*, pages 4477–4481.

Yinhan Liu et al. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Robert L Logan IV, Ivana Balažević, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2021. Cutting down on prompts and parameters: Simple few-shot learning with language models. *arXiv preprint arXiv:2106.13353*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Finn Årup Nielsen. 2011. A new ANEW: evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages, Crete, Greece*, volume 718 of *CEUR Workshop Proceedings*, pages 93–98.

KM Odell and RC Russell. 1918. Soundex phonetic comparison system. *US Patent*, 1261167.

Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net.

Alexander Ratner, Stephen H. Bach, Henry R. Ehrenberg, Jason A. Fries, Sen Wu, and Christopher Ré. 2020. Snorkel: rapid training data creation with weak supervision. *VLDB J.*, 29(2-3):709–730.

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, Online Event*, volume EMNLP 2020 of *Findings of ACL*, pages 3739–3754.

Teven Le Scao and Alexander M. Rush. 2021. How many data points is a prompt worth? In *NAACL-HLT, 2021, Online*, pages 2627–2636.

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, EACL*, pages 255–269.

Stefan Schweter and Alan Akbik. 2020. FLERT: document-level features for named entity recognition. *CoRR*, abs/2011.06993.

Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J. Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2019. Multimodal transformer for unaligned multimodal language sequences. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy*, pages 6558–6569.

Hao Wang, Bing Liu, Chaozhuo Li, Yan Yang, and Tianrui Li. 2019. Learning with noisy labels for sentence-level sentiment classification. 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 2019, Hong Kong*, pages 6285–6291.

Jason Wei et al. 2021. Finetuned language models are zero-shot learners. *CoRR*, abs/2109.01652.

Thomas Wolf et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang. 2021. Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*, pages 1063–1077.

Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. MOSI: multimodal corpus of sentiment intensity and subjectivity analysis in online opinion videos. *CoRR*, abs/1606.06259.

Jianguo Zhang et al. 2020. Discriminative nearest neighbor few-shot intent detection by transferring natural language inference. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online*, pages 5064–5082.

Ruiqi Zhong, Kristy Lee, Zheng Zhang, and Dan Klein. 2021. Adapting language models for zero-shot learning by meta-tuning on dataset and prompt collections. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic*, pages 2856–2878.

Zhi-Hua Zhou. 2018. A brief introduction to weakly supervised learning. *National science review*, 5(1):44–53.

A Appendix

A.1 Representatives: TAP and TSP prompts

We present some of the representative examples of task-agnostic (TAP) and task-specific (TSP) prompts we used in our work. Table 5 demonstrates examples of TSP. Here, we observe that the prompts design vary according to the task and associated class labels. However, design for TAP is independent of any task and its underlying class labels. Here, we demonstrate some examples of TAP:

- *The class best describing the text is* `[mask]`.
- *The text can be classified as* `[mask]`.

A.2 Hyper-parameters

We discuss the hyper-parameters used to run our experiments. We perform a grid search to optimize the performance of the proposed weak source with *cloze*-style prompts (as they require fine-tuning) on development set. Batch size is searched over {2, 4, 8} set. The best learning rate is searched in {1e-5, 2e-5, 3e-5}. We fix the number of demonstrations to 1 for faster training. As we use NLI-style prompts to directly infer the weak labels without performing any task fine-tuning, we do not need any hyper-parameter specifications for prompt-based weak source with this style of prompts. For training the weak supervised model on noisy labels obtained via prompt-based weak source, our experimental setup utilizes the AdamW optimizer (Loshchilov and Hutter, 2019) and the optimal learning rate is searched over {1e-5 , 2e-5, 3e-5}. The weak-supervised learning is carried out for a maximum of 5 epochs. A scheduler (with linear learning rate decay strategy) is utilized with 0.1 warm-up. However, there are some hyper-parameters which are specific to the model and the involved training strategy. Hence, we recommend to look into the works of Yu et al. (2021) and Gao et al. (2021) as our work is inspired by them. Their research work demonstrates the behavioral change in model performance due to change in such hyper-parameters, which we directly use in our work.

A.3 *Cloze*-style prompts with demonstrations

In the scope of our work, we utilize only the extended version of *cloze*-style prompt which require task fine-tuning as discussed in section 3.1.2 but we do not change the nomenclature and refer to it as *cloze*-style prompts everywhere. However, we would now like to differentiate the *cloze-style prompt-based weak source WITH (w/) task fine-tuning* and *cloze-style prompt-based weak source WITHOUT (w/o) task fine-tuning*. We conduct an experiment on the benchmark SLU datasets to investigate if *cloze*-style prompts w/o task fine-tuning can be used to infer class labels and we observe that there is a mandatory need for fine-tuning the PLM with this category of prompts on downstream tasks. We compare the columns in Table 6 to demonstrate the incapability of *cloze-style prompt-based weak source w/o task fine-tuning* in deriving good quality weak labels for training a WSM. We also observe the higher gains in performance post task fine-tuning. These observations encourage us to rely only on the extended version of *cloze*-style prompts to derive weak labels for training a WSM.

A.4 Does a PLM naturally understand some tasks better than the others?

From Table 4, we surprisingly observe that the performance gap between Oracle result and the proposed frameworks (WSL-TSP and WSL-TAP) is low for MOSI but significantly higher for IEMOCAP and SWBD-D. This could be related to the nature and complexity of the tasks performed with each of these datasets. We define the complexity of a task by a measure of how well a PLM performs on that task without any downstream task fine-tuning. As we use NLI-style prompts to directly infer the class labels for unlabeled samples, these prompts can be useful in quantifying the difficulty faced by PLM in solving a certain task. Hence, we compare our results on NLI prompts across the tasks from Table 3.

Apparently, the order of complexity is MOSI (sentiment) < IEMOCAP (emotion) < SWBD-D (disfluency). This shows that a PLM already comprehends the sentiment task to some degree. Hence, without requiring any task fine-tuning, PLM predicts class labels quite accurately. However, dis-

Dataset	Task-specific prompt	verbalizer : class label
MOSI	The sentiment of the speaker is `[mask]`.	positive : positive, negative : negative
SWBD-D	The speaker `[mask]` takes a pause while speaking!	never : fluent, often : disfluent
IEMOCAP	I have `[mask]` emotions.	happy : positive, sad : negative

Table 5: Representative examples of TAP

Dataset	w/o fine-tuning	w/ fine-tuning
MOSI	67.5±2.7	83.3±1.2
SWBD-D	49.3±5.6	73.2±1.6
IEMOCAP	51.6±8.4	69.1±2.7

Table 6: Performance (Macro-F1) of *cloze*-style prompts on the train-set of WSM w/ and w/o task fine-tuning

fluency classification appears to be a task that a PLM is not particularly adept at. The fact that we do not deduce the class labels for the disfluency classification using NLI-style prompts is an evidence for this. We come to a conclusion that PLMs straightaway do not comprehend the pauses in conversation. Hence, we fine-tune the PLM with few-shots to help it understand the conversational pauses (refer to Cloze column in Table 3). However, the performance gap between the Oracle and proposed approaches still remains high which explains that disfluency classification is not a natural property of PLM. For emotion classification, the training is challenging as it is performed at utterance level excluding the surrounding informative context in the dialogue. The struggle in learning intensifies when the PLM is trained on few shots only. This could be a logical justification for the large disparity between Oracle and the suggested technique on IEMOCAP.

However, we encourage further investigations on ways to figure out the reasons for the uneven variations in gaps between Oracle and proposed pipelines.

A.5 Guidelines for Designing Rules

While creating rules for a dataset, it requires efforts to keenly observe the data to identify patterns and map them to the class labels. We ask the experts to manually create the rules for the benchmark datasets, under the assumption that rule-based tools pre-exist only for limited tasks. A rule is considered acceptable if it performs better than a majority baseline in accuracy and can cover at least 25% of the unlabeled samples. As observing the patterns across the complete dataset is tedious, we ask them to consider only 16 annotated samples per class to observe the patterns and use them to evaluate the accuracy of the rules. To design heuristics, we ask annotators to observe the common patterns (lexicons, phrases) to design a strategy that can be utilized to annotate the unseen samples. We also provide liberty to annotators to combine a bunch of rules for a certain dataset or task as necessary.