RulePrompt: Weakly Supervised Text Classification with Prompting PLMs and Self-Iterative Logical Rules

Miaomiao Li
Institute of Software, Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China
limiaomiao22@mails.ucas.ac.cn

Jiaqi Zhu*
Institute of Software, Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China
zhujq@ios.ac.cn

Yang Wang
Institute of Software, Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China
wangyang223@mails.ucas.ac.cn

Yi Yang
Institute of Software, Chinese Academy of Sciences, Beijing, China

Yilin Li
Institute of Software, Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China

Hongan Wang
Institute of Software, Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China

ABSTRACT

Weakly supervised text classification (WSTC), also called zero-shot or dataless text classification, has attracted increasing attention due to its applicability in classifying a mass of texts within the dynamic and open Web environment, since it requires only a limited set of seed words (label names) for each category instead of labeled data. With the help of recently popular prompting Pre-trained Language Models (PLMs), many studies leveraged manually crafted and/or automatically identified verbalizers to estimate the likelihood of categories, but they failed to differentiate the effects of these category-indicative words, let alone capture their correlations and realize adaptive adjustments according to the unlabeled corpus. In this paper, in order to let the PLM effectively understand each category, we at first propose a novel form of rule-based knowledge using logical expressions to characterize the meanings of categories. Then, we develop a prompting PLM-based approach named RulePrompt for the WSTC task, consisting of a rule mining module and a rule-enhanced pseudo label generation module, plus a self-supervised fine-tuning module to make the PLM align with this task. Within this framework, the inaccurate pseudo labels assigned to texts and the imprecise logical rules associated with categories mutually enhance each other in an alternative manner. That establishes a self-iterative closed loop of knowledge (rule) acquisition and utilization, with seed words serving as the starting point. Extensive experiments validate the effectiveness and robustness of our approach, which markedly outperforms state-of-the-art weakly supervised methods. What is more, our approach yields interpretable category rules, proving its advantage in disambiguating easily-confused categories.

*Corresponding Author

WWW '24, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0171-9/24/05.
https://doi.org/10.1145/3589334.3645602

CCS CONCEPTS

• **Information systems** → **Web mining**; **Clustering and classification**; • **Computing methodologies** → *Learning settings*; *Rule learning*.

KEYWORDS

weak supervision; text classification; seed word; pre-trained language model; prompt; logical rule; rule mining; pseudo label

ACM Reference Format:
Miaomiao Li, Jiaqi Zhu, Yang Wang, Yi Yang, Yilin Li, and Hongan Wang. 2024. RulePrompt: Weakly Supervised Text Classification with Prompting PLMs and Self-Iterative Logical Rules. In *Proceedings of the ACM Web Conference 2024 (WWW '24), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3589334.3645602

1 INTRODUCTION

With the rapid development of Internet, an abundance of textual content is produced across news media and social networks. It is significant and challenging to classify these texts into predefined categories, especially when up-to-date labeled data are hard to access due to the dynamic and open nature of the Web. Consequently, there has been a growing interest in weakly supervised text classification (WSTC) [16, 23, 31, 32, 39, 40], also known as zero-shot or dataless text classification [3, 4, 13, 14, 22–24, 29, 30, 33, 34, 36, 41], which only requires a limited set of seed words (label names) for each category.

Recently, the proliferation of prompting Pre-trained Language Models (PLMs) greatly bolstered the WSTC task, but their performances still lag behind supervised methods [32]. Since no labeled data are available as evidence, relying solely on seed words for grasping category meanings proves inadequate. In previous researches, many approaches either provided manual verbalizers of categories or automatically discovered them based on word embedding similarity. Taking them as additional knowledge, some studies estimated category likelihoods by tapping into the generative capability of PLMs [40, 41], and others leveraged PLM's effective vector representations to calculate the similarity or entailment between

texts and categories [26, 30]. However, most of them failed to differentiate the effects of these category-indicative words (abbreviated as indicative words). Although NPPrompt [41] did calculate and utilize the weights of them, their roles in classification remained independent of each other and lacked adaptive adjustments for the current corpus, so cannot accommodate ever-changing Web environment.

However actually, the effect of each category-indicative word varies and is worth further explorations. Certain words can determine the category on its own, like the label names, while others need to be used cooperatively to distinguish between easily-confused categories. For example, the word "*penalty*" itself cannot signify the "*Sports*" category, but when combined with "*goal*", the text is likely to talk about a football match. Conversely, an additional word "*company*" could imply the "*Business*" category rather than "*Sports*". Therefore, a simplistic set of indicative words is not enough to cover the full meanings of categories. Instead, logical operations such as conjunction and disjunction are appropriate to capture the correlations of these words as enriched and precipitable knowledge for weakly supervised classification. Luckily, the flexibility of prompting PLMs just offers an opportunity to apply these logical rules in the template to achieve precise semantic representations of categories.

It is obvious that logical rules are difficult to set manually as prior knowledge, but they can be mined from preliminarily categorized texts with the aid of pseudo labels generated by the PLM. Furthermore, the mined rules and pseudo labels can mutually enhance each other in an alternative way, establishing a self-iterative closed loop for knowledge acquisition and utilization, with seed words as the starting point. That poses two main challenges: (1) When inaccurate pseudo labels are available, how to identify candidate category-indicative words using the PLM and build correlations among them by means of logical rules to characterize each category? (2) With imprecise logical rules, how to effectively transform them into the PLM template for classification by handling each logical operator discriminatively, and then update the pseudo label assigned to each text?

To address these issues, this paper at first proposes a novel kind of rule-based knowledge in the form of logical expressions for category understanding in WSTC. Each category is represented by a disjunctive normal form, where indicative words serve as atomic propositions. Specifically, a single disjunctive term (one-literal clause) denotes strong and self-explanatory indicative words, while a clause of conjunctive form depicts the synergistic effect of weak and polysemous indicative words.

Based on this, a prompting PLM-based approach for text classification is developed, through iteratively updating both the pseudo label of each text and the logical rule of each category. That is realized mainly via two modules, rule mining and rule-enhanced pseudo label generation. The former first extracts signal words from each text by the PLM, and then regards these words as a transaction of the relevant category decided by the current pseudo labels. For each category, we mine frequent 1-itemsets (items) and 2-itemsets respectively from specific subsets of transactions, and construct the disjunctive normal form. In the latter module, the current logical rule for each category is injected into three PLM-based models, each providing a different perspective. Afterwards, a new pseudo label is generated for each text via integrating the results of these models. In addition, in each iteration, the PLM can be fine-tuned with a self-supervised loss to better align with the task requirements.

In summary, the contributions of this paper include:

- To the best of our knowledge, this is the first attempt to differentiate the effects of category-indicative words in the WSTC task and characterize category meanings through logical rules, thereby establishing a new paradigm for knowledge representation in this field.
- A novel approach leveraging prompting PLMs is presented to make the pseudo labels of texts and the logical rules of categories enhance each other iteratively. That facilitates a sufficient fusion of automatically generated rule-based knowledge and unlabeled data.
- Comprehensive experiments conducted on multiple real datasets demonstrate the effectiveness and interpretability of our approach. It consistently outperforms state-of-the-art weakly supervised methods, and yields intuitive logical rules for categories to avoid confusion.

2 RELATED WORK

2.1 Weakly Supervised Text Classification

Weakly supervised text classification (WSTC) demands minimal seed information, such as label names or extended keywords for each category, thereby significantly reducing the cost of text annotations. At an early stage, some researchers used auxiliary knowledge bases like Wikipedia to establish the semantic correlation between texts and labels [3, 29]. Subsequently, topic-model based methods emerged [4, 13, 14, 33, 34], which inferred category-aware topics from a limited set of seed words. In the last few years, neural methods has gained prominance [22, 23, 31, 36, 39]. They trained neural classifiers using pseudo labels of texts, often relying on generated pseudo-texts or PLMs to detect category-indicative keywords. For example, LOTClass [24] used label names as the only seed words, and introduced BERT for category understanding.

In recent time, prompt-based methods [6, 10, 25] have been extensively developed for the WSTC task. A lot of work harnessed the strong generative capability of PLMs with instruction templates for classification. For instance, NPPrompt [41] used initial word embeddings by PLM to automatically construct verbalizers without manual design or unlabeled corpus, and estimated the probability distribution over categories through weighted sum of these words. PIEClass [40] introduced a noise-robust method to iteratively self-train text classifiers and update pseudo labels, employing two fine-tuning strategies of PLMs to improve the quality of pseudo labels. WDDC [35] utilized the generated words at the [MASK] token as supervision signals, and proposed a latent variable model to train a word distribution learner and a text classifier simultaneously. Other approaches explored the vector representation power of prompting PLMs. PESCO [30] incorporated label descriptions into predefined prompts, formulating the WSTC task as a neural matching problem. Meanwhile, LIME [26] used large textual entailment models trained with external data to suggest seed words and infer text labels.

Although these methods have demonstrated inspiring performances, a gap still exists when compared to fully supervised methods. Due to the absence of labeled data, there is a notable need to

automatically extract and apply additional knowledge from unlabeled data during the classification process. Existing methods just relied on a set of category-indicative words, but have not taken the varying effect of these words into account, which leads to imprecise category understanding.

2.2 Logical Rules for Natural Language Processing Tasks

Recently, there have been increasing researches on the integration of logical rules into natural language processing tasks, aiming to improve the interpretability of neural network models.

Hu et al. [11] proposed a teacher-student framework combining deep neural networks with first-order logic rules, and transformed the structured information of logic rules into the weights of neural networks. TALLOR [15] addressed the named entity tagging problem by using a small set of seed logical rules as weak supervision, and further selected new accurate logical rules based on a hand-tuned threshold. PTR [9] incorporated logic rules to encode human prior knowledge and composed several manually designed sub-prompts into final task-specific prompts. PRBoost [37] viewed the top-k predictions at the [MASK] token of large-error instances as candidate rules through the disjunction operation, and then used human-selected ones to generate weak labels for model training.

However, most of these previous work required seed rules as initial supervision or human feedback when selecting accurate rules. In contrast, our approach focuses on the WSTC task, and establishes self-iterative closed loop for the acquisition and utilization of logical rules, eliminating the need for human intervention. Additionally, while existing PLM-based methods primarily employed single operator when composing decision rules, we leverage both the disjunction and conjunction operators to distinguish the strength and effect of indicative words, enabling a more precise understanding of categories.

3 PRELIMINARIES

In this section, we formulate the task of weakly supervised text classification (WSTC), and briefly introduce prompting PLMs as well as two roles of them as the foundation of our approach.

3.1 Problem Formulation

Given a corpus of unlabeled texts $D = \{D_1, \ldots, D_N\}$ and a set of target categories $Z = \{z_1, \ldots, z_K\}$ with a label name $l(z)$ for each $z \in Z$, weakly supervised text classification (WSTC) aims to assign a category label $z(d)$ to each text d. Following the extremely weak supervision setting [31], only the sole label surface name of each class is used as supervision here, without other seed words.

3.2 Prompting PLMs for Estimating Likelihoods

Prompt-based tuning applies cloze-style tasks to tune PLMs. A prompt is composed of a template $\mathcal{T}(\cdot)$ and a set $\mathcal{V}$ of selected words. We can fill each text d into the template $\mathcal{T}(\cdot)$ to obtain the prompt input $\mathcal{T}(d)$. For example, for the text classification task on news, the prompt can be written as:

$$\mathcal{T}(d) = d \text{ It is about [MASK] news.} \tag{1}$$

In vanilla prompt engineering, the verbalizer, i.e., an injective mapping function $\phi : Z \to \mathcal{V}$, links the category set and the set of selected words. Then, at the masked position, we can calculate the likelihood for each category via word probability distributions:

$$P(z|d) = P([\text{MASK}] = \phi(z) \mid \mathcal{T}(d)). \tag{2}$$

Recently, A lot of work studied for a verbalizer with richer label words to represent the category. Typically, NPPropmt [41] constructs a K-nearest-neighbor verbalizer, through searching over the whole vocabulary $\mathcal{V}$ for the top-k nearest words to the label name of z in the embedding space of the PLM, denoted as $\mathcal{M}(z)$:

$$\mathcal{M}(z) = \underset{v \in \mathcal{V}}{\text{Top}-K_0}\{\text{sim}(\mathbf{emb}(v), \mathbf{emb}(l(z))\}, \tag{3}$$

where $\mathbf{emb}(v)$ and $\mathbf{emb}(l(z))$ are the embedding vectors of word v and label name $l(z)$ respectively, and $\text{sim}(\cdot)$ means cosine similarity.

Then, we get the unnormalized probability for each category:

$$Q(z|d) = \sum_{v \in \mathcal{M}(z)} w(v, l(z)) \cdot \Theta([\text{MASK}] = v \mid \mathcal{T}(d)), \tag{4}$$

where Θ is the logit vector instead of probability for kernel smoothing, and $w(v, l(z))$ is the weight of the word v on the label name $l(z)$, defined in the softmax form:

$$w(v, l(z)) = \frac{\exp(\text{sim}(\mathbf{emb}(v), \mathbf{emb}(l(z))\})}{\sum_{v' \in \mathcal{M}(z)} \exp(\text{sim}(\mathbf{emb}(v'), \mathbf{emb}(l(z)))}. \tag{5}$$

Besides, NPPrompt uses more than one keywords for certain categories. The final score is calculated as follows:

$$Q(z|d) = \max_{v \in \Phi(z)} Q(v|d)), \tag{6}$$

where $\Phi(z)$ contains all keywords for category z, and $Q(v|d)$ is computed similar to Equation 4, replacing the category z by one of its indicative words v and the label name $l(z)$ just by v itself.

3.3 Prompting PLMs for Getting Signal Words

In addition to estimating category likelihoods, some work [35] utilized prompting PLMs to generate words which can summarize the content of the given text. That also depends on the probability distribution over $\mathcal{V}$, and can be used to get better supervision information than the words themselves appearing in the text. Formally, given a threshold K_1, for each text d, the top K_1 words with higher logits can be seen as signal words of d, denoted as $SW(d)$:

$$SW(d) = \underset{v \in \mathcal{V}}{\text{Top}-K_1}\{P([\text{MASK}] = v \mid \mathcal{T}(d))\}. \tag{7}$$

4 METHOD

In this section, we at first define logical rules of categories as a new kind of knowledge. Based on this, the framework of RulePrompt is presented followed by details of the three key modules.

4.1 Logical Rules of Categories

In this paper, we propose a novel kind of rule-based knowledge representation for categories, as additional weak supervision information in text classification. It takes automatically mined category-indicative words as atomic propositions, and build their correlations through logical expressions with disjunction and conjunction operators. Specifically, each category can be represented by a disjunctive normal form.

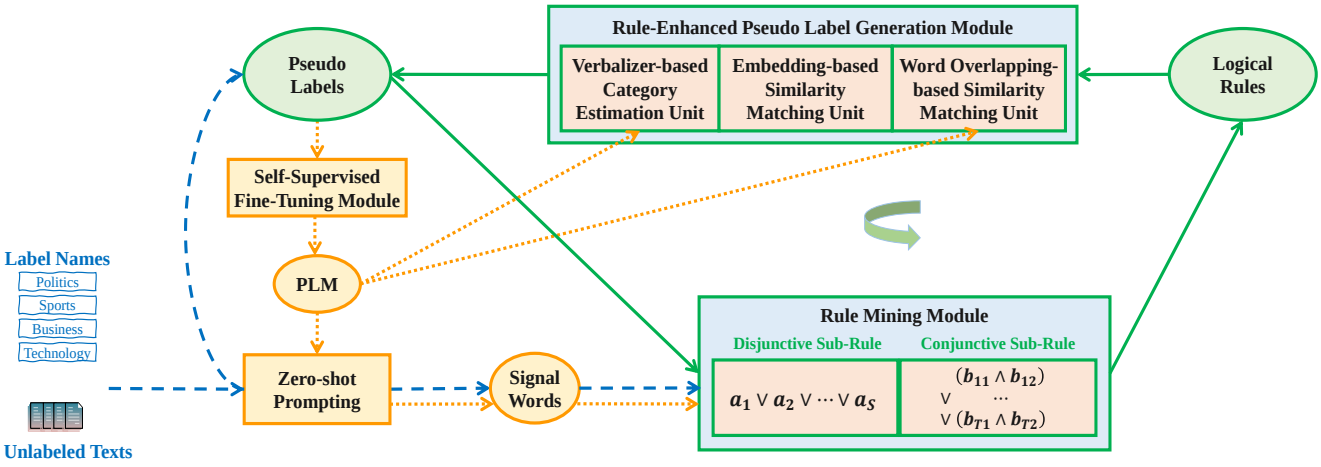


Figure 1: Framework of the Proposed Approach RulePrompt.

Definition 4.1 (Logical Rules of Categories). The meaning of each category z can be represented by a logical rule as follows:

$$r(z) = \Big(a_1 \vee \cdots \vee a_S\Big) \vee \Big((b_{11} \wedge b_{12}) \vee \cdots \vee (b_{T1} \wedge b_{T2})\Big), \quad (8)$$

where both a_j $(1 \le j \le S)$ and b_{j1}, b_{j2} $(1 \le j \le T)$ are indicative words of the category z. The rule can be divided into two sub-rules. The first S words are strong and can indicate the category on its own, so they are connected directly by the disjunction operator and compose the disjunctive sub-rule, denoted as $r^{\mathrm{d}}(z)$. On the contrary, the last $2T$ words are comparatively weak and need to act together to imply the category, so they are firstly paired with the conjunction operator, and then combined by disjunction. That is called the conjunctive sub-rule and denoted as $r^{\mathrm{c}}(z)$.

Despite the relations of indicative words above are not the same as those in classical logical rules, the ideas of conjunction and disjunction are actually utilized here to obtain precise semantic representations of categories in two views. Notice that a simplified version of logical operations is adopted by restricting the conjunction on just two words. That is reasonable and empirically effective, since the discrepancy between individual words and two-word pairs is essential, compared to n-word sets ($n > 2$).

4.2 Framework

On this basis, we propose a novel prompting PLM-based approach for the WSTC task as shown in Figure 1. At first, as the starting point with only label names, we leverage a classical zero-shot prompting method using PLM [41] to generate the initial pseudo labels and the signal words of texts (blue dashed line). Then, the approach enters the self-iteration between pseudo labels and category knowledge (logical rules) through mutual enhancement (green solid line). Meanwhile, the PLM is gradually optimized by self-supervised fine-tuning to adaptively support the main iteration above (yellow dotted line). To achieve the whole process, three modules are designed.

In the rule mining module, based on the current pseudo labels with confidence scores, we cluster the unlabeled texts assigned to each category into three sets. Then, with the signal words of each text obtained by PLM, frequent 1-itemsets (items) and 2-itemsets of each category are mined from the first two confident sets respectively, which composes the disjunctive normal form of the logical rule for each category.

In the rule-enhanced pseudo label generation module, we incorporate the current logical rules into three prompting PLM-based classification models from different perspectives to update pseudo labels. On the one hand, the words in the disjunctive sub-rule with higher support is directly used to obtain a richer verbalizer in a generation-based model. On the other hand, the whole rule is injected into templates to derive texts for similarity-based classification. That is realized in two views, global embedding similarity and local word overlapping. Finally, these results are averaged to get new pseudo labels of texts. s Moreover, in order to make the PLM accommodate this specific task, the self-supervised fine-tuning module is executed after each time pseudo labels are generated, employing self-supervised loss over high-confidence texts.

4.3 Rule Mining Module

In the weakly supervised setting, only label names are not adequate to reflect the meanings of categories. Thanks to the strong generative and representation capability of prompting PLMs, it is feasible to utilize the pseudo labels and signal words of texts to furthermore understand categories and enrich the prior knowledge. Since pseudo labels are imperfect, for the sake of mitigating error propagation, the selection of texts and signal words should be restricted to those with high confidence. Inspired by previous work [2, 17, 28], we define the confidence score (for the pseudo label) of a text as:

$$conf(d) = P(z_{(1)}|d) - P(z_{(2)}|d), \quad (9)$$

where $z_{(1)}$ and $z_{(2)}$ respectively denote the first and the second most probable label for text d computed by the prompting PLM. Compared to the highest probability, this difference value gives a better indication of how confident the PLM regards the current unique prediction.

However, for each category z, the numbers of texts appropriate to extract strong words and weak words are hard to determine, so

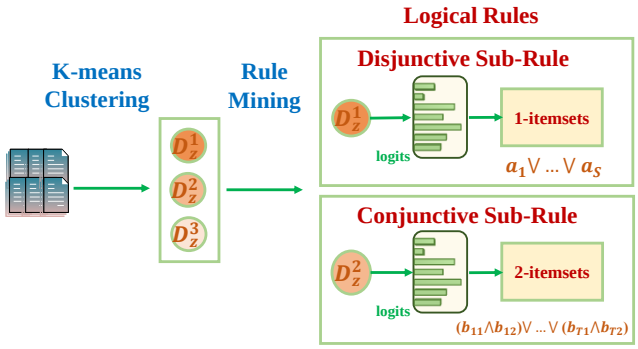


Figure 2: Rule Mining Module.

we adaptively cluster the texts assigned to z into three sets via K-means, based on the confidence scores. These texts with excellent, good and poor quality, are denoted as D_z^1, D_z^2 and D_z^3 respectively.

For the signal words of texts, the set $SW(d)$ computed by Equation 7 needs to be further filtered to guarantee their competence as indicative words. To this end, we utilize the whole corpus to pursue the speciality of signal words for the text, which we think can better imply the assigned category as well. The new unnormalized probability can be calculated as:

$$P'([\text{MASK}] = v \mid \mathcal{T}(d)) = \frac{P([\text{MASK}] = v \mid \mathcal{T}(d))}{\frac{1}{N}\sum_{d' \in D} P([\text{MASK}] = v \mid \mathcal{T}(d'))}. \quad (10)$$

Then, we select the top K_2 signal words with higher logits as the strong signal words, denoted as $SSW(d)$:

$$SSW(d) = \underset{v \in \mathcal{V}}{\text{Top}-K_2}\{P'([\text{MASK}] = v \mid \mathcal{T}(d))\}. \quad (11)$$

Next, we use frequent pattern mining [1, 8, 27] to obtain representative rules of categories. For D_z^1 and D_z^2, we treat each text as a transaction and each strong signal word of it as an item of the transaction. We at first pay attention to the most confident set D_z^1 to mine frequent 1-itemsets (items) with a predefined support threshold h_1, which compose the disjunctive sub-rule of z, as each of them alone is enough to indicate a category. The support of a word a in D_z^1 is calculated as:

$$sup(a, D_z^1) = \frac{\sum_{d \in D_z^1} I_1(a, d)}{|D_z^1|}, \quad (12)$$

where $I_1(a, d)$ is an indicator function expressing whether a is in the transaction of d, i.e.,

$$I_1(a, d) = \begin{cases} 1, a \in SSW(d), \\ 0, a \notin SSW(d). \end{cases} \quad (13)$$

In addition, for the set D_z^2 with moderate confidence scores, we mine 2-itemsets given another threshold h_2 to construct the conjunctive sub-rule. Although these words cannot represent a category individually, their co-occurrence in the set of strong signal words should also be captured. The support of a 2-itemset $b = b_1 \wedge b_2$ is calculated as:

$$sup(b, D_z^2) = \frac{\sum_{d \in D_z^2} I_2(b, d)}{|D_z^2|}, \quad (14)$$

where $I_2(b, d)$ is another indicator function expressing whether both b_1 and b_2 are in the transaction of d, i.e.,

$$I_2(b, d) = \begin{cases} 1, b_1 \in SSW(d) \wedge b_2 \in SSW(d), \\ 0, b_1 \notin SSW(d) \vee b_2 \notin SSW(d). \end{cases} \quad (15)$$

Besides, we need to exclude those pairs containing words also appearing in the frequent 1-itemsets of $D_{z'}^2$ for any other category z', which would bring confusion.

4.4 Rule-Enhanced Pseudo Label Generation Module

In this subsection, we present the reversed direction of the iteration, i.e., how to inject the mined logical rules of categories into the pseudo labels generation process. Considering the diverse capabilities of PLMs and the distinct roles that logical rules play within them, three units from different perspectives are designed to compute the probability of each text belonging to each category. Final results are obtained by averaging the outputs from the three units.

4.4.1 Verbalizer-based Category Estimation Unit. Since label names are too limited to characterize categories, the indicative words in our logical rules can be naturally used to expand the verbalizers in classical zero-shot prompting models (Equation 2). In view of the strictness of verbalizers, for each category, we only use the words in the first half of the disjunctive sub-rule according to their support values. The expanded set is written as $\Phi'(z) = \{l(z), a_1, a_2, \ldots, a_{\frac{S}{2}}\}$, which acts similarly with the manually crafted set of keywords in Equation 6. Besides, inspired by NPPrompt [41], the top-K_0 closest words to each of them are also used to complement the verbalizer (Equation 3). In this way, for a keyword $v \in \Phi'(z)$, we can get the probability $Q(v|d)$ and then take the maximum value among all keywords as the aggregated probability $Q(z|d)$ similar to Equation 6, as all of these words can imply the category independently.

Noticing that Q is an unnormalized probability, we use the softmax function to transform the value between 0 and 1, to get the normalized probability $P_1(z|d)$ from the first perspective:

$$P_1(z|d) = \frac{\exp(Q(z|d))}{\sum_{z' \in Z} \exp(Q(z'|d))}. \quad (16)$$

4.4.2 Embedding-based Similarity Matching Unit. To conduct a similarity-based matching between a text and a category through prompting PLM, an intuitive idea is to put the logical rule of each category into the [MASK] token of the template in Equation 1 to form a complete sentence [30, 37]. However, the expressions of conjunction and disjunction are not like natural language texts, which would affect the semantic understanding of the PLM. Hence, we handle each indicative word separately instead and combine them in different ways for disjunction and conjunction.

For the disjunctive sub-rule, we directly calculate embedding-based similarity between a text d and a category z as weighted sum of the similarity between d and each word a in the sub-rule of z:

$$ES^{\text{d}}(d, z) = \frac{\sum_{a \in r^{\text{d}}(z)} sup(a, D_z^1) \cdot \text{sim}(f(d), g(a))}{S}, \quad (17)$$

where $sup(a, D_z^1)$ is the support of word a in D_z^1, $f(d)$ is the sentence embedding of text d, and $g(a) = f(\mathcal{T}'(a))$ is the embedding of the template after removing "d" and replacing [MASK] with a.

While for the conjunctive sub-rule, besides that the outer disjunction operations can be handled in the same way, the similarity between d and each 2-itemset $b = b_1 \wedge b_2$ is computed through the weighted composition of vectors instead of similarity scores:

$$ES^{\mathrm{c}}(d,z) = \frac{\sum_{b \in r^{\mathrm{c}}(z)} sup(b, D_z^2) \cdot \mathrm{sim}(f(d), g'(b))}{T}, \quad (18)$$

where $sup(b, D_z^2)$ is the support value of the 2-itemset b in D_z^2, and the embedding variant $g'(b)$ can be computed as the weighted sum of the embedding vectors of two conjunctive terms of b, utilizing the 1-itemset support of them in D_z^2:

$$\begin{aligned} g'(b) \quad = \quad & \frac{sup(b_1, D_z^2)}{sup(b_1, D_z^2) + sup(b_2, D_z^2)} \cdot f(\mathcal{T}'(b_1)) \\ & + \frac{sup(b_2, D_z^2)}{sup(b_1, D_z^2) + sup(b_2, D_z^2)} \cdot f(\mathcal{T}'(b_2)). \end{aligned} \quad (19)$$

At last, the embedding-based similarity $ES(d,z)$ between d and z is defined as the maximum value for the two sub-rules, and regarded as the probability $P_2(z|d)$ from the second perspective:

$$P_2(z|d) = ES(d,z) = \max(ES^{\mathrm{d}}(d,z),\ ES^{\mathrm{c}}(d,z)). \quad (20)$$

4.4.3 Word Overlapping-based Similarity Matching Unit. Following the idea of PRBoost [37], besides embedding-based similarity matching in a global view, we also examine the word overlapping-based similarity in a local view, leveraging PLMs' capability of generating signal words from texts once again. In this way, the rule is no longer inserted into the [MASK] token, but occupies the position of the input text in the template as an independent sentence. Consequently, the word-level rule needs to be transformed to a coherent sentence with the help of text connectors. Considering the typical human speech patterns, we use the word "and" instead of "or" to connect indicative words within a rule, regardless of the actual logical operator. Here, the logical relations are reflected by different operations of transformations for the disjunctive and conjunctive sub-rules.

As to the disjunctive sub-rule, the overlapping of strong signal words is computed as:

$$OS^{\mathrm{d}}(d,z) = \frac{SSW(d) \cap SSW(\mathcal{T}(And(r^{\mathrm{d}}(z))))}{K_2}, \quad (21)$$

where $And(\cdot)$ is a transformation function from a logical rule to a sentence, which connects the indicative words of the rule with "and", i.e., $And(r^{\mathrm{d}}(z))$ = "a_1 and a_2 and ... and a_S".

For the conjunctive sub-rule, as the involved indicative words are weaker, the matching process should be more strict. Hence, we divide the sub-rule into two parts alternately, construct the sentences separately, and take the maximum of the similarity scores:

$$\begin{aligned} OS^{\mathrm{c}}(d,z) = \max\Big(& \frac{SSW(d) \cap SSW(\mathcal{T}(And(r^{\mathrm{c}1}(z))))}{K_2}, \\ & \frac{SSW(d) \cap SSW(\mathcal{T}(And(r^{\mathrm{c}2}(z))))}{K_2} \Big), \end{aligned} \quad (22)$$

where $r^{\mathrm{c}1}(z) = \{b_{11}, b_{12}, b_{31}, b_{32}, \ldots\}$ and $r^{\mathrm{c}2}(z) = \{b_{21}, b_{22}, b_{41}, b_{42}, \ldots\}$.

Finally, the similarity of word overlapping between a text d and a category z is defined as the sum over both sub-rules. The corresponding probability from the third perspective is then obtained through the softmax function:

$$OS(d,z) = OS^{\mathrm{d}}(d,z) + OS^{\mathrm{c}}(d,z), \quad (23)$$

$$P_3(z|d) = \frac{\exp(OS(d,z))}{\sum_{z' \in Z} \exp(OS(d,z'))}. \quad (24)$$

To get a final predictive probability, the three scores from different perspectives are averaged together to supplement each other:

$$P(z|d) = (P_1(z|d) + P_2(z|d) + P_3(z|d))/3. \quad (25)$$

Based on this, the pseudo label of a text in the i-th iteration can be assigned to the category with the maximum probability:

$$z^{(i)}(d) = \underset{z}{\mathrm{argmax}}(P(z|d)). \quad (26)$$

Algorithm 1 RulePrompt

Input: An unlabeled text corpus D; a set of categories Z with label names; a pre-trained language model (PLM) M.
Output: The category label $z(d)$ of each text $d \in D$.

```
Obtain initial pseudo labels z^(0)(d) via probability distribution P(z|d) for each text d ∈ D utilizing NPPrompt with Equation 4;
for i = 1 to Iter do
    Obtain the confidence score of each text with Equation 9;
    Obtain strong signal words SSW(d) for each text d ∈ D through the PLM M with Equation 11;
    for all category z ∈ Z do                                   ▷ Rule Mining
        cluster the texts assigned to z into D_z^1, D_z^2, D_z^3 based on their confidence scores;
        Mine 1-itemsets from D_z^1 with Equation 12;
        Mine 2-itemsets from D_z^2 with Equation 14;
        Compose logical rule r^(i)(z) according to Definition 4.1;
    end for
    for all text d ∈ D do                                       ▷ Pseudo Label Generation
        Obtain new pseudo label z^(i)(d) via probability distribution P(z|d) with Equation 25;
    end for
    Fine-tune the PLM M with Equation 27;                       ▷ Fine-Tuning
end for
return z^(Iter)(d);
```

4.5 Self-Supervised Fine-Tuning Module

Although prompting PLMs are strong enough to assist producing classification results in various manners, they are not specially designed for the WSTC task. Therefore, we introduce self-supervised fine-tuning into the closed loop, which uses the PLM's current prediction $P_1(d,z)$ in Equation 16 to refine the PLM itself, gradually enabling it to adapt to the specific task. Concretely, we adopt self-supervised entropy [20] as the loss function to sharpen the probability distribution of category assignments generated by the PLM. That can maximize the potential of PLM and mitigate the accumulation and propagation of errors during the model training process. Given the inaccuracy of pseudo labels, we just select a majority of texts (denoted as D') with tolerable predictive probability for fine-tuning. Formally, the loss is defined as follows:

$$L = \sum_{d \in D' \subset D} \sum_{z \in Z} -P_1(z|d) \log P_1(z|d). \quad (27)$$

Table 1: Dataset Statistics.

Dataset	#Texts	#Categories	Classification Type	Imbalance
AGNews	120000	4	News Topics	1.0
20News	17871	5	News Topics	2.02
NYT	31997	9	News Topics	27.09
IMDB	25000	2	Review Sentiment	1.0

The fine-tuning is conducted after each main iteration updates the pseudo labels of texts, so when the rule mining module is executed in the next iteration, new signal words derived by the fine-tuned PLM can be used. The pseudo-codes of the overall approach is shown in Algorithm 1, and the computational complexity is analyzed in Appendix A.

5 EXPERIMENTS

In this section, we first introduce datasets, baselines and experimental settings in the experiments. Then, overall results are presented to demonstrate the effectiveness and robustness of the proposed approach. Finally, we investigate the importance of key components by ablation study. Due to the page limit, the case study for interpretability analysis is put in Appendix C, and the choices of hyperparameters will be discussed in Appendix D.

The experiments were performed on NVIDIA A40 GPUs, and implemented based on an open-source toolkit OpenPrompt [5]. The dataset links and codes are available on the GitHub[1].

5.1 Experimental Setup

5.1.1 Datasets. We use four popular datasets from various domains for evaluation. The statistics of them are shown in Table 1.

- AGNews [38] is a news article dataset from AG's corpus.
- 20News[2] [12] is a collection of newsgroup documents.
- NYT [38] contains news articles written and published by New York Times, covering abundant news topics.
- IMDB [21] is for sentiment classification of movie reviews.

5.1.2 Baselines. We compare our approach with the following weakly supervised methods. The first two are seed-driven methods, which require at least three keywords for each category as input, and others belong to emerging PLM-based methods.

- **WeSTClass** [23] generates pseudo labels based on word embeddings and obtains the final classifier via self-training.
- **ConWea** [22] acquires pseudo labels based on the contextualized representations of keywords, and trains a text classifier to further expand the keyword sets.
- **LOTClass** [24] utilizes the pre-trained BERT to find indicative keywords, which are directly used for category understanding and feature representation learning.
- **XClass** [31] expands indicative words for category-oriented representations, and generates pseudo labels to fine-tune a text classifier via clustering.
- **ClassKG** [36] builds a keyword graph with co-occurrence relations, and gets pseudo labels through a self-trained subgraph annotator, used to update keywords iteratively.
- **NPPrompt** [41] constructs verbalizers based on initial word embeddings by PLM, and estimates the probability distribution over categories via weighted sum of these words.
- **PIEClass** [40] utilizes zero-shot prompting to generate pseudo labels and improves the quality of them through two fine-tuning strategies of PLMs.

Besides, we also inspect a fully supervised method, which uses the BERT classifier with fine-tuning based on the labels in the training set. It can be regarded as an upper-bound for WSTC methods.

5.1.3 Experimental Settings. We use the standard label name of each category for each dataset as input. As prompt-based methods are relatively robust with PLMs [32], we follow previous work [30, 41] to choose RoBERTa-large [18] as our PLM. The number of full iterations $Iter$ is unified to 3 across all datasets. To save space, we detail other settings and hyperparameters in Appendix B.

As usual, we use Micro-F1 and Macro-F1 as the evaluation metrics. The results of baselines are quoted from [37] with missing values marked as "-". Since NPPrompt uses more than one keyword on some datasets in its original setting, we re-run its codes provided by authors[3] using only the label names for fair comparison.

5.2 Overall Results

The overall results of RulePrompt, its variant without fine-tuning, and baseline methods are shown in Table 2.

It is evident that our model consistently outperforms baselines for all datasets, and almost catch up with the supervised methods on IMDB. That certifies the role of logical rules of categories in assisting prompting PLMs to understand the topics of texts, compared with independent category-indicative words. In addition, the advantage over PIEClass highlights the importance of the mutual enhancement of pseudo labels and logical rules, as they are both imperfect at the starting point. Although RulePrompt exhibits a slight gap with PIEClass on the Macro-F1 metrics in the imbalanced NYT dataset, which is probably caused by the amplification of categories with small samples, our approach is more stable across all datasets.

What is more, RulePrompt significantly enhances classification accuracy on the 20News dataset, where some categories are not completely disjoint and some label assignments are even inconsistent with general knowledge. For example, as discussed in prior work [35], this dataset combines "science" and "encryption" into one category, while placing "computer" in a separate class. However intuitively, "encryption" is supposed to fall into the domain of "science", where "computer" is considered as another subset. That suggests our approach can effectively fuse the general knowledge embodied in the prompting PLMs and the special characteristics of the target dataset, through expressive logical rules and self-supervised fine-tuning, making it more suitable for classifying texts in challenging tasks with overlapping and counter-intuitive categories.

Besides, the promotion over the variant without fine-tuning indicates that when there are sufficient evidences available for each category, even if unlabeled, it is still feasible to refine the PLM to accommodate the specific task and dataset, with the help of self-iterative logical knowledge of categories. However for the

[1] https://github.com/MiaomiaoLi2/RulePrompt
[2] http://qwone.com/~jason/20Newsgroups/
[3] https://github.com/XuandongZhao/NPPrompt

Table 2: Overall Results on Four Datasets by Two Metrics. The Best Scores of Weakly Supervised Methods are Marked in Bold.

Method	AGNews		20News		NYT		IMDB	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
WeSTClass	0.823	0.821	0.713	0.699	0.683	0.570	0.774	-
ConWea	0.746	0.742	0.757	0.733	0.817	0.715	-	-
LOTClass	0.869	0.868	0.738	0.725	0.671	0.436	0.865	-
XClass	0.857	0.857	0.786	0.778	0.790	0.686	-	-
ClassKG	0.881	0.881	0.811	0.820	0.721	0.658	0.888	0.888
NPPrompt	0.692	0.628	0.663	0.660	0.768	0.591	0.941	0.941
PIEClass (RoBERTa+RoBERTa)	0.895	0.895	0.755	0.760	0.760	0.694	0.906	0.906
PIEClass (ELECTRA+ELECTRA)	0.884	0.884	0.816	0.817	0.832	**0.763**	0.931	0.931
RulePrompt without Fine-Tuning	0.843	0.838	0.706	0.700	0.821	0.690	**0.943**	**0.943**
RulePrompt	**0.897**	**0.896**	**0.831**	**0.829**	**0.833**	0.716	**0.943**	**0.943**
Fully Supervised	0.940	0.940	0.965	0.964	0.943	0.899	0.945	-

Table 3: Results of Ablation Study for One Iteration. The Best Scores are Marked in Bold.

Method	AGNews		20News		NYT		IMDB	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
RulePrompt-1 (−Conj**)**	0.853	0.850	0.702	0.694	0.823	0.709	0.938	0.938
RulePrompt-1 ($-D_z$**)**	0.497	0.414	0.637	0.613	0.749	0.650	0.933	0.933
RulePrompt-1 ($-U1$**)**	0.760	0.759	0.647	0.639	0.583	0.530	0.913	0.913
RulePrompt-1 ($-U2$**)**	0.853	0.850	0.695	0.683	0.818	0.698	0.937	0.937
RulePrompt-1 ($-U3$**)**	0.852	0.849	0.700	0.691	0.822	0.708	0.935	0.935
RulePrompt-1	**0.854**	**0.851**	**0.705**	**0.699**	**0.825**	**0.712**	**0.941**	**0.941**

relatively simple and small IMDB dataset, adopting a fixed PLM can also achieve comparable performances.

5.3 Ablation Study

The ablation results for the two main modules are shown in Table 3. In order to make the role of each component more prominent, the experiments were carried out in the first iteration (denoted as RulePrompt-1), i.e., without self-supervised fine-tuning.

In terms of rule mining. The variants include removing the conjunctive sub-rule (−Conj), and mining rules from all texts without clustering-based set division ($-D_z$). At first, the lack of conjunction part will lower the performance. That confirms the discrepancy among indicative words on characterizing category meanings, and the combined effect of relatively weaker words cannot be neglected. Besides, when the rules are mined from the whole corpus, the accuracy is distinctly declined. That can be attributed to the inaccurate pseudo labels which contaminate the mining object. Therefore, the confidence scores for the predicted labels are vital to help choose appropriate texts to search for rules in an adaptive way.

In terms of rule-enhanced pseudo label generation. The variants contain the methods without either of the three units respectively. For all cases, the full approach performs the best. That reflects different capabilities of the PLM as well as the different manners of logical rules enhancing the PLM. Since it is hard to decide which is the best one for a specific task beforehand, averaging the predictive results of them to supplement each other is a good choice, especially in the weakly supervised setting.

6 CONCLUSION

Addressing the limitations of relying solely on seed words (label names) for supervision in weakly supervised text classification task, this paper explores a kind of novel knowledge representation to characterize category meanings, which facilitates the effective integration of knowledge and unlabeled corpus. The proposed logical rules for categories can be automatically mined based on the pseudo labels of texts and iteratively self-optimized through mutual enhancement with them. Thanks to the enriched symbolic knowledge, the potential of prompting PLMs are further exploited in terms of generative capability and semantic representations, which is realized by incorporating the PLM into the rule-based iteration process. With this framework, RulePrompt exceeds the SOTA weakly supervised methods, and the logical rules we extract are intuitive and provide valuable guidance by disambiguating easily-confused categories.

For future work, we will strengthen the expressiveness of the category rules, such as adding the negation operator to better avoid category confusions. Additionally, more effective iteration strategies are also worth studying, and the manner of iteratively updating pseudo labels and logical rules can be applied in other prompting PLM-based scenarios.

ACKNOWLEDGMENTS

This work is supported by CAS Project for Young Scientists in Basic Research (YSBR-040) and Strategic Priority Research Program of Chinese Academy of Sciences (XDC02060500).

REFERENCES

[1] Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*. 487–499.

[2] Samuel Budd, Emma C Robinson, and Bernhard Kainz. 2021. A survey on active learning and human-in-the-loop deep learning for medical image analysis. *Medical Image Analysis* 71 (2021), 102062.

[3] Ming-Wei Chang, Lev Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: dataless classification. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 830–835.

[4] Xingyuan Chen, Yunqing Xia, Peng Jin, and John Carroll. 2015. Dataless text classification with descriptive LDA. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 2224–2231.

[5] Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. 2022. OpenPrompt: An Open-source Framework for Prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. 105–113.

[6] Yu Fei, Zhao Meng, Ping Nie, Roger Wattenhofer, and Mrinmaya Sachan. 2022. Beyond prompting: Making Pre-trained Language Models Better Zero-shot Learners by Clustering Representations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 8560–8579.

[7] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (ACL)*. 6894–6910.

[8] Jiawei Han, Jian Pei, Behzad Mortazavi-Asl, Qiming Chen, Umeshwar Dayal, and Meichun Hsu. 2000. FreeSpan: Frequent pattern-projected sequential pattern mining. In *Proceedings of the sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, Raghu Ramakrishnan, Salvatore J. Stolfo, Roberto J. Bayardo, and Ismail Parsa (Eds.). ACM, 355–359.

[9] Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2022. PTR: Prompt tuning with rules for text classification. *AI Open* 3 (2022), 182–192.

[10] Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, and Maosong Sun. 2022. Knowledgeable Prompt-tuning: Incorporating Knowledge into Prompt Verbalizer for Text Classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL)*. 2225–2240.

[11] Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. 2016. Harnessing Deep Neural Networks with Logic Rules. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL)*. 2410–2420.

[12] Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Machine Learning Proceedings 1995*. Elsevier, 331–339.

[13] Chenliang Li, Shiqian Chen, and Yan Qi. 2019. Filtering and classifying relevant short text with a few seed words. *Data and Information Management* 3, 3 (2019), 165–186.

[14] Chenliang Li, Jian Xing, Aixin Sun, and Zongyang Ma. 2016. Effective document labeling with very few seed words: A topic model approach. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management (CIKM)*. ACM, 85–94.

[15] Jiacheng Li, Haibo Ding, Jingbo Shang, Julian McAuley, and Zhe Feng. 2021. Weakly Supervised Named Entity Tagging with Learnable Logical Rules. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (ACL)*. Online, 4568–4581.

[16] Miaomiao Li, Jiaqi Zhu, Xin Yang, Yi Yang, Qiang Gao, and Hongan Wang. 2023. CL-WSTC: Continual Learning for Weakly Supervised Text Classification on the Internet. In *Proceedings of the ACM Web Conference 2023 (WWW)*. ACM, 1489–1499.

[17] Chonghua Liao, Yanan Zheng, and Zhilin Yang. 2022. Zero-Label Prompt Selection. *arXiv preprint arXiv:2211.04668* (2022).

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

[19] Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

[20] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)(ACL)*. 8086–8098.

[21] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. 142–150.

[22] 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 (ACL)*. 323–333.

[23] 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)*. ACM, 983–992.

[24] 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)*. 9006–9017.

[25] Seongmin Park, Kyungho Kim, and Jihwa Lee. 2023. Cross-task Knowledge Transfer for Extremely Weakly Supervised Text Classification. In *Findings of the Association for Computational Linguistics: ACL 2023*. 5329–5341.

[26] Seongmin Park and Jihwa Lee. 2022. LIME: Weakly-Supervised Text Classification without Seeds. In *Proceedings of the 29th International Conference on Computational Linguistics*. 1083–1088.

[27] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Meichun Hsu. 2004. Mining Sequential Patterns by Pattern-Growth: The PrefixSpan Approach. *IEEE Transactions on Knowledge and Data Engineering* 16, 11 (2004), 1424–1440.

[28] Tobias Scheffer, Christian Decomain, and Stefan Wrobel. 2001. Active Hidden Markov Models for Information Extraction. In *Proceedings of the 4th International Conference on Advances in Intelligent Data Analysis*. 309–318.

[29] Yangqiu Song and Dan Roth. 2014. On dataless hierarchical text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. 1579–1585.

[30] Yau-Shian Wang, Ta-Chung Chi, Ruohong Zhang, and Yiming Yang. 2023. PESCO: Prompt-enhanced Self Contrastive Learning for Zero-shot Text Classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL)*. 14897–14911.

[31] 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*. 3043–3053.

[32] Zihan Wang, Tianle Wang, Dheeraj Mekala, and Jingbo Shang. 2023. A Benchmark on Extremely Weakly Supervised Text Classification: Reconcile Seed Matching and Prompting Approaches. In *Findings of the Association for Computational Linguistics: ACL 2023*. 3944–3962.

[33] Yi Yang, Hongan Wang, Jiaqi Zhu, Wandong Shi, Wenli Guo, and Jiawen Zhang. 2021. Effective seed-guided topic labeling for dataless hierarchical short text classification. In *International Conference on Web Engineering (ICWE)*. 271–285.

[34] Yi Yang, Hongan Wang, Jiaqi Zhu, Yunkun Wu, Kailong Jiang, Wenli Guo, and Wandong Shi. 2021. Dataless short text classification based on biterm topic model and word embeddings. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI)*. 3969–3975.

[35] 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. In *Findings of the Association for Computational Linguistics: NAACL*. 2295–2305.

[36] Lu Zhang, Jiandong Ding, Yi Xu, Yingyao Liu, and Shuigeng Zhou. 2021. Weakly-supervised text classification based on keyword graph. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2803–2813.

[37] Rongzhi Zhang, Yue Yu, Pranav Shetty, Le Song, and Chao Zhang. 2022. 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) (ACL)*. 745–758.

[38] Xiang Zhang, Junbo Zhao, and Yann Lecun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 28rd International Conference on Neural Information Processing Systems (NIPS)*. 649–657.

[39] Yu Zhang, Shweta Garg, Yu Meng, Xiusi Chen, and Jiawei Han. 2022. Motifclass: Weakly supervised text classification with higher-order metadata information. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining (WSDM)*. ACM, 1357–1367.

[40] Yunyi Zhang, Minhao Jiang, Yu Meng, Yu Zhang, and Jiawei Han. 2023. PIEClass: Weakly-Supervised Text Classification with Prompting and Noise-Robust Iterative Ensemble Training. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACM, 12655–12670.

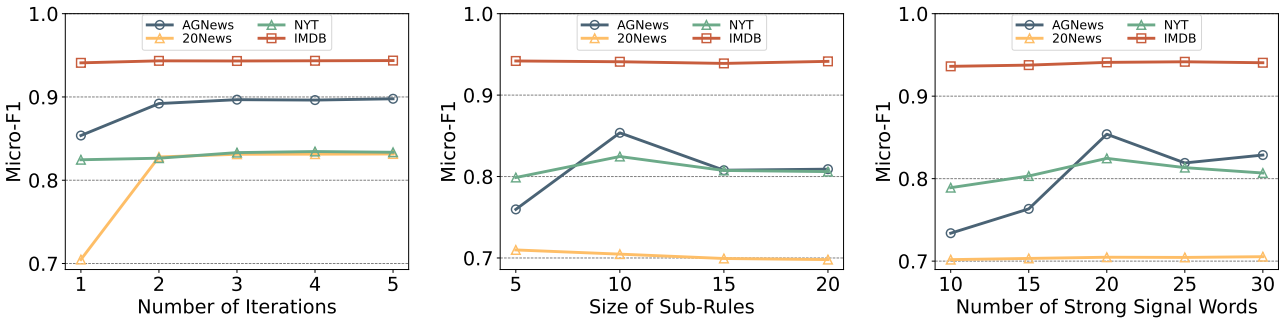
[41] Xuandong Zhao, Siqi Ouyang, Zhiguo Yu, Ming Wu, and Lei Li. 2023. Pre-trained Language Models Can be Fully Zero-Shot Learners. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (ACL)*. 15590–15606.

A COMPUTATIONAL COMPLEXITY

On the basis of Algorithm 1, we reckon the computational complexity of RulePrompt, assuming N is the number of texts to be classified. At first, the initial pseudo labels of texts can be obtained in $O(N)$ (line 1). Then, inside each iteration, the confidence scores and the strong signal words of these texts are got in $O(N)$ (lines

Table 4: Label Names and Templates for RulePrompt.

Dataset	Label Names	Template
AGNews	politics, sports, business, technology	A [MASK] news: d
20News	computer, sports, science, politics, religion	A [MASK] news: d
NYT	business, politics, sports, health, education, estate, arts, science, technology	Topic: [MASK] d
IMDB	good, bad	d In summary, the film was [MASK].



(a) Varied Number of Iterations (b) Varied Size of Sub-Rules (c) Varied Number of Strong Signal Words

Figure 3: Results with Varied Hyperparameters.

3-4), and for each category, the clustering process also takes $O(N)$ (line 6). For the frequent itemset mining, since the number of items (words) is up to $O(N \cdot K_2)$, with the classical Apriori algorithm, the frequent 1-itemsets and 2-itemsets can be obtained in $O((N \cdot K_2)^2)$ (line 7) and $O((N \cdot K_2)^3)$ (line 8) respectively, and we do not need to mine long patterns. The updated pseudo labels are computed in $O(N)$ (line 12) and the fine-tuning process takes $O(N)$ with bounded text sizes. As the number threshold K_2 of top strong signal words, the number of categories K and the iteration number $Iter$ are all constants, the overall computational complexity can be estimated as $O(N^3)$. Therefore, the proposed approach RulePrompt is scalable for larger datasets. Moreover, since the size of the sub-rules is fixed in our approach ($S = T = 10$), more complex rules would not bring greater computational complexity.

B DETAILED EXPERIMENTAL SETTINGS

For a fair comparison, we use the same label names of categories for each dataset as used and reported previously. Meanwhile, based on the characteristics of these datasets, we employ suitable templates according to the previous work, and list them as well as label names in Table 4.

With regard to getting signal words and strong signal words of texts, we set $K_1 = 100$ and $K_2 = 20$. In the process of frequent pattern mining, we set support thresholds $h_1 = h_2 = 0.05$ for the imbalanced NYT, and 0.1 for the other three datasets. As only top 1-itemsets and 2-itemsets can enter the rule, these thresholds are insensitive and can thus be a low value. The maximum numbers of terms in the disjunctive sub-rule and conjunctions in the conjunctive sub-rule are both $S = T = 10$. For the more complicated 20News, we only use the word with the highest support value in the disjunctive sub-rule instead of the first half, to meet the stricter requirement for verbalizers. In the embedding-based similarity matching unit, we choose Roberta-SimCSE [7] as the sentence encoder to realize the function $f(.)$.

As for the self-supervised fine-tuning process, we train 7 epochs in each iteration, except for 20News which needs more training to understand categories, so the number of epochs is set as 30. The learning rate is 1e-8 for AGNews and 20News, while 1e-9 and 1e-10 for NYT and IMDB respectively. The maximum sequence length is specified to 150 for AGNews and NYT, but 500 for 20News and IMDB. We use AdamW [19] as the optimizer. Besides, the proportion of texts used for fine-tuning (D') is set to 90% for the imbalanced NYT, and 85% for the other three datasets.

C CASE STUDY

To analyze the interpretability of logical rules derived by RulePrompt, we observe that for the "Arts" category in the NYT dataset, "art", "museums", "galleries", "artwork" and "cultural" are mined as 1-itemsets to constitute the disjunctive sub-rule. While, "ballet ∧ dancing" and "dancers ∧ theater" are identified as 2-itemsets. These paired words can indeed complement each other and form the conjunctive sub-rule. These rules align with common intuitions and significantly contribute to a more comprehensive representation of respective categories. Moreover, the word "architecture" is found within the rules associated with two different categories: "Estate" and "Arts". It is paired with "residential" and "apartments" for the former, but "museum" and "cultural" for the latter. That exemplifies the ability of our approach to disambiguate easily-confused categories with polysemous words.

Furthermore, although the initial predictions may be incorrect, it is still beneficial for the subsequent rule mining process. Through the clustering of texts based on the pseudo labels with confidence scores, we can find appropriate strong signal words as well as their patterns to compose the rules for characterizing category meanings. During several iterations, the rules and the predictions will be optimized in the manner of mutual enhancement. Taking the "Business"

category in the AGNews dataset for example, "x ∧ bloomberg" is initially mined as one of the 2-itemsets within one iteration, which is not very meaningful, but after three iterations, two new 2-itemsets "economic ∧ company" and "corporate ∧ econom" are mined replacing the previous one, which are more informative and consistent with the category.

D HYPERPARAMETER ANALYSIS

In this section, we pay attention to the key hyperparameters in our approach, such as the iteration number, the sub-rule size, and the number of top strong signal words, to certify the robustness of RulePrompt.

D.1 Number of Iterations

We vary the number of full iterations $Iter$ from 1 to 5 for all datasets, and Figure 3(a) shows the respective Micro-F1 values. It can be seen that the performance shows a trend of first rising and then stabilizing after about three iterations. That is consistent across all datasets, and thus verifies our approach is insensitive on this setting as long as at least three iterations are fulfilled.

D.2 Size of Sub-Rules

We make the maximum number of terms in the disjunctive sub-rule (S) and the maximum number of conjunctions in the conjunctive sub-rule (T) equal to each other, and vary them together between 5 and 20 with the step size 5. The results in Figure 3(b) show a trend of first rising and then declining, with an optimal value of 10, which is nearly consistent for all datasets.

An exception appears for the complicated dataset 20News, where the accuracy always decreases mildly along with the increase of sub-rule sizes. This is mainly because some categories in 20News are largely overlapping, and the stricter rules are thereby required to distinguish them.

D.3 Number of Strong Signal Words

We change the number threshold K_2 of top strong signal words between 10 and 30 with the step size 5. The performances are shown in Figure 3(c). Again, an almost optimal value 20 is reached for all datasets with a similar trend as the size of sub-rules.

Note that another threshold K_1 is used to determine a larger candidate signal word set, so does not directly affect the results and behaves insensitive obviously.