SciPrompt: Knowledge-augmented Prompting for Fine-grained Categorization of Scientific Topics

Zhiwen You[1], **Kanyao Han**[1], **Haotian Zhu**[2], **Bertram Ludäscher**[1], **Jana Diesner**[1,3]
[1] University of Illinois Urbana-Champaign
[2] University of Washington
[3] Technical University of Munich
{zhiweny2, kanyaoh2, ludaesch}@illinois.edu
haz060@uw.edu jana.diesner@tum.de

Abstract

Prompt-based fine-tuning has become an essential method for eliciting information encoded in pre-trained language models for a variety of tasks, including text classification. For multi-class classification tasks, prompt-based fine-tuning under low-resource scenarios has resulted in performance levels comparable to those of fully fine-tuning methods. Previous studies have used crafted prompt templates and verbalizers, mapping from the label terms space to the class space, to solve the classification problem as a masked language modeling task. However, cross-domain and fine-grained prompt-based fine-tuning with an automatically enriched verbalizer remains unexplored, mainly due to the difficulty and costs of manually selecting domain label terms for the verbalizer, which requires humans with domain expertise. To address this challenge, we introduce SciPrompt, a framework designed to automatically retrieve scientific topic-related terms for low-resource text classification tasks. To this end, we select semantically correlated and domain-specific label terms within the context of scientific literature for verbalizer augmentation. Furthermore, we propose a new verbalization strategy that uses correlation scores as additional weights to enhance the prediction performance of the language model during model tuning. Our method outperforms state-of-the-art, prompt-based fine-tuning methods on scientific text classification tasks under few and zero-shot settings, especially in classifying fine-grained and emerging scientific topics[1].

1 Introduction

Scientific text classification tasks involve categorizing scientific abstracts into specific disciplines or topics. Recent studies leverage prompt-based fine-tuning method (Ding et al., 2022a; Gu et al., 2022; Schick and Schütze, 2020; Liu et al., 2023a), transferring the text classification problem as a masked language modeling task. Masked Language Models (MLMs) are developed by extensively training on large text corpora with a percentage of the input tokens being randomly replaced with a `[MASK]` token. Traditional fine-tuning, which requires additional training on labeled domain- or task-specific data (Ovadia et al., 2023), may not be suitable in limited data scenarios, such as few and zero-shot settings. Prompt-based fine-tuning has emerged as an effective alternative. This approach uses a prompt to guide the MLM in generating a specific token through masking a `[MASK]` token in the prompt template, addressing the text classification tasks (Schick and Schütze, 2020; Hu et al., 2021; Chen et al., 2022b; Gao et al., 2021a) under low-resource conditions (i.e., few and zero-shot settings) through a verbalizer. As defined by Schick and Schütze (2020), the verbalizer refers to the mapping from label words (e.g., "cryptanalysis") to the corresponding class (e.g., "Cryptography"), serving as a projection function between the vocabulary and the class label space. However, in the context of classifying scientific literature, the complexity of scientific language and scarcity of fine-grained (i.e., a wide range of scientific fields that are labeled with sub-categories) or emerging topics make it hard to automatically classify cross-domain scholarly articles with limited training samples and manually created verbalizers (Schick and Schütze, 2020).

The goal of this paper is to address the challenge of multi-class classification in low-resource settings, specifically focusing on classifying scientific abstracts into different domains with only a limited number of labeled examples. We introduce a prompt-based fine-tuning approach enriched with domain knowledge as a new strategy for retrieving domain-adaptive label terms (i.e., scientific terms in various fields) without manual in-

[1]Our code is available at https://github.com/zhiwenyou103/SciPrompt.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 6087–6104
November 12-16, 2024

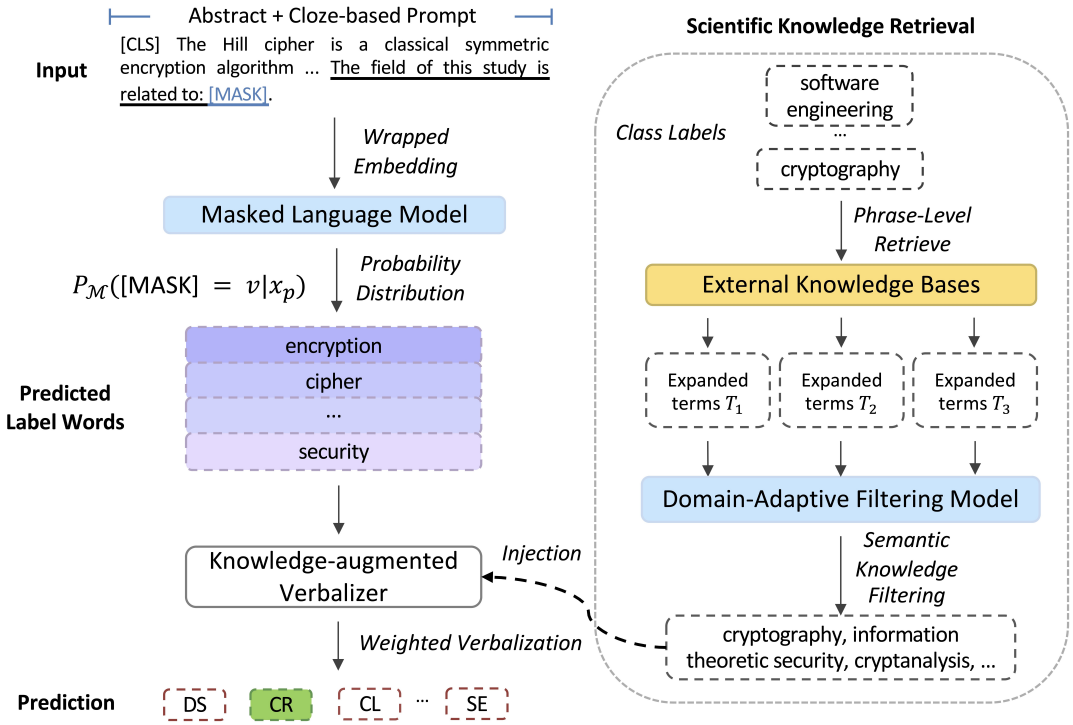

Figure 1: Overall framework of SciPrompt. The left side shows the overall process of masked language modeling for performing the text classification task. The right side shows our proposed knowledge retrieval and domain-adaptive filtering phase (§3). The prediction results, such as CR and SE, correspond to the class labels for Cryptography and Software Engineering, respectively, and are used for scientific knowledge retrieval.

tervention. We enhance our approach for low-resource scenarios by retrieving scientific phrases from external knowledge bases (KBs) to expand label terms of the verbalizer (Hu et al., 2021) from the token-level to term phrases. We fine-tune Natural Language Inference (NLI) models for semantic similarity search between retrieved label terms and class labels to select domain-related scientific phrases. Our method differs from previous studies (Hu et al., 2021; Ding et al., 2022b), which rely on word frequency filtering and are limited to single-token verbalizer projection for text classification. Given the complexity of scientific terminology (see Appendix B for more details), we refine the traditional verbalization approach (Ding et al., 2022a) by integrating scientific terms through deploying a weight-aware label term mapping function. This approach improves the projection performance from MLM's predictions to probabilities of a specific class compared with prior studies (Hu et al., 2021; Gao et al., 2021b; Chen et al., 2022a).

Our approach consists of three stages: 1) retrieval of scientific terms, 2) label term filtering, and 3) prediction of scientific topics. Initially, we use a cloze-style prompt and an input scientific abstract to guide the MLM to generate label words to fill the `[MASK]` token (Figure 1). Then, we use each class label as a query to retrieve class-related domain phrases (also denote as "label terms") from external KBs. To filter the potentially irrelevant terms gathered in the retrieval stage, we fine-tune both bi-encoder and cross-encoder models using the SciNLI dataset (Sadat and Caragea, 2022), enabling the selection of the most relevant domain phrases. Finally, with the selected sets of knowledge-enriched scientific terms, we incorporate these label terms into the verbalizer to convert the MLM's prediction into a specific class through a semantic score-weighted average loss, enhancing the precision of the probability projections for the augmented verbalizer. Our method extends beyond token-to-token verbalization by encompassing token-to-phrase verbalization that enriches the semantic meaning of scientific domain vocabulary. This broader scope allows for an advanced interpretation of scientific language and classifying emerging topics under weak supervision.

In summary, our contributions are the presentation of:

- A domain-adaptive prompt-based fine-tuning framework, named SciPrompt, for fine-

grained and low-resource scientific text classification tasks.

- A new knowledge retrieval and filtering strategy for automatically enriching the verbalizer with domain knowledge.
- A weighted verbalization approach tailored for mapping filtered scientific label terms from model predictions to specific classes.
- Evaluation via experiments on four scientific datasets show that SciPrompt largely outperforms most state-of-the-art methods in few and zero-shot settings.

2 Related Work

2.1 Knowledge-Powered Prompting for Text Classification

A Pattern-Exploiting Training (PET) framework (Schick and Schütze, 2021a,b), which initially investigated how cloze-based prompt templates can guide language models to tackle classification tasks (Han et al., 2022; Ding et al., 2022b; Min et al., 2022; Wang et al., 2022a; Zhang et al., 2022; Wang et al., 2022b), has inspired research on incorporation more diverse label words into the verbalizer. Specifically, Hu et al. (2021) added external knowledge to the verbalizing process to help an MLM predict masked tokens more accurately. AdaPrompt (Chen et al., 2022b) applied a different knowledge injection method that leveraged task and prompt characteristics to retrieve external knowledge for continuous pre-training of MLMs adaptively. However, classifying scientific literature presents challenges that previous methods have not addressed, including projecting phrase-level label terms in the verbalization process. Other challenges, to which a broad range of solutions have been developed, include handling complex semantic structures in a wide range of scientific topics (Eykens et al., 2021; Khadhraoui et al., 2022) and the scarcity or imbalance of labeled data across multiple disciplines (Cunha et al., 2021).

2.2 Label Terms Refinement

Prior research on prompt-based fine-tuning has used the verbalizer module to map MLM's predictions to specific classes. Schick and Schütze (2021a) introduced an automatic verbalizer search that identifies suitable label words from training data and language models to enrich the verbalizer. This approach has been further explored in different studies to improve the classification performance (Gao et al., 2021a; Shin et al., 2020; Liu et al., 2023b), although these methods typically need extensive training data, making them less suitable for low-resource scenarios. To address these challenges, one can manually expanding the verbalizer with more label words (Shin et al., 2020), which has limitations when classifying fine-grained and domain-related categories that need expert knowledge. Recently, external KBs have been used to enrich the verbalizer by sourcing class-related label words (Hu et al., 2021; Chen et al., 2022b).

3 Methodology

Our framework of SciPrompt uses a two-stage approach for scientific text classification: (1) masked language modeling and (2) domain knowledge retrieval and filtering.

3.1 Cloze-Style Masked Language Modeling

MLMs $\mathcal{M}$ (e.g., SciBERT (Beltagy et al., 2019)) are created by randomly masking tokens in the training text and training the model to predict the masked tokens. Similarly, prompt-based fine-tuning typically leverages a cloze- or prefix-based prompt template, reformulating the input into a masked language modeling task. This strategy enables $\mathcal{M}$ to predict the masked token, facilitating the execution of downstream tasks based on $\mathcal{M}$ outputs. Building upon Hu et al. (2021), our framework employs a few-shot prompt-based fine-tuning strategy that conceptualizes scientific text classification as an N-way K-shot task, where N indicates the number of classes and K is the number of labeled examples per class.

We provide a limited number of labeled examples for each class to tune $\mathcal{M}$. We construct a training $\mathcal{D}_{train}$ and validation set $\mathcal{D}_{val}$ following previous studies (Gao et al., 2021a; Perez et al., 2021; Wang et al., 2022a; Hu et al., 2021) with n examples per class. For the few-shot setting, given a cloze-based prompt template $\mathcal{P}_t$ and an input abstract a_n, where $a_n \in \mathcal{D}_{train}$, $\mathcal{M}$ predicts the label word l to fill into the `[MASK]` position in the prompt template. After that, the verbalizer function f_v maps the predicted label word l onto pre-defined label term set $\mathcal{L}$ to classify it into a class, i.e., $\mathcal{L} \rightarrow \mathcal{Y}$. We use a cross-entropy loss (Gao et al., 2021a) to update the parameters of $\mathcal{M}$ through verbalization outputs. For instance, the

prompt is designed as "`[Abstract]`. The field of this article is related to: `[MASK]`". $\mathcal{M}$ will predict suitable label word l to fill into the `[MASK]`. Then, f_v calculates the probability of classifying l into a topic y_i, where $y_i \in \mathcal{Y}$:

$$P(y_i \mid a_n) = f_v(P(\texttt{[MASK]} = \mathcal{M}(l) \mid a_n)), \quad (1)$$

where $l \in \mathcal{L}$. In the zero-shot setting, given $\mathcal{M}$ can directly generate a label word to fill into `[MASK]`, we use the output of $\mathcal{M}$ as the final label word and send the output into the verbalizing function to calculate class probabilities without tuning loss updates.

3.2 Scientific Knowledge Retrieval

Predicting masked tokens using an MLM involves generating a range of potential label words, each with varying probabilities of matching a specific class. Enhancing the verbalizer with a more extensive set of label terms has been proven to improve the accuracy of word-to-class mapping (Hu et al., 2021; Chen et al., 2022b; Wang et al., 2022a; Shin et al., 2020). To implement this approach, we use two external KBs, `Related Words`[2] and `Reverse Dictionary`[3] for scientific knowledge retrieval. `Related Words` identifies relevant terms using vector similarity and resources like word embeddings and ConceptNet. `Reverse Dictionary`, which acts as a word search engine, finds terms based on definitions or phrases. `Reverse Dictionary` is particularly useful in phrase-level retrieval, where straightforward labels from `Related Words` may not suffice given a domain-specific phrase (e.g., Networking and Internet Architecture). We set class labels $C = \{y_1, y_2, ..., y_n\}$ as queries to retrieve from `Related Words` $\mathcal{G}_{RW}$.

When $\mathcal{G}_{RW}$ fails to produce terms with similarity scores above zero, we use `Reverse Dictionary`, denoted as $\mathcal{G}_{RD}$, for additional phrase retrieval. Each retrieved term is assigned a single relevance score. Initially, we adopted the same threshold (i.e., threshold = 0) as KPT (Hu et al., 2021) for term retrieval based on topic names. Subsequently, we impose two additional thresholds for further selection of retrieved terms (§3.3). Utilizing these KBs enables the compilation of knowledge-enhanced term sets $\mathcal{T}_i = t_1, t_2, ..., t_m$ for each dataset, where $i \in n$ and t represents the retrieved label terms. Note that the number of terms m may vary for each class.

3.3 Domain Adaptive Model Tuning

To effectively identify the most relevant label words for each class from a set of initial raw terms, it is crucial to use a model tailored or adaptable to specific fields. Drawing from Chen et al. (2022b), who employed a pre-trained NLI model to filter label words produced by an MLM, we present a method that enhances the accuracy of selecting label terms related to specific topics by integrating domain knowledge. We apply a newly introduced scientific NLI dataset $\mathcal{D}_{SciNLI}$ (Sadat and Caragea, 2022), consisting of labeled sentence pairs (s_i, s_j) from scholarly articles in the fields of NLP and computational linguistics. This dataset serves to fine-tune both cross-encoder $\mathcal{M}ce$ and bi-encoder $\mathcal{M}_{be}$ NLI models[4], where $\mathcal{M}_{be}$ produces for a given sentence a sentence embedding and $\mathcal{M}_{ce}$ passes a sentence pair to the encoder to produce an output value between 0 and 1 indicating the similarity of the input sentence pair (Reimers and Gurevych, 2019). The training labels are defined as "*Entailment*" or "*Contradiction*", thus framing the model fine-tuning as a binary classification task:

$$\mathcal{M}'(s_i, s_j) = \begin{cases} > 0 & \text{if } s_i \text{ entails } s_j \\ < 0 & \text{if } s_i \text{ contradicts } s_j \end{cases},$$

where $\mathcal{M}'$ denotes either $\mathcal{M}_{ce}$ or $\mathcal{M}_{be}$.

3.4 Semantic Knowledge Filtering

We merge each retrieved scientific label term with a standard prompt (see Appendix G), encode prompts using the fine-tuned $\mathcal{M}_{be}$, and use these encoded embeddings as queries for sentence-level semantic searches to select topic-related label terms and calculate semantic similarity scores w_l for each label term. We apply SentenceTransformers[5] to conduct the cosine similarity search using $\mathcal{M}_{be}$ within each retrieved label term set. Then, we use $\mathcal{M}_{ce}$ to rerank these label terms for every prompt pair of each topic, selecting relevant sentences based on predefined thresholds ($\mu_{be} = 0.5$, $\mu_{ce} = 0.1$). As these scores also help predict label words, we apply this method in few and zero-shot scenarios (for more details, see Appendix F).

Following KPT (Hu et al., 2021), we also apply a label term calibration approach with a full training

[2] https://relatedwords.org
[3] https://reversedictionary.org
[4] https://www.sbert.net/examples/applications/cross-encoder/
[5] https://www.sbert.net/index.html

set to directly remove irrelevant label terms in the verbalizer that are less likely to be predicted by $\mathcal{M}$. The retrieved label terms for each class with lower probabilities (i.e., less than 0.5) produced by $\mathcal{M}$ are removed. The probability of t is:

$$\hat{P}_{\mathcal{M}}(\texttt{[MASK]} = t|a_n) \propto \frac{P_{\mathcal{M}}(\texttt{[MASK]} = t|a_n)}{prior(p_t)}, \tag{2}$$

where $prior(p_t)$ is the prior probability of the label term t produced by $\mathcal{M}$ using the training set.

3.5 Weighted Verbalizer Transformation

Given that retrieved label terms may be tokenized into multiple tokens, we adopt a "mean" method to average the tokens of a label term (Ding et al., 2022b), considering all parts of a term as significant.

Adopting the verbalizer structure from Ding et al. (2022b), we introduce a verbalization approach that maps $\mathcal{M}$'s output to specific classes y_i, using predefined semantic scores w_l as weights for each label term. This method aims to enhance the accuracy of classifying $\mathcal{M}$'s predictions l into topic y_i:

$$\begin{aligned} P(y_i|a_n) &= \arg\max_{y_i \in \mathcal{Y}} s(v_{y_i}|h_{mask}, w_l) \\ &= \frac{\exp\left(v_{y_i} \cdot h_{mask} \cdot w_l\right)}{\sum_{y \in \mathcal{Y}} \exp\left(v_y \cdot h_{mask} \cdot w_l\right)}, \end{aligned} \tag{3}$$

where the objective function $s(v_{y_i}|h_{mask}, w_l)$ calculates $\mathcal{M}$'s probability for the output v_{y_i} of the `[MASK]` token, with v_{y_i} as the label term embeddings, and h_{mask} as the hidden states at the `[MASK]` position. This objective function can be optimized through the cross-entropy loss as denoted in Equation (1).

3.6 Vector-Based Verbalizer Mapping

Incorporating the filtered label terms into the verbalizer is crucial for making accurate predictions and eliminating noise simultaneously. Moving beyond simple summing (Wang et al., 2022a) or weighted averaging (Hu et al., 2021) of label words, the Word-level Adversarial ReProgramming (WARP) model introduced in (Hambardzumyan et al., 2021) uses vector representations for class mapping, which is distinct from conventional single word projection. We introduce a new method named SciPrompt$_{Soft}$ based on the uniqueness of our phrase-level verbalizer. Specifically, we refine the verbalization in SciPrompt$_{Soft}$ by drawing from the soft verbalizer concept introduced by WARP. In the experiments with SciPrompt$_{Soft}$, we aggregate all retrieved label terms per topic with semantic scores into a vector for topic probability prediction and optimize the aggregated vector during model tuning (detailed in Appendix A).

4 Experiments

We present the experimental settings of SciPrompt across scientific classification datasets in few and zero-shot scenarios.

4.1 Datasets

We use three publicly available datasets in English for our experiments: SDPRA 2021 (Reddy and Saini, 2021), arXiv (Meng et al., 2019), and S2ORC (Lo et al., 2020). SDPRA 2021 contains scientific articles from computer science across seven categories. arXiv (Meng et al., 2019) includes abstracts sourced from the arXiv website[6] across 53 sub-categories, and S2ORC contains academic papers from across 19 disciplines. For the S2ORC data, we only select abstracts with a single discipline label through the Semantic Scholar Public API[7]. The statistics and category examples of these datasets are shown in Table 5 and Appendix B.

4.2 Experimental Settings

SciPrompt is built upon the OpenPrompt framework (Ding et al., 2022b). We apply a consistent prompt template across all experiments (see Appendix G for more details). The experimental details are shown in Appendix A.

In the few-shot setting, we benchmark SciPrompt alongside standard fine-tuning, simplified prompt-tuning (PT), and previous state-of-the-art text classification models, including LM-BFF (Gao et al., 2021b), RetroPrompt (Chen et al., 2022a), and KPT (Hu et al., 2021). Standard fine-tuning takes all labeled training examples as input to tuning an MLM for text classification. We take the final representation of the `[CLS]` token as the output vector of the abstract (Cohan et al., 2020). Standard PT with a manually defined verbalizer (Ding et al., 2022b) only takes each lowercase topic name as a seed word for verbalization. We apply the same setting as in SciPrompt, including a unified prompt template, MLM, and the model's hyper-parameters.

[6] https://arxiv.org/

[7] https://www.semanticscholar.org/product/api

Examples	Method	SDPRA 2021	arXiv	S2ORC	Avg.
1	Fine-tuning$_{SciBERT}$	12.72 ± 3.70	2.03 ± 0.21	4.76 ± 0.85	6.50 ± 1.59
	Prompt-tuning$_{Manual}$	**71.68** ± 4.73	34.95 ± 1.45	40.88 ± 1.92	49.17 ± 2.70
	LM-BFF	68.95 ± 1.68	35.07 ± 1.31	41.50 ± 1.43	48.51 ± 1.47
	KPT	50.74 ± 3.03	32.18 ± 1.08	43.20 ± 1.33	42.04 ± 1.81
	SciPrompt	64.42 ± 3.64	**40.57** ± 1.60	**47.92** ± 1.67	**50.97** ± 2.30
	SciPrompt$_{Soft}$	62.65 ± 4.94	31.06 ± 1.74	29.94 ± 1.94	41.22 ± 2.87
5	Fine-tuning$_{SciBERT}$	16.45 ± 4.35	2.36 ± 0.55	5.63 ± 1.37	8.15 ± 2.09
	Prompt-tuning$_{Manual}$	83.46 ± 1.41	47.58 ± 1.68	49.53 ± 0.88	60.19 ± 1.32
	LM-BFF	79.97 ± 2.52	50.11 ± 0.88	48.67 ± 1.02	59.58 ± 1.47
	RetroPrompt	64.76 ± 3.57	31.37 ± 0.72	47.09 ± 1.38	47.74 ± 1.89
	KPT	77.71 ± 3.34	53.68 ± 1.69	50.40 ± 1.84	60.60 ± 2.29
	SciPrompt	81.81 ± 3.34	56.36 ± 0.95	**52.12** ± 1.59	**63.43** ± 1.96
	SciPrompt$_{Soft}$	**83.70** ± 2.86	**58.01** ± 0.94	47.44 ± 1.60	63.05 ± 1.80
10	Fine-tuning$_{SciBERT}$	17.44 ± 4.50	3.14 ± 1.15	6.31 ± 0.81	8.96 ± 2.15
	Prompt-tuning$_{Manual}$	85.60 ± 0.81	50.86 ± 2.89	52.15 ± 0.98	62.87 ± 1.56
	LM-BFF	82.66 ± 2.40	56.03 ± 0.65	50.51 ± 1.19	63.07 ± 1.41
	RetroPrompt	74.44 ± 1.63	36.49 ± 1.07	49.82 ± 0.78	53.58 ± 1.16
	KPT	83.82 ± 0.72	61.83 ± 0.83	52.91 ± 0.66	66.19 ± 0.74
	SciPrompt	84.71 ± 0.89	62.37 ± 0.57	**53.65** ± 0.22	66.91 ± 0.56
	SciPrompt$_{Soft}$	**85.96** ± 0.60	**63.42** ± 0.50	52.41 ± 0.30	**67.26** ± 0.47
20	Fine-tuning$_{SciBERT}$	17.16 ± 3.90	3.53 ± 0.86	7.29 ± 1.32	9.33 ± 2.03
	Prompt-tuning$_{Manual}$	87.76 ± 0.70	52.92 ± 2.72	54.32 ± 0.89	65.00 ± 1.44
	LM-BFF	86.71 ± 1.36	60.90 ± 0.22	53.31 ± 1.07	66.97 ± 0.88
	RetroPrompt	77.89 ± 1.02	41.79 ± 0.81	50.55 ± 1.33	56.74 ± 1.05
	KPT	87.74 ± 0.51	66.25 ± 0.73	54.67 ± 0.43	69.55 ± 0.56
	SciPrompt	**87.95** ± 0.41	66.59 ± 0.64	**55.49** ± 0.56	**70.01** ± 0.54
	SciPrompt$_{Soft}$	87.90 ± 0.51	**66.86** ± 0.46	54.70 ± 0.42	69.82 ± 0.46
50	Fine-tuning$_{SciBERT}$	27.50 ± 9.48	11.07 ± 1.93	12.02 ± 2.22	16.86 ± 4.54
	Prompt-tuning$_{Manual}$	88.93 ± 0.57	60.63 ± 1.32	56.08 ± 0.29	68.55 ± 0.73
	LM-BFF	87.94 ± 0.56	64.75 ± 0.23	54.97 ± 0.69	69.22 ± 0.49
	RetroPrompt	83.14 ± 0.63	44.86 ± 1.22	53.04 ± 0.73	60.35 ± 0.86
	KPT	88.93 ± 0.37	69.95 ± 0.63	56.50 ± 0.81	71.79 ± 0.60
	SciPrompt	**88.99** ± 0.75	69.89 ± 0.63	**56.66** ± 0.49	**71.85** ± 0.62
	SciPrompt$_{Soft}$	88.97 ± 0.71	**70.15** ± 0.52	56.02 ± 0.60	71.71 ± 0.61
Full Set	Fine-tuning (Full) *	90.71	54.58	53.74	66.34

Table 1: Experimental results under few-shot settings. We report the mean accuracy (expressed in percentages %) and standard deviation based on five iterations across five learning shots. Fine-tuning (Full)* represents using a fully labeled training set. RetroPrompt experiments are only conducted in settings above five shots, as this method requires at least two labeled examples for model tuning.

KPT (Hu et al., 2021) applied external knowledge to enrich the verbalizer with additional word relevance and frequency filtering strategies. Our experiments use the same MLM (i.e., SciBERT) for equal comparison. Besides, training and validation examples per class (Ding et al., 2022b; Hu et al., 2021; Wang et al., 2022a) are uniform during model tuning, conducting tests with 1, 5, 10, 20, and 50 shots across all datasets and reporting accuracy as an evaluation metric. We evaluate model performance across five random seeds to account for variability (Hu et al., 2021; Ding et al., 2022b).

For the zero-shot setting, we sample approximately 10% of each dataset for testing, ensuring adequate representation for each topic. For broader model comparison, we introduce two additional models specific to the zero-shot scenario: SimPTC (Fei et al., 2022) and NPPrompt (Zhao et al., 2023). Moreover, we extend our evaluation to include Llama 2 (Touvron et al., 2023), ChatGPT (OpenAI, 2024), and the latest Llama 3 (AI@Meta, 2024) using in-context learning for a broader range of comparisons. Random seeds are applied in KPT, which samples an unlabeled support set of 200 examples to calibrate label words.

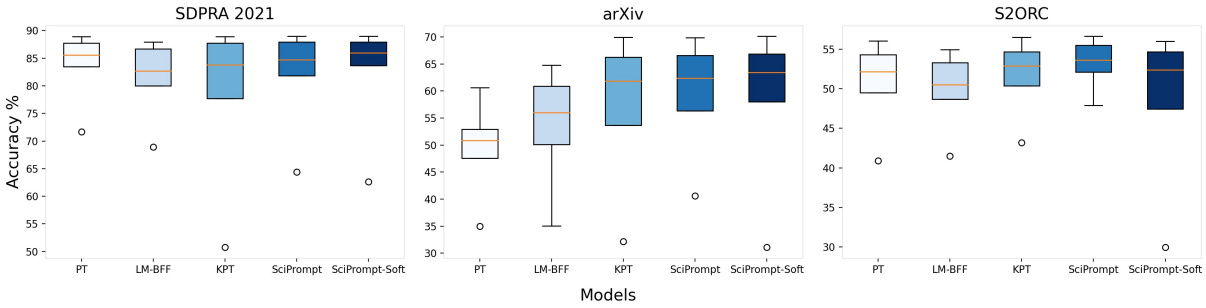

Figure 2: Performance comparison of few-shot methods over three datasets in Table 1. We report the mean accuracy of each setting. Our method shows high stability in the accuracy distribution compared to the considered baseline models.

5 Results and Analysis

5.1 Main Results

We highlight the performance of SciPrompt against baseline models across our three considered datasets in both few-shot and zero-shot settings, focusing on the fine-grained and cross-domain scientific text classification tasks. The experimental shown are listed in Table 1. Results are averaged over five runs as the same as KPT (Hu et al., 2021) to counteract sampling randomness, reported as mean accuracy with standard deviation.

Few-shot Results. SciPrompt achieves the best average accuracy on all three datasets for all settings. Specifically, SciPrompt and SciPrompt$_{Soft}$ excel in low-data scenarios (e.g., one-shot and five-shot), particularly on arXiv and S2ORC, often outperforming baseline models. SciPrompt also outperforms KPT by 8.93% in the one-shot setting and 2.83% in the five-shot setting. As the number of training examples increases, the margin of improvement over baseline models narrows. Notably, SciPrompt exceeds the full-set fine-tuning by an average of 0.57%, 3.67%, and 5.51% with 10, 20, and 50 shots, respectively. Despite variability in performance improvements across different training sizes, our method consistently achieves the highest accuracy on arXiv and S2ORC across all configurations. Also, the standard deviation of all three datasets decreases as the number of input training examples increases across all three datasets.

Additionally, Figure 2 provides a comprehensive comparison of performances across all few-shot settings, ranging from one-shot to fifty-shot, for each dataset as outlined in Table 1. SciPrompt consistently delivers high and stable accuracy across all three datasets compared to the baseline models. Particularly on S2ORC, SciPrompt achieves a higher median accuracy and a narrower interquartile range, indicating more consistent performance across different few-shot scenarios. The SciPrompt$_{Soft}$ method shows high stability on the SDPRA 2021 dataset, while SciPrompt is more effective in fine-grained datasets.

Methods	SDPRA 2021	arXiv	S2ORC	Avg.
Llama 2	62.04	26.98	40.30	43.11
Llama 3	**81.15**	**54.87**	**49.58**	**61.87**
ChatGPT	79.43	54.51	46.95	60.30
PT	62.97	20.81	32.93	**38.90**
SimPTC	15.79	3.25	11.35	10.13
NPPrormpt	35.00	13.98	37.23	28.74
LM-BFF	**64.79**	14.96	34.07	37.94
RetroPrompt	18.32	7.83	35.47	20.54
KPT	41.50±3.00	20.83±0.18	38.42±0.66	33.58±1.28
SciPrompt	51.97	**22.28**	**41.30**	38.52

Table 2: Performance of zero-shot setting. Only KPT is reported through mean accuracy (%) and standard deviation (§4.2). We apply the same instruction for ChatGPT, Llama 2, and Llama 3 on the test sets.

Zero-shot Results. Shown in Table 2, the Llama 3 70B model leads in performance across all datasets. Nonetheless, SciPrompt outperforms other baseline models, especially on arXiv and S2ORC, where it outperforms PT and KPT by margins of 1.47% and 2.88%, respectively. Meanwhile, LM-BFF leads among all baseline models on the SDPRA 2021 dataset. These results underscore the effectiveness of SciPrompt in leveraging domain-specific knowledge for fine-grained scientific text classification, even in the absence of labeled training data. Llama 3's average accuracy exceeds SciPrompt by 23.35% and Llama 2's by 18.76%. However, on the S2ORC dataset, SciPrompt surpasses Llama 2. Note that SciPrompt$_{Soft}$ is not designed for zero-shot testing since it needs trainable tokens in the decoding layer during model tuning.

Method	K=1	K=5	K=10	K=20	K=50	Avg.	Zero-shot
KPT	32.18±1.08	53.68±1.69	61.83±0.83	66.25±0.73	69.95±0.63	56.78	20.83
SciPrompt	40.57±1.60	56.36±0.95	62.37±0.57	66.59±0.64	69.89±0.63	**59.16**	**22.28**
w/o *CL*	40.19±1.46	55.84±0.98	62.32±0.50	66.45±0.61	69.92±0.64	58.94	21.87
w/o *SS*	38.70±0.86	55.19±0.80	62.48±0.59	66.70±0.77	69.73±1.01	58.56	6.17
w/o *SS*+*CL*	38.36±0.86	54.76±0.86	62.25±0.56	66.54±0.81	69.86±0.92	58.35	5.62
w/o *FL*+*CL*	29.77±0.74	50.13±0.88	59.57±0.97	65.77±0.47	69.55±0.70	54.96	3.77
SciPrompt$_{Soft}$	31.06±1.74	58.01±0.94	63.42±0.50	66.86±0.46	70.15±0.52	57.90	-
w/o *CL*	38.65±0.90	58.33±1.62	63.64±0.66	67.05±0.55	70.41±0.56	59.62	-
w/o *SS*	41.49±1.38	58.36±0.99	63.70±0.75	67.26±0.75	70.20±0.21	**60.20**	-
w/o *SS*+*CL*	42.22±1.32	57.72±1.46	63.53±0.57	67.03±0.78	70.35±0.49	60.17	-
w/o *FL*+*CL*	37.50±1.31	57.66±1.49	63.63±0.49	67.13±0.93	70.24±0.49	59.23	-

Table 3: Ablation study of SciPrompt for mean accuracy and standard deviation for the arXiv dataset under few-shot and zero-shot settings.

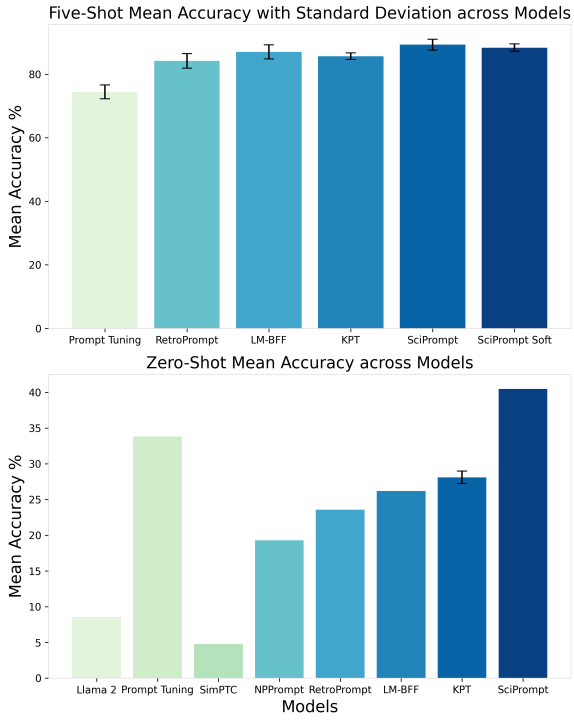

Figure 3: Model comparison through the Emerging NLP dataset under five-shot and zero-shot settings (§5.2).

5.2 Emerging Topics Classification

To assess our method's effectiveness in classifying emerging scientific topics, we manually collect a dataset centered around recent developments in the field of NLP, drawing inspiration from Ahmad et al. (2024). Specifically, we first extract NLP topics from Taxonomy4CL[8], focusing on topics that have emerged since 2000, as identified through Semantic Scholar[9]. We then select scientific articles published after 2019 that are beyond the knowledge cutoff of the SciBERT model. For each selected topic, we gather 30 abstracts, applying the same random seeds for few-shot experiments as those introduced in Table 1. We create a new dataset named **Emerging NLP** by collecting 21 fine-grained NLP-related topics and their corresponding abstracts. Appendix B provides detailed dataset statistics and topic examples. Figure 3 compares the performance of various baseline models. Notably, SciPrompt exceeds the performance of the Llama 2 70B model by 31.91% and outperforms the PT method by 6.67% in the zero-shot setting. Overall, our method outperforms all state-of-the-art methods in classifying emerging scientific topics, especially in the zero-shot setting, highlighting our method's efficacy in highly low-resource scenarios.

5.3 Ablation Study

Our ablation study on the arXiv dataset (Table 3) demonstrates the advantages of our models over KPT, with a 1.45% increase in zero-shot accuracy. SciPrompt and SciPrompt$_{Soft}$ outperform KPT by 2.38% and 3.42%, respectively, in terms of average accuracy under the few-shot setting. We examine the impact of removing full-size calibration ("w/o *CL*"), semantic scores ("w/o *SS*"), and both ("w/o *SS*+*CL*"), finding that both components improve the performance, especially in the zero-shot setting where their absence lowers accuracy by 0.41% ("w/o *CL*") and 16.11% ("w/o *SS*") compared to SciPrompt, underlining the critical role of *SS* in bolstering the model's effectiveness.

Interestingly, SciPrompt$_{Soft}$ performs better without *SS* than when both components are included. Removing both *SS* and *CL* yields the best 1-shot performance, suggesting that less intervention

[8] https://github.com/DFKI-NLP/Taxonomy4CL

[9] https://www.semanticscholar.org/

Method	SDPRA 2021	arXiv	S2ORC
SciPrompt			
w/o CL	9.5%	12.5%	12.4%
w/ CL	29.3%	51.2%	59.6%
SciPrompt$_{Soft}$			
w/o CL	12.3%	12.3%	12.3%
w/ CL	12.8%	13.2%	13.4%

Table 4: The usage percentage of GPU memory during model tuning.

optimizes model tuning in low-data contexts. Furthermore, comparing setups without pre-filtering and calibration ("w/o *FL+CL*") to those with pre-filtering shows an accuracy increase by 3.39% and 0.94% for SciPrompt and SciPrompt$_{Soft}$ respectively, highlighting the effectiveness of pre-filtering of augmented verbalizer for text classification. The ablation studies of SDPRA and S2ORC shows the same pattern as on arXiv.

5.4 Model Tuning Efficiency

Table 4 shows that SciPrompt$_{Soft}$ reduces GPU memory usage by 16.5 percentage points (p.p.) for SDPRA 2021, 38 p.p. for arXiv, and 46.2 p.p. for S2ORC compared to SciPrompt's full-size label term calibration. Although SciPrompt achieves higher average accuracy rates in the few-shot setting on the S2ORC dataset (see Table 6 in Appendix C), SciPrompt$_{Soft}$ outperforms SciPrompt on SDPRA 2021 and arXiv, suggesting that SciPrompt$_{Soft}$ can achieve competitive results with less GPU usage. Moreover, while ChatGPT and Llama 2 exhibit superior performance in the zero-shot setting, as shown in Table 2, it is worth noting that these language models are either mainly for commercial use or require substantial GPU resources, incurring higher costs or more time. For instance, for the S2ORC dataset, our method not only cuts down the combined training and testing (inference) time by 93 p.p. compared to Llama 2 70B but also enhances accuracy by 1 p.p. over Llama 2, highlighting the efficiency and effectiveness of our approach.

6 Conclusion

We introduced a knowledge-enhanced, prompt-based fine-tuning framework for fine-grained scientific text classification using minimally or no labeled abstracts. Acknowledging the complexity of domain knowledge within scientific literature, we employed a prompt-tuned MLM augmented with domain knowledge injection and semantic filtering. This approach enables the automatic extraction of domain-specific phrases and their integration into a weighted verbalizer for topic projection. Our findings highlight the effectiveness of our methods over existing state-of-the-art models and standard full-set fine-tuning, particularly for emerging topic classification and scenarios requiring high levels of topic granularity. Notably, SciPrompt demonstrates competitive accuracy compared to the advanced Llama 2 70B model in the zero-shot setting, showing its potential to categorize scholarly topics with a lightweight and efficient approach.

7 Limitations

Our study's limitations are as follows: 1) Our external knowledge sources are limited to two non-scientific domain databases for retrieving topic words, potentially missing fine-grained scientific terminologies. Despite the challenge of identifying a universally applicable, cross-domain, scientific knowledge resource, future efforts should aim to discover more precise terminology databases (Han et al., 2020). 2) We focus solely on a multi-class classification task and exclude abstracts that span multiple scientific sub-domains. Advancing towards a multi-label classification system capable of identifying publications across various domains would enhance the robustness of our approach. 3) Although SciPrompt and SciPrompt$_{Soft}$ surpassed baseline methods during evaluation, the enhancements are modest, and results fluctuate, particularly with an increase in labeled training data. Further investigation into the causes of these minimal gains as well as more comprehensive, interpretable experiments are needed to better understand and improve the model performance. 4) We only used classification accuracy and standard deviation as model evaluation metrics. The experimental results can change when using other metrics (e.g., Micro F1 and Macro F1). Additionally, while the standard deviation of our methods shrinks as the number of training examples increases, one could do statistical significance testing to draw robust conclusions by comparing system performance against baseline models.

8 Ethics Statement

The datasets and MLM employed in our study are publicly accessible and extensively utilized in the

research community. To enhance the quality of our data, we applied heuristic filtering to exclude short-length abstracts across these datasets, acknowledging that this process may impact experimental accuracy. Our methodology includes extracting data from external knowledge bases via public APIs. Furthermore, as we used MLMs as the foundation of our approach, it is essential to note that the predictive behavior of these models can be challenging to regulate due to the implicit knowledge embedded within the MLMs, which is difficult to decode explicitly. Therefore, caution should be exercised when adapting our method to other tasks, especially in the context of text classification through prompting.

References

Raia Abu Ahmad, Ekaterina Borisova, and Georg Rehm. 2024. Forc4cl: A fine-grained field of research classification and annotated dataset of nlp articles. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7389–7394.

AI@Meta. 2024. Llama 3 model card.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Xiang Chen, Lei Li, Ningyu Zhang, Xiaozhuan Liang, Shumin Deng, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022a. Decoupling knowledge from memorization: Retrieval-augmented prompt learning. *Advances in Neural Information Processing Systems*, 35:23908–23922.

Yulong Chen, Yang Liu, Li Dong, Shuohang Wang, Chenguang Zhu, Michael Zeng, and Yue Zhang. 2022b. AdaPrompt: Adaptive model training for prompt-based NLP. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6057–6068, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. 2020. SPECTER: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282, Online. Association for Computational Linguistics.

Washington Cunha, Vítor Mangaravite, Christian Gomes, Sérgio Canuto, Elaine Resende, Cecilia Nascimento, Felipe Viegas, Celso França, Wellington Santos Martins, Jussara M Almeida, et al. 2021. On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study. *Information Processing & Management*, 58(3):102481.

Ning Ding, Yulin Chen, Xu Han, Guangwei Xu, Xiaobin Wang, Pengjun Xie, Haitao Zheng, Zhiyuan Liu, Juanzi Li, and Hong-Gee Kim. 2022a. Prompt-learning for fine-grained entity typing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6888–6901.

Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. 2022b. Openprompt: An open-source framework for prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 105–113.

Joshua Eykens, Raf Guns, and Tim CE Engels. 2021. Fine-grained classification of social science journal articles using textual data: A comparison of supervised machine learning approaches. *Quantitative Science Studies*, 2(1):89–110.

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*, pages 8560–8579.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021a. 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 (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021b. 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 (Volume 1: Long Papers)*, pages 3816–3830.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2022. Ppt: Pre-trained prompt tuning for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8410–8423.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. 2021. WARP: Word-level Adversarial ReProgramming. 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), pages 4921–4933, Online. Association for Computational Linguistics.

Kanyao Han, Pingjing Yang, Shubhanshu Mishra, and Jana Diesner. 2020. Wikicssh: extracting computer science subject headings from wikipedia. In *ADBIS, TPDL and EDA 2020 Common Workshops and Doctoral Consortium: International Workshops: DOING, MADEISD, SKG, BBIGAP, SIMPDA, AIMinScience 2020 and Doctoral Consortium, Lyon, France, August 25–27, 2020, Proceedings 24*, pages 207–218. Springer.

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

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Juanzi Li, and Maosong Sun. 2021. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. *arXiv preprint arXiv:2108.02035*.

Mayara Khadhraoui, Hatem Bellaaj, Mehdi Ben Ammar, Habib Hamam, and Mohamed Jmaiel. 2022. Survey of bert-base models for scientific text classification: Covid-19 case study. *Applied Sciences*, 12(6):2891.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2023b. Gpt understands, too. *AI Open*.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S Weld. 2020. S2orc: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983.

Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2019. Weakly-supervised hierarchical text classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6826–6833.

Sewon Min, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Noisy channel language model prompting for few-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5316–5330, Dublin, Ireland. Association for Computational Linguistics.

OpenAI. 2024. Chatgpt. https://openai.com/chatgpt/. Accessed: 2024-05-20.

Oded Ovadia, Menachem Brief, Moshik Mishaeli, and Oren Elisha. 2023. Fine-tuning or retrieval? comparing knowledge injection in llms. *arXiv preprint arXiv:2312.05934*.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *Advances in neural information processing systems*, 34:11054–11070.

Saichethan Miriyala Reddy and Naveen Saini. 2021. Overview and insights from scope detection of the peer review articles shared tasks 2021. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 73–78. Springer.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Mobashir Sadat and Cornelia Caragea. 2022. Scinli: A corpus for natural language inference on scientific text. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7399–7409.

Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

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

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

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

Han Wang, Canwen Xu, and Julian McAuley. 2022a. Automatic multi-label prompting: Simple and interpretable few-shot classification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5483–5492.

Jianing Wang, Chengyu Wang, Fuli Luo, Chuanqi Tan, Minghui Qiu, Fei Yang, Qiuhui Shi, Songfang Huang, and Ming Gao. 2022b. Towards unified prompt tuning for few-shot text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 524–536.

Zhiwen You, HaeJin Lee, Shubhanshu Mishra, Sullam Jeoung, Apratim Mishra, Jinseok Kim, and Jana Diesner. 2024a. Beyond binary gender labels: Revealing gender bias in LLMs through gender-neutral name predictions. In *Proceedings of the 5th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 255–268, Bangkok, Thailand. Association for Computational Linguistics.

Zhiwen You, Shruthan Radhakrishna, Shufan Ming, and Halil Kilicoglu. 2024b. UIUC_BioNLP at BioLaySumm: An extract-then-summarize approach augmented with Wikipedia knowledge for biomedical lay summarization. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pages 132–143, Bangkok, Thailand. Association for Computational Linguistics.

Haoxing Zhang, Xiaofeng Zhang, Haibo Huang, and Lei Yu. 2022. Prompt-based meta-learning for few-shot text classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1342–1357.

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)*, pages 15590–15606.

Datasets	#Abstracts	# Classes	Avg. Length	Test
arXiv	55300	53 (sub)	129	5300
SDPRA 2021	28000	7	155	2800
S2ORC	65700	19	136	5700
Emerging NLP	630	21	227	420

Table 5: Datasets Statistics. #Abstracts represents the total number of labeled abstracts, including train and test sets. Emerging NLP dataset is for five-shot and zero-shot settings only.

A Experimental Details

All models use the maximum input length of 256 tokens over 5 epochs, using the same hyperparameters as KPT (Hu et al., 2021), with a learning rate of 3e-5 and a batch size of 5. The experiments are performed on a 32 GB Tesla V100 GPU.

In few-shot setting, we apply the same backbone MLM for all experiments, with the exception of RetroPrompt (Chen et al., 2022a). RetroPrompt only supports RoBERTa-based models and requires at least two examples per class for model tuning. Therefore, we apply `roberta-base` as base model for RetroPrompt and only conduct experiments with more than five shots.

The main distinction between SciPrompt and SciPrompt$_{Soft}$ lies in the verbalization, as discussed in Section 3.6. Unlike SciPrompt, which uses single label term projection, SciPrompt$_{Soft}$ employs a vector-based mapping method to represent each filtered set of label terms.

In zero-shot setting, we include ChatGPT[10], open-sourced Llama 2[11], and the latest Llama 3[12] for zero-shot classification using the same instruction. For ChatGPT, we use `gpt-3.5-turbo-instruct`, which contains 175 million model parameters developed by OpenAI. We apply `llama-2-70b-chat` and `meta-llama-3-70b-instruct` as the backbone models for Llama 2 and Llama 3 respectively through the Replicate API[13]. We additionally investigate the classification performance of the Llama 2 models with 7B and 13B parameters under the zero-shot setting. However, their outputs are not coherent with the predefined class label sets and often include redundant information, making the calculation of prediction accuracy unreliable. Therefore, we only conduct experiments of the Llama family on the 70B models.

B Datasets and Examples of Domain Topic Categories

We present a more detailed introduction to datasets used for our experiments.

SDPRA 2021 contains topics of scientific articles from the field of computer science, consisting of abstracts sourced from arXiv and categorized under one of seven predefined domain labels. We combined the training and validation sets, reallocating them into new training (90%) and validation (10%) sets.

arXiv includes abstracts sourced from the arXiv website collected by Meng et al. (2019), categorized into 53 sub-categories and 3 parent categories (i.e., Math, Physics, and CS). We select 100 samples for each category as test set.

S2ORC includes academic papers across 19 disciplines. We filter abstracts to those with a single discipline label from the 2023-11-07 release through the Semantic Scholar Public API[14].

Emerging Topics of NLP encompasses 21 newly developed research fields within the broader category of Computation and Language[15]. We collect 30 examples for each topic, assigning five instances for training and another five for validation. The rest of the examples are used for testing.

In our experiments, abstracts shorter than 30 tokens were excluded to remove invalid abstracts, leading to final training and test sizes of 25,110 and 2,790 for SDPRA, 49,300 and 5,300 for arXiv, 60,000 and 5,700 for S2ORC, and 210 and 420 for Emerging NLP. We used sub-categories for arXiv and parent categories for both SDPRA and S2ORC in text classification tasks. Detailed class labels for each dataset are presented in Table 11. We report parent and sub-categories of four datasets.

C Experiments of Various Verbalizer Sizes

As presented in Table 6, we document the performance metrics across various verbalizer sizes following the configurations outlined in Figure 4. We report the mean accuracy for each setting. The findings indicate that the model's performance is

[10] https://openai.com/chatgpt
[11] https://llama.meta.com/
[12] https://ai.meta.com/blog/meta-llama-3/
[13] https://replicate.com/
[14] https://www.semanticscholar.org/product/api
[15] https://arxiv.org/list/cs.CL/recent

Paradim	K=1	K=5	K=10	K=20	K=50	Avg.	Zero-Shot
SciPrompt (SDPRA)							
w/o *FL*	45.23	73.20	81.61	87.40	88.94	75.28	25.56
w/ *FL*	61.25	81.33	84.67	87.78	89.05	80.83	34.98
w/ *CL*	63.56	81.57	84.62	88.02	89.02	**81.36**	**51.40**
SciPrompt$_{Soft}$ (SDPRA)							
w/o *FL*	55.00	80.62	83.84	87.91	88.69	79.21	-
w/ *FL*	65.53	83.43	85.26	88.13	88.97	**82.26**	-
w/ *CL*	64.92	81.78	85.46	87.79	89.14	81.82	-
SciPrompt (arXiv)							
w/o *FL*	29.77	50.13	59.57	65.77	69.55	54.96	3.77
w/ *FL*	38.36	54.76	62.25	66.54	69.86	58.35	5.62
w/ *CL*	38.70	55.19	62.48	66.70	69.73	**58.56**	**6.17**
SciPrompt$_{Soft}$ (arXiv)							
w/o *FL*	37.50	57.66	63.63	67.13	70.24	59.23	-
w/ *FL*	42.22	57.72	63.53	67.03	70.35	60.17	-
w/ *CL*	41.49	58.36	63.70	67.26	70.20	**60.20**	-
SciPrompt (S2ORC)							
w/o *FL*	41.27	49.22	52.69	55.30	56.31	50.96	25.25
w/ *FL*	46.00	51.23	53.43	55.25	56.15	52.41	26.11
w/ *CL*	47.55	51.85	53.52	55.32	56.67	**52.98**	**40.79**
SciPrompt$_{Soft}$ (S2ORC)							
w/o *FL*	42.35	50.10	51.89	54.52	56.17	51.01	-
w/ *FL*	46.33	50.24	52.83	54.76	56.17	52.07	-
w/ *CL*	46.34	51.09	53.02	54.59	55.82	**52.17**	-

Table 6: Performance comparison under various number of label terms in the verbalizer. We report the mean accuracy after five runs for each shot.

enhanced across all scientific domain text classification datasets in both few-shot and zero-shot scenarios, attributable to implementing more sophisticated label term filtering techniques.

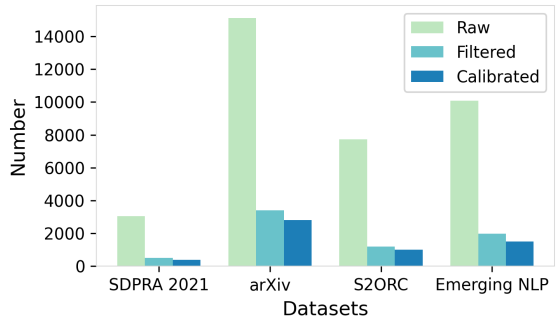

Figure 4: Various numbers of label terms across four datasets under three phrases.

D Calibration of Domain Knowledge

Figure 4 compares the verbalizer label term counts across datasets: "Raw" reflects the initial count after knowledge retrieval from two KBs (§3.2); "Filtered" shows counts post-semantic filtering (§3.4), reducing terms by 84%, 77%, and 85%; "Calibrated" involves removing low-likelihood terms before model tuning. Appendix C and Table 6 reveal that NLI filtering and calibration enhance the model's accuracy in few-shot and zero-shot settings, linking domain-relevant phrases in the verbalizer to improve the classification performance.

E Overall Model Performance Analysis

We present an overview comparison of the results from Table 1 across all three datasets (i.e., SDPRA 2021, arXiv, and S2ORC) in Figure 5. Overall, SciPrompt exhibits the most stable performance compared to other baseline methods. Notably, SciPrompt consistently outperforms the state-of-the-art model KPT across all three datasets. In contrast, SciPrompt$_{Soft}$ demonstrates variability and inconsistency compared with SciPrompt while showing a similar median accuracy. We excluded the RetroPrompt method from this comparison due to its inability to perform in the one-shot setting.

Datasets	$\mathcal{M}_{ce}$ < 0.1	$\mathcal{M}_{ce}$ < 0.3	$\mathcal{M}_{ce}$ < 0.6	$\mathcal{M}_{ce}$ < 0.9	$\mathcal{M}_{ce}$ > 0.9
SDPRA	495	501	514	531	738
arXiv	3,384	3,477	3,553	3,678	5,646
S2ORC	1,182	1,216	1,239	1,283	1,771

Table 7: The number of filtered label terms applying various thresholds.

Cross-Encoder	$\mathcal{M}_{be}$ < 0.5	$\mathcal{M}_{be}$ > 0.5	$\mathcal{M}_{be}$ > 0.6	$\mathcal{M}_{be}$ > 0.7	$\mathcal{M}_{be}$ > 0.8	$\mathcal{M}_{be}$ > 0.9
$\mathcal{M}_{ce}$ < 0.1	64.18±5.83	64.42±3.64	65.94±4.84	64.69±5.24	64.79±4.19	66.67±3.90

Table 8: Ablation study of SciPrompt in various $\mathcal{M}_{be}$ values under the fixed $\mathcal{M}_{ce}$ using the SDPRA 2021 dataset.

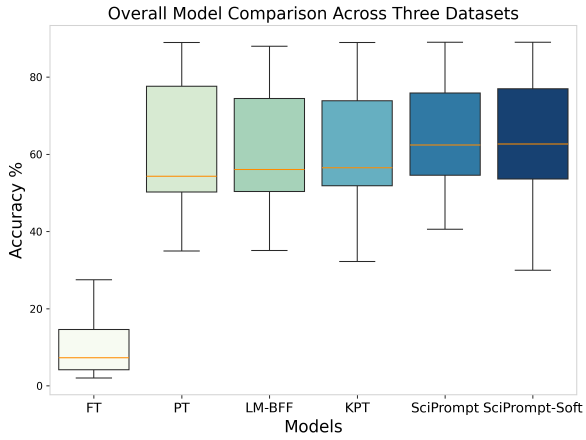

Figure 5: Box chart for all methods in the few-shot setting over three datasets.

F Knowledge-Retrieval Threshold Selection

As we introduced in Section 3.4, during the label term filtering stage, we employ a bi-encoder for $\mathcal{M}_{be}$ and a cross-encoder for $\mathcal{M}_{ce}$ calculation. In our experimentation, a higher $\mathcal{M}_{be}$ score indicates a more notable similarity between the topic labels and the retrieved label terms, thus enhancing the relevance of the selected terms. Conversely, a lower $\mathcal{M}_{ce}$ score signifies higher relevance during the re-ranking stage. Our analysis of the SDPRA dataset reveals that $\mathcal{M}_{ce}$ scores predominantly clustered above 0.9 and below 0.1. Consequently, the median value of $\mathcal{M}_{ce}$ exerts minimal influence on the final Verbalization process. Even when reducing the threshold of $\mathcal{M}_{ce}$ to 0.5, only a marginal difference in the number of selected label terms across various $\mathcal{M}_{ce}$ scores within the range of 0.1 to 0.9 is observed (Table 7).

We also explored the impact of different $\mathcal{M}_{be}$ values under the fixed $\mathcal{M}_{ce}$ score (0.1) to assess performance variations of SciPrompt in the 1-shot setting through the SDPRA dataset. Our findings indicate that while $\mathcal{M}_{be} > 0.9$ yields the optimal performance, $\mathcal{M}_{be} > 0.5$ kept the lowest standard deviation (Table 8). Consequently, we assume setting $\mathcal{M}_{be} > 0.5$ as the filtering threshold is more stable across different experimental conditions.

Bi-Encoder	$\mathcal{M}_{ce}$ < 0.1	$\mathcal{M}_{ce}$ < 0.5	$\mathcal{M}_{ce}$ > 0.5
$\mathcal{M}_{be} > 0.5$	64.42±3.64	63.80±5.11	34.86±6.61

Table 9: Ablation study of SciPrompt in various $\mathcal{M}_{ce}$ values under the fixed $\mathcal{M}_{be}$ using the SDPRA 2021 dataset.

To further validate our choices, we conducted experiments of SciPrompt with varying $\mathcal{M}_{ce}$ values under the 1-shot setting using the SDPRA dataset while maintaining a constant $\mathcal{M}_{be}$ threshold of 0.5. Notably, performance consistently improved and the standard deviation is stable when $\mathcal{M}_{ce}$ is set below 0.1 (Table 9). Therefore, we adopted $\mathcal{M}_{ce}$ = 0.1 as the filtering threshold.

G Prompt Templates of LLMs

Cloze-Based Prompt Template of MLM

`Abstract`. The field of this study is related to: `[MASK]`.

Above is the cloze-based prompt template we applied for all MLM prompt-based fine-tuning tasks. We also explored various prompt templates as introduced by (Hu et al., 2021; Gao et al., 2021a; You et al., 2024b) to evaluate performance variations using the SDPRA 2021 dataset, where the results are found to be similar. Note that our method focuses on improving domain-related verbalization

process rather than creating diverse prompts for model tuning.

As detailed in Section 5.1, we used ChatGPT, Llama 2, and Llama 3 to perform the task of scientific text classification guided by specific instructions. The same instructions were applied to all LLMs to infer the topics from scientific abstracts. We employed a distinct task-oriented (You et al., 2024a) prompt from that used with MLMs due to our observation that the original prompt from SciPrompt fails to yield relevant field names, given the LLMs' limitations in comprehension. Consequently, we crafted a more elaborate set of instructions to direct the LLMs in classifying topics, employing a projection of pre-defined class names similar to those used in the verbalization.

Instructions of LLMs

Based on the given article's abstract, please classify the abstract to a specific field of study. Only select field words from the following field words I provided. Only select one field name as output for each abstract. Your output should be all in lower cases.\n

Field Words List: logic in computer science, distributed computing, software engineering, data structures and algorithms, computational linguistics... \n

Abstract: For the purpose of developing applications for Post-K at an early stage, RIKEN has developed a post-K processor simulator. This simulator is based on the general-purpose processor simulator gem5...\n

Field of Study:

The "Field Words List" represents the original class names in the dataset. We concatenated the above instructions to LLMs and extract the predictions that appear after "Field of Study:" to evaluate the classification performance.

H Examples of Retrieved Label Terms

In Table 10, we report some cases of filtered label terms using the KBs we introduced in Section 3.2 through four datasets we apply for this work.

Datasets	Class Labels	Filtered Label Terms
arXiv	Databases	document-oriented database, hierarchical database, database management system, object database, database application
	Accelerator Physics	accelerator physics, particle accelerator, particle beam, velocity,accelerator
	Group Theory	symmetry group, group homomorphism, representation theory of finite groups, compact lie group
SDPRA 2021	Cryptography	cryptographers, secure communication, ciphertext, cryptanalytics, cryptographers, secure communication, data encryption standard
S2ORC	Political Science	political behavior, aspects, politics, elections, practical politics, american political science, constitutions, governing
	Psychology	psychological science, mental condition, mental state, mental function, psychological state, psychological condition
Emerging NLP	Large Language Models (LLMs)	bert, semi-supervised learning, chain-of-thought prompting, encoding, lstm
	Recurrent Neural Networks (RNNs)	tensor, language modeling, generative model, feedforward neural networks, gated recurrent unit

Table 10: Examples of filtered label terms in four datasets (§3.4).

Datasets	Parent-category	Sub-category
arXiv	Math (25)	numerical analysis, algebraic geometry, functional analysis, number theory, complex variables, applied mathematics, general mathematics, logic, optimization and control, statistics, probability, differential geometry, combinatorics, operator algebras, representation theory, classical analysis, dynamical systems, group theory, quantum algebra, rings and algebras, symplectic geometry, algebraic topology, commutative algebra, geometric topology, metric geometry
	Physics (10)	optics, fluid dynamics, atomic physics, instrumentation and detectors, accelerator physics, general physics, plasma physics, chemical physics, sociophysics, classical physics
	CS (18)	computer vision, game theory, information theory, machine learning, distributed computing, cryptography, networking and internet architecture, computational linguistics, computational complexity, software engineering, artificial intelligence, systems and control, logic in computer science, cryptography and security, data structures and algorithms, programming languages, other computer science, databases
SDPRA 2021	Computer Science (7)	logic in computer science, distributed computing, software engineering, data structures and algorithms, computational linguistics, networking and internet architecture, cryptography
S2ORC	engineering, chemistry, computer science, business, political science, environmental science, physics, economics, geography, medicine, psychology, art, materials science, mathematics, sociology, geology, philosophy, biology, history	-
Emerging NLP	Natural Language Processing (21)	sign language and fingerspelling recognition, rule-based machine translation (RBMT), transformer models, prompt engineering recurrent neural networks (RNNs), large language models (LLMs), bilingual lexicon induction (BLI), hate and offensive speech detection, email spam and phishing detection, fake news detection, fake review detection, aspect-based sentiment analysis (ABSA), dialogue state tracking (DST), visual question answering (VQA), open-domain question answering, multiple choice question answering (MCQA), nlp for for social media, nlp for the legal domain, acronyms and abbreviations detection and expansion, paraphrase and rephrase generation, named entity recognition for nested entities

Table 11: Detailed topic categories of four datasets. Note we classify sub-categories for arXiv, SRPRA 2021, and Emerging NLP datasets.