Weakly-Supervised Scientific Document Classification via Retrieval-Augmented Multi-Stage Training

Ran Xu*
Emory University
Atlanta, GA, USA

Yue Yu*
Georgia Institute of Technology
Atlanta, GA, USA

Joyce Ho
Emory University
Atlanta, GA, USA

Carl Yang†
Emory University
Atlanta, GA, USA

ABSTRACT

Scientific document classification is a critical task for a wide range of applications, but the cost of collecting human-labeled data can be prohibitive. We study scientific document classification using label names only. In scientific domains, label names often include domain-specific concepts that may not appear in the document corpus, making it difficult to match labels and documents precisely. To tackle this issue, we propose WanDeR, which leverages *dense retrieval* to perform matching in the embedding space to capture the semantics of label names. We further design the label name expansion module to enrich its representations. Lastly, a self-training step is used to refine the predictions. The experiments on three datasets show that WanDeR outperforms the best baseline by 11.9%. Our code will be published at https://github.com/ritaranx/wander.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**.

KEYWORDS

Scientific Document Classification, Weak Supervision, Retrieval

ACM Reference Format:
Ran Xu, Yue Yu, Joyce Ho, and Carl Yang. 2023. Weakly-Supervised Scientific Document Classification via Retrieval-Augmented Multi-Stage Training. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3539618.3592085

1 INTRODUCTION

Scientific document classification aims to assign scientific literature to pre-defined categories, supporting various applications [5, 27, 38]. Recently, pretrained language models (PTLMs) have demonstrated impressive performance in document classification [1, 7]. However, they often require a large number of annotations for fine-tuning, which restricts their deployment in real-world applications. While practitioners cannot afford to label many documents, it is often easier for them to provide category-descriptive label names as *weak supervision* for each class [17, 34]. Motivated by this, we focus on scientific document classification under the setting where only the label name for each class as well as the unlabeled corpus are available [18]. This task is challenging as the label names can be short and succinct, often containing a few words only. How to mine class-relevant knowledge with weak supervision is nontrivial.

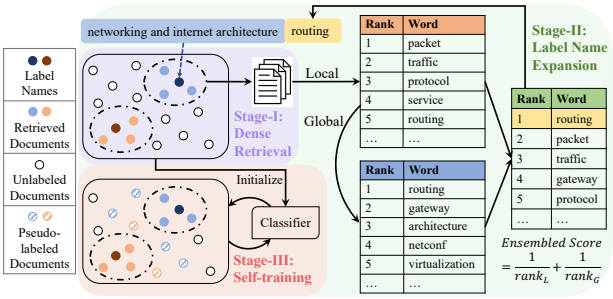


Figure 1: Framework of WanDeR.

There exist plenty of studies on automatic document categorization using class-relevant keywords [16–18, 25, 33]. These methods often leverage the keywords as input to extract relevant documents with hard matching for pseudo label generation. Although these methods achieve competitive performance, they mainly focus on tasks from *general domains*. For these tasks, the keywords can be commonly used words (e.g. 'Good/Bad' for reviews), and they can be matched with many examples. However, for scientific documents, the label names can either be too domain-specific, or contain multiple concepts [37]. As a result, they often have limited coverage over the corpus, which causes performance degradation when applying prior weakly-supervised techniques to the scientific domain.

In this work, we propose WanDeR (**We**akly-supervised Sci**en**tific Text Classification using **De**nse **R**etrieval), a multi-stage training framework for weakly supervised text classification using dense retrieval (DR), as shown in Figure 1. In DR, both queries and documents are represented as dense vectors, and the relevance between them is calculated via similarity metrics (e.g. dot product) [14]. This makes DR an ideal choice to tackle the above challenges, as it captures the semantics for different classes and circumvents the mismatch issue. To incorporate DR into the framework, we regard label names as queries, and retrieve the most relevant documents from the unlabeled corpus for each class (**Stage-I**, Sec. 3.1) to create

*Ran and Yue contributed equally to this research. E-mail: ran.xu@emory.edu.
†Corresponding Author. Email: j.carlyang@emory.edu.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3592085

an initial set of pseudo-labeled documents, which can be used to fine-tune the PTLM for the target task.

Although Stage-I is able to extract relevant documents, their performance can be less satisfactory as label names are insufficient to capture all the class-specific information. To overcome this drawback, in **Stage-II**, we expand the label names with the extracted keywords using local and global information (Sec. 3.2). Specifically, we first adopt the TF-IDF algorithm [11] on the retrieved documents to select the top-ranked words. In addition, we use the PTLM to calculate the embedding similarity between the candidate words and the label names as the global score. The local and global information is connected via an ensemble ranking module, and we augment the label name for each class by selecting the word with the highest score. The above expansion step is repeated multiple times to enrich the query [8] and help the DR model retrieve more relevant documents from the corpus.

To leverage all unlabeled data to further improve the performance, an additional step is to harvest *self-training* [18, 28] (**Stage-III**, Sec. 3.3) to refine the PTLM classifier by bootstrapping over high-confident examples and improve its generalization ability.

We verify the effectiveness of WanDeR by conducting experiments on three datasets and show that our model outperforms the previous weakly-supervised approaches by a large margin. Our analysis further confirms the advantage of leveraging dense retrieval for tackling the limited coverage issue of label names as well as the efficacy of multi-stage training for improving the performance.

2 PRELIMINARIES

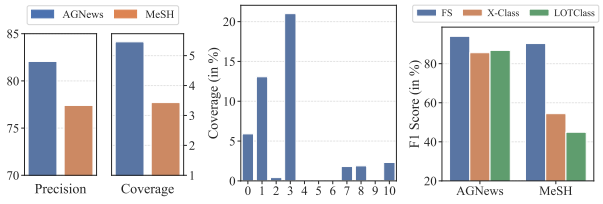
2.1 Problem Definition

Our weakly-supervised scientific document classification with C classes is defined as follows. The input is a training corpus $\mathcal{X} = \{d_1, d_2, \ldots, d_{|\mathcal{X}|}\}$ of documents without any labels. In addition, for each class c ($1 \le c \le C$), a label-specific name w_c is given, which consists of one or a few words. We aim to learn a classifier $f(x;\theta): \mathcal{X} \to \mathcal{Y}$. Here $\mathcal{X}$ denotes all samples and $\mathcal{Y} = \{1, 2, \cdots, C\}$ is the label set. While there exist works on multi-label classification [22] or metadata-aware classification [9, 36], we focus on the basic setting by assuming (1) each document only belongs to one category and (2) no other metadata information are available.

2.2 Challenges for Scientific Text Classification

While existing weakly supervised methods [18, 25] achieve competitive performance on general-domain datasets, applying them directly to scientific datasets often causes performance degradation. To illustrate this, we use AGNews [35] as the general-domain dataset and MeSH [5] as the scientific dataset. The average *precision* (i.e. the portion of correctly matched examples) and *coverage* (i.e. the portion of examples that can be matched by label names) for label names are shown in Figure 2a. We observe that for the scientific domain, the precision and coverage decline by 6% and 37% respectively. Moreover, the results of per-class coverage (presented in Figure 2b) indicate that the label distribution is more *imbalanced* for scientific data. For MeSH, there are 4 out of 11 classes where the label name cannot match any examples from the unlabeled corpus.

These two issues prevent the previous weakly-supervised models [18, 25] from performing well. As shown in Figure 2c, gaps to the



(a) Prec./Coverage (in %). (b) Per-class Coverage. (c) Performance.

Figure 2: Pilot Studies. FS means *fully-supervised* model.

performance of the fully-supervised model are much larger for scientific datasets (36%) than general-domain datasets (8%), which indicates that these advanced techniques cannot resolve the unique challenges that exist in the scientific domain.

3 METHOD

From the analysis in the above section, we conclude that it is necessary to propose techniques beyond hard matching to better harvest the semantic label name information. Towards this goal, we present our framework WanDeR in Figure 1, a multi-stage training scheme based on dense retrieval, to perform document classification using label names only. The three stages are detailed below.

3.1 Stage-I: Dense Retrieval with Label Names

Directly using the label-indicative keywords to extract documents is sub-optimal for scientific documents, due to their limited coverage and inferior ability to capture the class-related semantics. Motivated by this, we propose to leverage *dense retrieval* (DR) [14] to effectively retrieve the most relevant documents. Specifically, DR represents the input information ("query") q and target corpus ("document") d in the continuous embedding space as $g(q;\phi), g(d;\phi)$ respectively, where $g(\cdot;\phi)$ is the dense retrieval model with ϕ being the parameter of g. Then, DR matches queries and documents via approximate nearest neighbor (ANN) using the relevance score $r(q,d;\phi) = \langle g(q;\phi), g(d;\phi)\rangle$, where $\langle\cdot,\cdot\rangle$ is the cosine similarity. Next, we introduce the approach to train the DR model as well as leverage DR to extract documents from the corpus $\mathcal{X}$.

□ **Task-adaptive DR Model Pretraining.** To pretrain a DR model $g(\cdot;\phi)$ on the corpus $\mathcal{X}$, we use the contrastive learning widely adopted in recent research [10, 12, 32]. Specifically, for each document $d_i \in \mathcal{X}$, we sample two sentences $d_{i,1}, d_{i,2}$ as the positive pair. The training objective for d_i can be written as

$$\ell_{\text{CL}} = -\log \frac{\exp\left(\tau \cdot \langle g(d_{i,1};\phi), g(d_{i,2};\phi)\rangle\right)}{\sum_{j=1,2}\sum_{d^- \in \mathcal{D}_i^-} \exp\left(\tau \cdot \langle g(d_{i,j};\phi), g(d_i^-;\phi)\rangle\right)}, \quad (1)$$

where $d_i^- \in \mathcal{D}_i^-$ are the in-batch negatives, and $\tau = 0.01$ is the parameter for temperature. Contrastive pretraining improves both the alignment and uniformity for embeddings [13, 24, 29, 30], which can better support the retrieval task in our framework.

□ **Document Retrieval using Label Names.** With the DR model, we aim to extract an initial set of labeled data for each class by feeding the label names (as queries) to the DR model. The initial retrieved document set $\mathcal{D}_i$ for the i-th class can be written as

$$\mathcal{D}_i = \text{Top-}k^{\text{ANN}}_{d\in\mathcal{X}} r(w_i, d;\phi), \quad (2)$$

where k is the number of retrieved examples, and the label of the retrieved document is determined by the category of the label name.

In this way, we get rid of the challenge brought by those infrequent label names and provide a flexible way to encode the label-related semantics. All retrieved examples $\mathcal{D} = \cup_{i=1}^{C} \mathcal{D}_i$ are then used for classification, which will be discussed in the following part.

□ **Training Classifiers with Retrieved Text.** With the retrieved document set $\mathcal{D}$, one can simply finetune a classifier $f(\cdot;\theta)$ with the standard cross-entropy loss:

$$\min_{\theta} \mathbb{E}_{(x_i,y_i)\in\mathcal{D}} \ell_{\text{CE}}\left(f(x_i;\theta), y_i\right). \tag{3}$$

The fine-tuned model is used for target classification tasks.

3.2 Stage-II: Expand Label Names with Local and Global Information

One drawback of the above stage is that the label names are often too abstract to fully represent the semantics information for classes. As such, the retrieved documents still contain label noise, which hurts the downstream performance. To tackle this, we propose to automatically extract class-related keywords to expand the label name, by using both local information from the retrieved documents and global information from the general pretrained models.

□ **Local Information for Keyword Extraction.** To identify the class-related keywords, we assume terms that appear frequently within documents from a specific class while infrequently for other classes are more likely to be class-indicative words for that class [16]. Inspired by TF-IDF [11], we measure the indicativeness of word w for class c from the retrieved document $\mathcal{D}$ as

$$L_{w,c} = \text{tf}_{w,c}^{\alpha} \cdot \log\left(1 + A/\text{tf}_w\right) \cdot \text{cnt}_{w,c}. \tag{4}$$

Here $\text{tf}_{w,c}$, $\text{cnt}_{w,c}$ stands for the frequency and occurrence time of word w within documents from class c and tf_w, is the frequency of w in corpus, A is the average number of words per class. In this way, words appear commonly in the class-related documents while being less generic will receive higher score. For each class, we extract m words with the highest score as the candidate set C.[1]

□ **Global Information for Keyword Semantics.** The above step only considers the word occurrence in the local corpus, without modeling the semantic information. An ideal keyword, however, should also have a closer meaning to the label name. Motivated by this, we leverage the PTLM to transfer the *global* knowledge from pretraining corpora and encode the contextual information for each word. We calculate the embeddings of both label names and candidate words by averaging the output of all tokens from the last layer of PTLM $h(\cdot;\psi)$. For word $w \in C$ from the candidate set of class c, the global score is calculated between w and the label name w_c using the embedding similarity as

$$G_{w,c} = \langle h(w;\psi), h(w_c;\psi)\rangle. \tag{5}$$

□ **Ensemble Reranking.** To effectively combine the local and global information, we sort candidate words $w \in C_i$ for i-th class using the score $L_{w,c}, G_{w,c}$, respectively. Then, each word w will have two ranks as $\text{rank}_{L,c}(w)$ and $\text{rank}_{G,c}(w)$. We rerank the words using the ensemble score based on Reciprocal Rank Fusion (RRF) [6]:

$$\text{score}_{w,c} = 1/\text{rank}_{G,c}(w) + 1/\text{rank}_{L,c}(w). \tag{6}$$

For each class, we add one word with the highest score to expand the label name. For expansion, we simply concatenate the previous label name and the newly identified word for enrichment [2, 19].

[1]We omit words that already appeared in label names during the expansion (stage-II).

Table 1: Dataset statistics.

Dataset	Domain	# Train	# Test	# Class	# OOV	Avg. Len.
MeSH	BioMedical	16.3k	3.5k	11	4 (36%)	254.3
arXiv-Math	Mathematics	62.5k	6.3k	16	3 (19%)	214.4
arXiv-CS	Computer Science	75.7k	5.1k	20	5 (25%)	188.2

□ **Iterative Label Name Expansion.** The above process can be conducted multiple times. In each iteration, we first use local and global scores to detect the expanded words using Eq. (4)–(6) and enrich the label names. Then, we use the expanded label names as queries to update the retrieved documents $\mathcal{D}$ with Eq. (2) as we expect the quality of $\mathcal{D}$ will improve by incorporating additional class-indicative words. With the updated $\mathcal{D}$, more relevant words can be extracted to enrich the class information. The above iteration is repeated 5 times, and the retrieved documents after the final iteration can be used to train another classifier using Eq. (3).

3.3 Stage-III: Refine Classifier with Self-training

The pseudo-labeled samples in Stage-II are only from the top retrieved documents with the expanded label names. To generalize its current knowledge to the whole unlabeled corpus, *self-training* is adopted to bootstrap the model on the entire unlabeled corpus [15, 18, 31] as

$$\min_{\theta} \mathbb{E}_{(x,\widetilde{y})\in\mathcal{X}} \mathbb{1}\left\{[f(x;\theta)]_{\widetilde{y}} > \gamma\right\} \times \ell_{\text{CE}}\left(f(x;\theta), \widetilde{y}\right), \tag{7}$$

where $\widetilde{y} = \text{argmax}\, f(x;\theta)$ is the hard pseudo label, γ is the confidence threshold. With self-training, the model is refined by its high-confident predictions to improve generalization ability. Stage-III stops when less than 1% of samples change their labels.

4 EXPERIMENTS

4.1 Experiment Setups

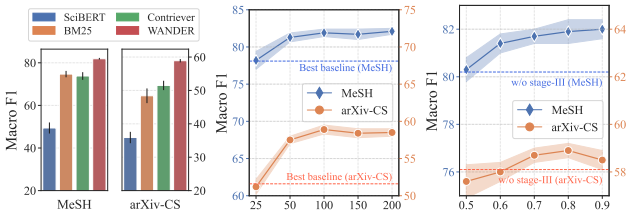
□ **Datasets.** We conduct experiments on three datasets from multiple domains including MeSH [5], arXiv-CS [4], arXiv-Math [4]. The statistics for each dataset are shown in Table 1. For arXiv-CS and arXiv-Math, we select papers from years 2017-2020 as the training set, 2021-2022 as the test set, and use the topic from the main category as the label.

□ **Baselines.** We compare WanDeR with these baselines: (1) **IR** [23] leverages TF-IDF to assign labels for documents. (2) **Dataless** [3] uses Wikipedia to embed labels and documents. Each document is classified to the label with the highest similarity. (3) **SentenceBERT** [20] is trained on NLI data to embed labels and documents for classification. (4) **LOTClass** [18], (5) **X-Class** [25], and (6) **FastClass** [26] are three recent methods that use PTLMs for label-name-only text classification by using masked language modeling or pretrained representations.

□ **Implementations.** We use the pre-trained SciBERT [1] as the backbone. The retrieval model g (Eq. (1)) and PTLM h (Eq. (5)) are initialized from SciBERT, and g is pretrained on the corpus $\mathcal{X}$ for 5 epochs. The maximum length is set to 512. For Stage-I and II, we finetune $f(\cdot;\theta)$ for 5 epochs with Adam as the optimizer and set the batch size and learning rate to 32 and 2e-5. Other hyperparameters include τ in Eq. (1), k for ANN in Eq. (2), γ in Eq. (7), m in Sec. 3.2. We set $\tau = 0.01, m = 100, k = 100, \gamma = 0.8, \alpha = 0.5$ without tuning. We study the effect of k, γ in Sec. 4.3.

Table 2: Performance on three datasets. Bold and blue indicate the best and second-best results for each dataset. *Macro-F1* is the main metric as the label distribution is imbalanced.

Method	MeSH		arXiv-Math		arXiv-CS	
	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1
Fully Supervised	90.5±0.3	90.3±0.2	80.6±0.4	79.1±0.3	83.0±0.2	78.2±0.4
IR [23]	40.6	37.6	27.8	22.9	24.5	22.8
Dataless [3]	36.1	26.8	18.9	13.4	20.5	18.2
SentenceBERT [20]	68.6	66.0	48.9	41.1	50.7	47.7
LOTClass [18]	57.9±1.7	44.9±1.6	43.8±2.0	35.2±1.5	51.5±1.4	47.1±1.8
X-Class [25]	55.2±1.4	54.4±1.8	46.5±1.4	39.1±1.4	60.6±1.2	51.6±1.3
FastClass [26]	78.5±1.3	78.1±1.1	53.5±1.3	44.5±1.2	59.8±0.8	50.5±0.9
WanDeR	**82.0**±0.4	**81.9**±0.4	**58.0**±0.8	**51.9**±0.7	**65.6**±0.8	**58.9**±0.6
Gain Δ	3.5 (4.4%)	3.8 (4.9%)	4.5 (8.4%)	7.4 (16.6%)	5.0 (8.2%)	7.3 (14.1%)
WanDeR (Stage-I)	76.6±1.0	75.6±0.8	56.4±1.4	49.8±0.9	61.8±1.1	54.7±1.2
WanDeR (Stage-II)	79.9±0.6	80.2±0.7	57.1±1.1	51.0±1.0	64.6±1.0	58.1±0.6



(a) Study of Retrievers (b) Study on k (c) Study on γ

Figure 3: Studies of Different Retrieval Models and Hyperparameters (Best View in Colors).

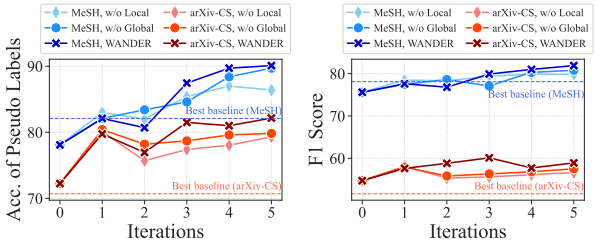
4.2 Experiment Results

□ **Main Experiments.** We report both Macro-F1 and Micro-F1 scores for WanDeR and baselines in Table 2. The mean and variance over 5 runs are calculated when fine-tuning is used. We observe that WanDeR consistently achieves the best performance on three datasets, with an average gain of 11.9%. In contrast, X-Class and LOTClass, which achieve strong results on general-domain tasks, fail to perform well on the scientific domain, as they cannot handle the challenges mentioned in Sec. 2.2. Moreover, traditional baselines, such as IR and Dataless, are inferior to other methods using PTLMs, indicating their limited ability for modeling scientific text. Although SentenceBERT and FastClass use extra labeled data for embedding learning, they fail to expand the label names for enriching representations, leading to sub-optimal performance.

□ **Effect of Multi-stage Training.** The bottom two rows in Table 2 show the performance of WanDeR after Stage-I and II, which justifies that all three stages contribute to the final performance. Moreover, WanDeR outperforms all baselines even without self-training (Stage-III), indicating that it can retrieve a small set of high-quality data to support downstream tasks sufficiently.

4.3 Ablation and Hyperparameter Studies

□ **Study of DR Models.** To illustrate the effect of task-adaptive contrastive learning (TAPT) for DR model pretraining, we substitute $g(\cdot)$ with other models including *BM25* [21], *SciBERT* [1] without TAPT, the strong unsupervised DR model *Contriever* [12], and compare the performance in Figure 3a. Overall, our model achieves the best performance, which justifies the need for TAPT as it effectively reduces the distribution shifts and also produces better embeddings. Instead, using sparse retrieval model (BM25) yields undesirable performance as it cannot understand label names well.



(a) Pseudo Label Accuracy (b) Macro-F1 Score

Figure 4: Study on Effects of Local and Global Information.

Table 3: Case Study on expanded keywords for three tasks.

Dataset	Class	Expanded Keyword
MeSH	Diabetes	insulin, glucose, diabetic, metformin, glycemic
MeSH	Neoplasms	tumor, carcinoma, cell, tumour, chemotherapy
arXiv-Math	Combinatorics	graph, combinatorial, vertex, edge, bipartite
arXiv-Math	Statistics theory	estimation, sample, regression, treatment, inference
arXiv-CS	Information theory	entropy, channel, shannon, capacity, decoder
arXiv-CS	Game Theory	player, equilibrium, nash, payoff, strategy

□ **Effect of Hyperparameters.** We study the effect of k and γ in WanDeR on MeSH and arXiv-CS, as shown in Figure 3b and 3c. We observe that the performance first increases with larger k as the model benefits from more retrieved examples. When k reaches 100, the performance remains stable, as too many retrieved examples introduce label noise and diminish the performance gain. We also run experiments with different thresholds γ. The result indicates that the model performance is insensitive to γ, and the self-training component leads to performance gain in most studied regions.

□ **Effect of Local and Global Information.** Figure 4 illustrates the performance of WanDeR and its variants over 5 expansion iterations. Overall, we observe that removing local or global information hurts the performance, since these two modules provide complementary information. Combining these two terms together results in better pseudo labels and improves downstream performance.

4.4 Case Studies

We present a case study in Table 3 to showcase that WanDeR is able to discover class-related keywords to expand label names. Take *diabetes* as an example, it is often related to high *glucose* level and *glycemic* index. Besides, *insulin* and *metformin* are used as treatments for diabetes. Moreover, take *machine learning* as another example, it is applied to *classifiation* tasks. *Boosting*, *ensemble*, *tree* are all techniques to tackle machine learning problems. These all indicate that WanDeR can enrich the semantics of label names.

5 CONCLUSION

We propose WanDeR, a multi-stage training framework for weakly-supervised scientific document classification with label name only. We leverage *dense retrieval* to go beyond hard matching and harness the semantics of label names. In addition, we propose a label name expansion module to enrich its representations, and use self-training to improve the model's generalization ability. Experiments on three datasets demonstrate that WanDeR outperforms the baselines by 11.9% on average. For future works, we plan to extend WanDeR to other scenarios such as multi-label classification.

REFERENCES

[1] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *EMNLP-IJCNLP*. 3615–3620.

[2] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. 2008. Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR*. 243–250.

[3] Ming-Wei Chang, Lev-Arie Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of Semantic Representation: Dataless Classification.. In *AAAI*. 830–835.

[4] Colin B Clement, Matthew Bierbaum, Kevin P O'Keeffe, and Alexander A Alemi. 2019. On the Use of ArXiv as a Dataset. *arXiv preprint arXiv:1905.00075* (2019).

[5] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S Weld. 2020. SPECTER: Document-level Representation Learning using Citation-informed Transformers. In *ACL*. 2270–2282.

[6] Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *SIGIR*. 758–759.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

[8] Fernando Diaz, Bhaskar Mitra, and Nick Craswell. 2016. Query Expansion with Locally-Trained Word Embeddings. In *ACL*.

[9] Soumyajit Ganguly and Vikram Pudi. 2017. Paper2vec: Combining graph and text information for scientific paper representation. In *ECIR*. 383–395.

[10] Luyu Gao and Jamie Callan. 2022. Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval. In *ACL*. 2843–2853.

[11] Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv preprint arXiv:2203.05794* (2022).

[12] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised Dense Information Retrieval with Contrastive Learning. *TMLR* (2022).

[13] Xuan Kan, Hejie Cui, Joshua Lukemire, Ying Guo, and Carl Yang. 2022. Fbnetgen: Task-aware gnn-based fmri analysis via functional brain network generation. In *MIDL*.

[14] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *EMNLP*. 6769–6781.

[15] Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. Bond: Bert-assisted open-domain named entity recognition with distant supervision. In *KDD*. 1054–1064.

[16] Dheeraj Mekala and Jingbo Shang. 2020. Contextualized weak supervision for text classification. In *ACL*. 323–333.

[17] Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018. Weakly-supervised neural text classification. In *CIKM*. 983–992.

[18] 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. *EMNLP* (2020).

[19] Shahrzad Naseri, Jeffrey Dalton, Andrew Yates, and James Allan. 2021. Ceqe: Contextualized embeddings for query expansion. In *ECIR*. 467–482.

[20] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP*. 3982–3992.

[21] Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval* 3, 4 (2009), 333–389.

[22] Jiaming Shen, Wenda Qiu, Yu Meng, Jingbo Shang, Xiang Ren, and Jiawei Han. 2021. TaxoClass: Hierarchical multi-label text classification using only class names. In *NAACL-HLT*. 4239–4249.

[23] Bruno Trstenjak, Sasa Mikac, and Dzenana Donko. 2014. KNN with TF-IDF based framework for text categorization. *Procedia Engineering* 69 (2014), 1356–1364.

[24] Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*. 9929–9939.

[25] Zihan Wang, Dheeraj Mekala, and Jingbo Shang. 2021. X-Class: Text Classification with Extremely Weak Supervision. In *NAACL*. 3043–3053.

[26] Tingyu Xia, Yue Wang, Yuan Tian, and Yi Chang. 2022. FastClass: A Time-Efficient Approach to Weakly-Supervised Text Classification. *EMNLP* (2022).

[27] Yi Xie, Yuqing Sun, and Elisa Bertino. 2021. Learning domain semantics and cross-domain correlations for paper recommendation. In *SIGIR*. 706–715.

[28] Ran Xu, Yue Yu, Hejie Cui, Xuan Kan, Yanqiao Zhu, Joyce Ho, Chao Zhang, and Carl Yang. 2023. Neighborhood-Regularized Self-Training for Learning with Few Labels. In *AAAI*, Vol. 37.

[29] R. Xu, Y. Yu, C. Zhang, M. K Ali, JC. Ho, and C. Yang. 2022. Counterfactual and factual reasoning over hypergraphs for interpretable clinical predictions on ehr. In *Machine Learning for Health*. PMLR, 259–278.

[30] Yi Yang, Hejie Cui, and Carl Yang. 2022. Pre-train Graph Neural Networks for Brain Network Analysis. In *IEEE-Big Data*.

[31] Yue Yu, Lingkai Kong, Jieyu Zhang, Rongzhi Zhang, and Chao Zhang. 2022. AcTune: Uncertainty-Based Active Self-Training for Active Fine-Tuning of Pretrained Language Models. In *NAACL*. 1422–1436.

[32] Yue Yu, Chenyan Xiong, Si Sun, Chao Zhang, and Arnold Overwijk. 2022. COCO-DR: Combating the Distribution Shift in Zero-Shot Dense Retrieval with Contrastive and Distributionally Robust Learning. In *EMNLP*. 1462–1479.

[33] Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang. 2021. Fine-Tuning Pre-trained Language Model with Weak Supervision: A Contrastive-Regularized Self-Training Approach. In *NAACL*. 1063–1077.

[34] Jieyu Zhang, Yue Yu, Yinghao Li, Yujing Wang, Yaming Yang, Mao Yang, and Alexander Ratner. 2021. WRENCH: A Comprehensive Benchmark for Weak Supervision. In *NeurIPS*.

[35] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level Convolutional Networks for Text Classification. In *NIPS*.

[36] Yu Zhang, Shweta Garg, Yu Meng, Xiusi Chen, and Jiawei Han. 2022. Motifclass: Weakly supervised text classification with higher-order metadata information. In *WSDM*. 1357–1367.

[37] Yu Zhang, Yu Meng, Xuan Wang, Sheng Wang, and Jiawei Han. 2022. Seed-Guided Topic Discovery with Out-of-Vocabulary Seeds. In *NAACL*. 279–290.

[38] Yuchen Zhuang, Yinghao Li, Junyang Zhang, Yue Yu, Yingjun Mou, Xiang Chen, Le Song, and Chao Zhang. 2022. ReSel: N-ary Relation Extraction from Scientific Text and Tables by Learning to Retrieve and Select. In *EMNLP*. 730–744.