FEW-TK: A Dataset for Few-shot Scientific Typed Keyphrase Recognition

Avishek Lahiri[*] **Pratyay Sarkar**[*] **Medha Sen**[*]

Debarshi Kumar Sanyal[*] **Imon Mukherjee**[†]

[*]Indian Association for the Cultivation of Science, Kolkata, India
[†]Indian Institute of Information Technology, Kalyani, India
{avisheklahiri2014,pratyay2sarkar,medhasen1001}@gmail.com
debarshi.sanyal@iacs.res.in, imon@iiitkalyani.ac.in

Abstract

Scientific texts are distinctive from ordinary texts in quite a few aspects like their vocabulary and discourse structure. Consequently, Information Extraction (IE) tasks for scientific texts come with their own set of challenges. The classical definition of Named Entities restricts the inclusion of all scientific terms under its hood, which is why previous works have used the terms Named Entities and Keyphrases interchangeably. We suggest the rechristening of Named Entities for the scientific domain as Typed Keyphrases (TK), broadening their scope. We advocate for exploring this task in the few-shot domain due to the scarcity of labeled scientific IE data. Currently, no dataset exists for few-shot scientific Typed Keyphrase Recognition. To address this gap, we develop an annotation schema and present FEW-TK, a dataset in the AI/ML field that includes scientific Typed Keyphrase annotations on abstracts of 500 research papers. To the best of our knowledge, this is the introductory few-shot Typed Keyphrase recognition dataset and only the second dataset structured specifically for few-shot NER, after FEW-NERD. We report the results of several few-shot sequence-labelling models applied to our dataset. The data and code are available at https://github.com/AvishekLahiri/Few_TK.git

1 Introduction

The recent past has witnessed an explosion in the amount of scientific literature available to us, especially with the advent of the Web and scholarly search engines. The expansiveness and variations in even a single scientific domain today requires a wide-ranging set of Information Extraction tools and datasets.

Named Entity Recognition (NER) is the Information Extraction task of identifying references to rigid designators (Nadeau and Sekine, 2007) and is the basic building block for a great number of Natural Language Processing and Information Retrieval tasks like relation extraction, question answering, knowledge graphs, and text summarization (Li et al., 2022; Yadav and Bethard, 2018).

> Retrieval Augment Generation (RAG) is a recent advancement in Open-Domain Question Answering (ODQA). RAG has only been trained and explored with a Wikipedia-based external knowledge base and is not optimized for use in other specialized domains such as healthcare and news. In this paper, we evaluate the impact of joint training of the retriever and generator components of RAG for the task of domain adaptation in ODQA. We propose RAG-end2end, an extension to RAG...

Table 1: Example of an annotated TACL abstract with scientific keyphrase mentions Algorithm/Tool-NLP, Focus-NLP, Allied Term-Misc., Allied Term-NLP, Study Domain-Application, Allied Term-AI/ML/DL, Focus-AI/ML/DL, Proposed Technique-NLP.

There is a dearth of labeled scientific text data that may be used for Information Extraction tasks and also a shortage of annotation schema that is able to provide a satisfactory coverage of the entire scientific information present in the text. Moreover, prior annotation schemata often lack portability for transfer to other scientific domains; for example, the "language resource" entity type in computational linguistics papers (QasemiZadeh and Schumann, 2016) is not relevant for the biology domain.

NER in scientific domain is frequently referred to as keyphrase extraction, which is due to the constrictive nature of the classical definition of Named Entities, which states that proper nouns are the only words that can be allowed as Named Entities (Petasis et al., 2000). Keyphrase extraction often singularly refers to the span detection of scientific terms, and not assigning types to them. Therefore, we term the scientific NER task as the Typed

Findings of the Association for Computational Linguistics: NAACL 2024, pages 4011–4025
June 16-21, 2024 ©2024 Association for Computational Linguistics

Keyphrase Recognition task. We present the reasons in detail in Section 2.

We aim to study Typed Keyphrase Recognition for the scientific domain in a more challenging low-resource context, specifically the few-shot setting, alongside the standard supervised setting. There is a substantial amount of research available on deep learning-based approaches to classify Named Entities (Li et al., 2022; Yadav and Bethard, 2018). But the difficulty with these approaches is that they are data-intensive approaches. The stumbling block is the collection of such an inordinately large amount of labeled data. This is where few-shot learning comes into the picture. Few-shot learning enables the generalization of the model to new unseen classes based on only a few labeled samples.

Therefore, we introduce the task of few-shot scientific Typed Keyphrase Recognition and design a novel annotation schema for the same. We use this schema to annotate scientific paper abstracts. Both coarse-grained and fine-grained keyphrase types have been included in the annotation schema to get an extended keyphrase type set. This helps us in the few-shot scenario because the latter requires testing and validation on unseen class types that we can easily get from the large number of keyphrase types. A sample annotation is shown in Table 1. Fine-graining of the keyphrase types has not been attempted in a similar scientific setting before. Our schema consists of 9 coarse-grained and 38 fine-grained keyphrase types as opposed to previous scientific NER research works which use only 1 to 7 coarse-grained types. We design the schema in such a way that our coarse-grained keyphrase types are portable to other scientific domains.

In summary, we make the following contributions: *(a)* We present the first human-annotated dataset, called FEW-TK, for few-shot Typed Keyphrase Recognition in scientific domain that is focused on the AI/ML literature. This dataset serves to mitigate the scarcity of labeled scientific IE data to some extent. To the best of our knowledge, ours is only the second few-shot dataset for NER, following FEW-NERD (Ding et al., 2021). We present this as a challenge dataset to the community because detecting scientific typed keyphrases from such an expanded label set is notably more challenging than in typical scenarios. *(b)* We introduce a new annotation schema for the scholarly domain that is portable to other scientific domains. This schema differs significantly from previous supervised NER schemata in terms of entity types and facilitates a broader coverage of entities. *(c)* We demonstrate the challenging nature of our dataset using several state-of-the-art deep neural models that have been developed, both for the standard supervised setting and the few-shot setting.

2 Scientific Typed Keyphrases

The term "Named Entity" was first coined at MUC-6 (Grishman and Sundheim, 1996), with its scope primarily limited to proper nouns (Petasis et al., 2000). In standard texts, Named Entity types such as Person, Organization, and Location exclusively pertain to proper names. However, there has been an unwritten agreement among researchers regarding the inclusion of temporal and numerical expressions as Named Entities (Nadeau and Sekine, 2007). Previous studies in the scientific domain have attempted to address this by framing the task simply as scientific NER (Luan et al., 2018; Hou et al., 2019; D'Souza et al., 2020; Kabongo et al., 2021; Jain et al., 2020). Yet, this definition of Named Entities limits the coverage of scientific terminology because scientific literature often uses many terms that are indispensable in terms of the semantic meaning they provide but that do not qualify as proper names. For example, in Table 1, the term "external knowledge base" may be quite useful for tasks like question answering, yet it does not fit into the standard definition of Named Entities.

In this paper, we investigate the task of extraction of keyphrases and their classification, which is similar to NER, but use the term "keyphrases" instead of Named Entities, to give the task a broader scope. Previous works using the "keyphrase" term (Hulth, 2003; Kim et al., 2010; Meng et al., 2017; Santosh et al., 2020; Tokala et al., 2020; Santosh et al., 2021) have predominantly focused on the Keyphrase Extraction task alone. In contrast, we amalgamate the classification and extraction tasks into a single task – Typed Keyphrase Recognition.

We have used the term "keyphrase" in a manner that is more consistent with its usage in (Augenstein et al., 2017). The authors categorized keyphrases into three types, namely, Process (including methods, equipment), Task, and Material (including corpora, physical materials); this is similar to our attributing types to keyphrases.

Thus, we introduce the terminology "Typed Keyphrases", which denotes words or phrases that have significance in the given scholarly text. They have a wider scope in the scientific domain as compared to Named Entities. This gives rise to the task of scientific Typed Keyphrase Recognition.

3 Problem Definition

In this section, we first offer a brief overview of few-shot learning, followed by a definition of Few-shot Typed Keyphrase Recognition.

3.1 Few-shot Learning

Few-Shot Learning (FSL) has been defined by (Wang et al., 2020; Song et al., 2023) as a type of machine learning problem (specified by experience E, task T and performance P), where E contains only a limited number of examples with supervised information for the target T.

3.2 Few-shot Typed Keyphrase Recognition

Few-shot Typed Keyphrase Recognition is symmetrical to Few-shot Named Entity Recognition when we formally define it in terms of tokens and labels. The main difference between the two tasks is in their semantic interpretation.

Given a sequence of tokens $X = x_1, x_2, ..., x_t$, we need the keyphrase recognition model to output a label $y_i \in Y$ for each token, where Y is the keyphrase type set.

In N-way, K-shot scientific Typed Keyphrase Recognition, it essentially means that there are N new categories during one test process, while there are K support samples for each category. Episodes in few-shot learning are defined as one sample of data that is composed of $N \times K$ support data and $N \times K'$ query data. For each episode in training, N classes (N-way) and K examples (K-shot) for each class are sampled to build a support set $S_{train} = \{x^{(i)}, y^{(i)}\}_{i=1}^{N*K}$, while K' examples for each of N classes are sampled to construct a query set $Q_{train} = \{x^{(j)}, y^{(j)}\}_{j=1}^{N*K'}$, such that $S_{train} \cap Q_{train} = \phi$ (Ding et al., 2021).

4 Dataset Creation

We annotate scientific keyphrases on 500 abstracts from four sub-domains of Artificial Intelligence/Machine Learning in the broad spectrum of Computer Science. Namely, these sub-domains are Natural Language Processing (NLP), classical Artificial Intelligence (AI) together with Machine Learning (ML), Data Mining together with Information Retrieval and Computer Vision (CV). The reason behind annotating at the abstract-level is that by taking abstracts instead of either only titles (D'Souza and Auer, 2021) or full texts (Augenstein et al., 2017; Hou et al., 2019), we get a distilled representation of the entire paper. Besides, the sentence length in abstracts is found to be substantially longer, yet not too long to resist processing by typical deep neural models used in NLP.

4.1 Abstract Selection

We hand-pick one highly reputed journal for each sub-domain, the details of which are shown in Table 2. The motivation for choosing journal abstracts over abstracts of conference-length papers is because of the considerably expanded length of abstracts in journals. We start selecting the abstracts from the latest issue available of the respective journal at the start of the year 2023.

Venue	Domain	No. of papers
TACL	NLP	240
JMLR	AI/ML	100
TPAMI	CV	80
TKDD	Data Mining	80

Table 2: Statistics of paper abstracts in Few-TK. TACL: Transactions of the Association for Computational Linguistics, JMLR: Journal of Machine Learning Research, TPAMI: IEEE Transactions on Pattern Analysis and Machine Intelligence, TKDD: ACM Transactions on Knowledge Discovery from Data.

4.2 Annotation Schema

Annotating scientific datasets for named entities is an inherently challenging task due to the inability of categorizing all words/phrases within a specific set of entities, while at the same time ensuring that all the necessary scientific information in a given text is captured. A particular challenge in classifying scientific keyphrases is that the number of classes easily explodes to a large number if we want to ensure a large coverage of the text.

To alleviate this problem, we propose the expansion of a core set of Typed Keyphrases (TK) so that there is a set of fine-grained Typed Keyphrases for each coarse-grained type. Such a schema not only divides the keyphrases on a conceptual level, but also provides a greater coverage of the scientific text than formerly explored entity schemata. The novel keyphrase schema that we develop takes the best out of the previous entity schemata that were developed in the scientific domain. Additionally,

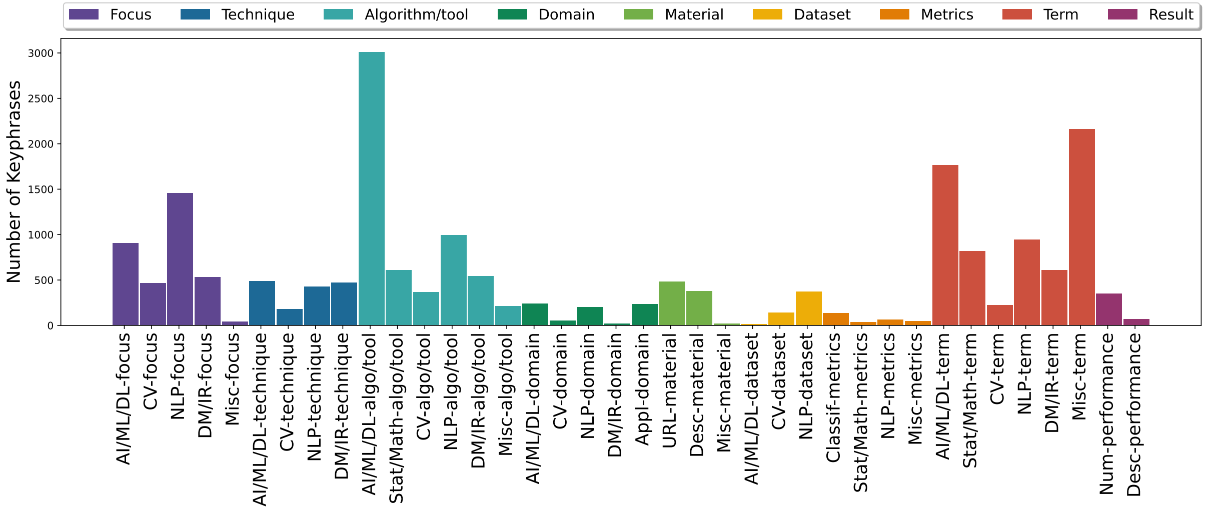


Figure 1: Keyphrase Set in FEW-TK.

we bring new keyphrase types into the fold and also refine these types into finer-grained categories.

In all, we present 9 coarse-grained and 38 fine-grained Typed Keyphrases. This type of a schema gives us the freedom of using either the coarse or the fine-grained types.

Although we have designed the entity schema exclusively for use in the context of Artificial Intelligence/Machine Learning literature, we argue that this schema can be exported to any other scientific domain literature if we take only the coarse-grained categories and align the fine-grained ones to the respective scientific domain. We use the Brat rapid annotation tool[1] for annotating the abstracts. An example annotation with Brat is presented in Appendix A.

5 Keyphrase Types

There are 9 coarse-grained keyphrase types in our dataset FEW-TK. The **Focus** keyphrase type mainly refers to the area of interest or problem that is being tackled in the article, i.e., the center of attention of the article. The **Proposed Technique** keyphrase type alludes specifically to the name of the modus operandi put forward in the paper. The **Algorithm/Tool** keyphrase type chiefly refers to any pre-existing concept that has been used in the paper. The **Allied Terms** keyphrase type primarily refers to all those phrases which do not fall in the above coarse-grained categories. The **Study Domain** keyphrase type calls attention to those phrases that refer to a particular discipline or subject area. The **Supplementary Material (Code/Library)** keyphrase type principally refers to any auxiliary resource that has been provided along with the paper. The **Dataset** keyphrase type refers to any dataset that has been used in the research article. The **Metric** keyphrase type essentially contains all those measures that are used in the AI/ML domain for evaluating various types of learning approaches. The **Performance** keyphrase type accommodates the results reported in the scientific text. The full set of Typed Keyphrases (coarse-grained and fine-grained) are described in Appendix B and presented in tabular format in Appendix C.

The Allied Terms category is very significant because it includes those entities that would have been easily overlooked in any other scientific schema; these entities often have considerable importance attached to their occurrence in the scientific text. For example, in Artificial Intelligence/ Machine learning literature, we encounter the word "training" several times, and it is quite an important term in this literature. However, existing entity schemata often overlook it or categorize it as a *generic* entity. Through our schema, such terms are afforded greater refinement with fine-grained types that better capture their nuanced meanings.

If we closely examine the pattern of our fine-grained keyphrase types, we observe that for most of the coarse-grained types, the fine-grained categories are predominantly theoretical AI/ML, NLP, Computer Vision, and Data Mining/Information Retrieval, because these are the leading areas of study within Artificial Intelligence. We argue that

[1] http://brat.nlplab.org/

Corpora	Domain	Classes	Papers	Tokens	Entities
FTD (Gupta and Manning, 2011)	CL	3	426	57,182	5,382
ACL RD-TEC (QasemiZadeh and Schumann, 2016)	CL	7	300	32,758	4,391
SciERC (Luan et al., 2018)	AI	5	500	60,749	8,089
NLP-TDMS (Hou et al., 2019)	CL	4	332	1,115,987	1,384
SciREX (Jain et al., 2020)	ML	4	438	2,487,091	156,931
NCG (D'Souza et al., 2021)	CL, CV	1	405	47,127	908
ORKG-TDM (Kabongo et al., 2021)	AI	3	5,361	-	18,219
CL-Titles (D'Souza and Auer, 2021)	CL	6	50,237	284,672	87,567
PwC (D'Souza and Auer, 2022)	AI	2	12,271	1,317,256	29,273
ACL (D'Souza and Auer, 2022)	CL	7	31,044	263,143	67,270
Few-NERD (Ding et al., 2021)	General (Few-shot)	66	188.2k sents	4601.2k	491.7k
Few-TK	AI	38	500	115,745	20064

Table 3: Comparison of Few-TK with other scientific-domain NER datasets and Few-NERD.

the coarse-grained keyphrase types can be used in other scientific domains like Physics and Chemistry. Similar to our fine-grained categories, the Physics domain can be divided into Astrophysics, Nuclear Physics, Thermodynamics, Biophysics, etc; the Chemical domain may be divided into Physical Chemistry, Organic Chemistry, Inorganic Chemistry, Analytical Chemistry, and Biochemistry. By incorporating keyphrase types relevant to these specialized fields, we can greatly assist downstream tasks such as question answering and knowledge graph construction.

5.1 Comparison with Other Datasets

We have proposed a schema that is significantly different from previous works in the scientific NER area. Details of previous scientific NER research can be found in Table 3. Our proposed dataset is more beneficial because it is portable to other domains, it incorporates fine-grained types, and is able to capture more nuanced scientific information. We are the first to come up with the few-shot setting in the scientific domain.

SciERC (Luan et al., 2018), which is one of the most popular datasets not only for the purpose of scientific NER, uses the following entity types: Task, Method, Dataset, Evaluation Metric, Material, Other Scientific Term, and Generic. At least four other entity schemata (Hou et al., 2019; D'Souza et al., 2020; Kabongo et al., 2021; Jain et al., 2020) either also use a subset of these entity types or have entity types bearing close resemblance to these entity types. However, none of them adequately captures the domain or sub-domain of the keyphrase.

5.2 Human Annotation

The authors of the present paper collectively decided the set of coarse-grained and fine-grained keyphrase types based on a sample annotation of 5 abstracts from each domain. The main first-draft annotation is then done by a domain expert in this field. Subsequently, two students who are very familiar in Machine Learning and Deep Learning concepts and terminology were each assigned to annotate 15% of the abstracts. This serves as the second annotation of (part of) our dataset. Each assigned subset contained 15% of the abstracts from each of the four domains in the dataset, with no overlap between the two subsets. Agreement between annotators was measured using Cohen's Kappa score to assess annotation quality. We instructed the annotators to closely follow the description provided for various keyphrase types and annotation guidelines that we had prepared. The annotation guidelines are present in detail in Appendix B. In cases of ambiguity regarding the span length, we have tried to resolve it by deciding it on the basis of its immediate context on a case-to-case basis. Conflicts between the annotations were resolved as much as possible through discussion between the annotators. The Cohen's Kappa scores between each student's annotation and the original annotation are 79.50% and 81.38% respectively.

6 Experiments

We demonstrate the challenging nature of our dataset by evaluating it on SOTA NER models that have been developed previously both for the fully-supervised settings and the few-shot settings. We now briefly describe these models.

6.1 BERT-tagger (Fully Supervised)

The output of a BERT-type model is fed into a linear classifier and trained using the cross-entropy training loss for the standard supervised setting.

6.2 Few-shot Models

We show the performance of the following models on our dataset, FEW-TK:

ProtoBERT: It is based on prototypical networks developed by Snell et al. (2017) and it principally computes the embeddings of the tokens that share the same label through an embedding function. The average of these embeddings gives an embedding representation known as the prototype. For each token in the query set, we calculate the prediction probability of that token with all the prototypes using the L_2 distance.

NNShot: Developed by Yang and Katiyar (2020), each token here is represented by its contextual representation in the sentence, and the query tag is decided by calculating the token-level Euclidean distance. Here, the similarity score is determined between a token in the query set and all tokens in the support set.

StructShot: This model is also developed by Yang and Katiyar (2020) and utilizes an additional Viterbi decoder (Forney, 1973) using an abstract tag transition distribution and an emission distribution over the basic architecture of NNShot (Hou et al., 2020). This method dispenses with the CRF training phase.

CONTAiNER: Introduced by Das et al. (2022), this model employs contrastive learning to refine the distributional divergence between similar and dissimilar classes. For this purpose, they use Gaussian embeddings instead of traditional token embeddings. In the calculation of the contrastive loss, positive samples consist of tokens with the same tag. The loss is measured by computing the KL-divergence between the respective token Gaussian embeddings. An instance level nearest neighbor classifier is used for the inferencing part.

MAML-ProtoNet: Ma et al. (2022) establish a decomposed meta-learning approach and address the problem in two steps: entity span detection and entity typing, the first of which is modelled as a sequence labelling problem, while for the second standard prototypical networks (Snell et al., 2017) are used. Model-agnostic meta-learning (MAML) (Finn et al., 2017) is used upon both the steps for better representative learning.

6.3 Benchmark Settings

We test the difficulty of our dataset for the fully supervised setting as well as for the few-shot setting using state-of-the-art models. In this section, we specify the details of modifying the dataset based on the respective setting.

6.3.1 Fully Supervised Setting

For the fully supervised setting, the whole dataset is simply split into train, validation and test, where we use the same ratio as Ding et al. (2021) i.e. the train:validation:test split is $70 : 10 : 20$, for the BERT-Tagger model.

6.3.2 Few-shot Setting

In few-shot NER, the overall entity set (ε) is split into three mutually disjoint subsets, $\varepsilon_{train}, \varepsilon_{dev}, \varepsilon_{test}$ such that $\varepsilon_{train} \cup \varepsilon_{dev} \cup \varepsilon_{test} = \varepsilon$ and $\varepsilon_{train} \cap \varepsilon_{dev} \cap \varepsilon_{test} = \phi$. This is done so that the few-shot setting of learning new classes from a limited number of examples may be preserved.

Ding et al. (2021) propose two settings for testing few-shot NER datasets, namely, the FEW-NERD (INTRA) and FEW-NERD (INTER) settings. For FEW-NERD (INTRA), $\varepsilon_{train}, \varepsilon_{dev}, \varepsilon_{test}$ are constructed by dividing the coarse-grained entity types among the three subsets ensuring that these subsets do not have any common entity type. In the case of FEW-NERD (INTER), the fine-grained categories are shared in a $60 : 20 : 20$ ratio among $\varepsilon_{train}, \varepsilon_{dev}, \varepsilon_{test}$, respectively.

We also replicate similar settings for FEW-TK, wherein FEW-TK (INTRA) is constructed such that the validation set holds the coarse-grained types Technique and Result, the test set contains the types Focus and Metric, while the train set contains the remaining coarse-grained keyphrase types. For FEW-TK (INTER), we randomly assign the fine-grained types based on the given ratio.

6.3.3 Experimental Setup

For the fully supervised scenario, the maximum sequence length is taken as 128 and batch size is 16. We test the BERT-Tagger model with both the uncased version of BERT (Devlin et al., 2019) and the uncased version of SciBERT (Beltagy et al., 2019).

We take a batch size of 8 for the Proto, NNShot and StructShot models and use $10,000$ steps to train the model while using a learning rate of $1e-4$. The batch size used for the CONTAiNER model is also 8, but the learning rate for finetuning is $5e-$

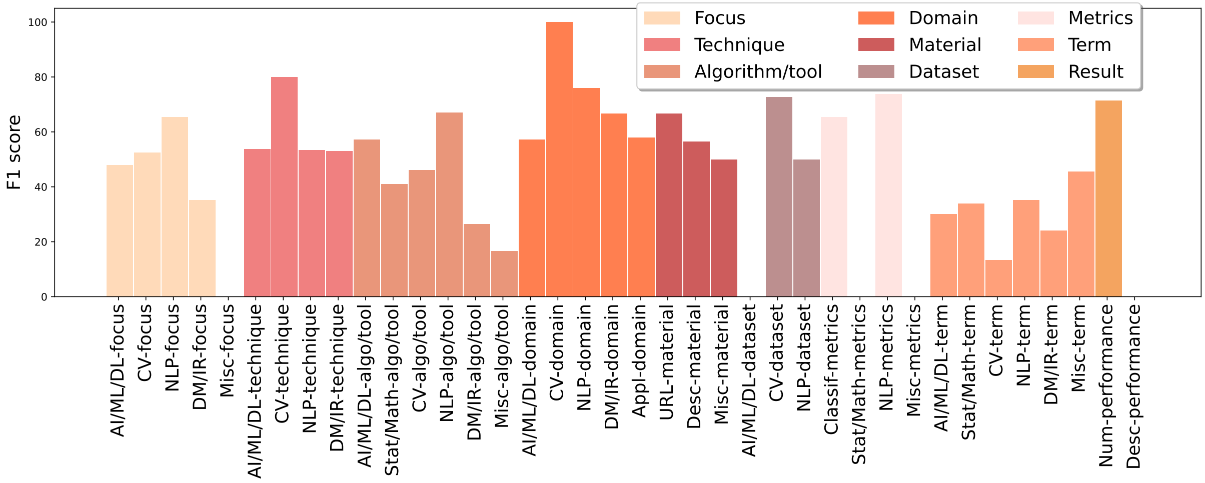


Figure 2: F1 score for every fine-grained keyphrase type in FEW-TK in the supervised setting when using SciBERT.

5. We have reported the results for each few-shot model by averaging the results from three separate runs of the model, each for a different random seed value. Here also, we use both the uncased version of BERT (Devlin et al., 2019) and the uncased version of SciBERT (Beltagy et al., 2019).

We measure the precision, recall and the micro-F1 score for each few-shot model to evaluate the complexity of our dataset. We train the models using an A100 GPU.

7 Results

The results for both the fully-supervised and few-shot frameworks are detailed out below.

7.1 Fully Supervised Setting

Dataset	Model	F1
SCIERC	BERT-Tagger	64.89
FEW-TK	BERT-Tagger	46.48 ↓
SCIERC	SciBERT-Tagger	65.81
FEW-TK	SciBERT-Tagger	48.91 ↓

Table 4: Performance of state-of-the-art fully supervised models on FEW-TK

Table 4 shows the results of the tagging model using two BERT-type models. We see that SciBERT (Beltagy et al., 2019) gives better results than BERT (Devlin et al., 2019) when used in the tagging model. However, the results on our dataset are significantly worse than that achieved on SCIERC (Luan et al., 2018), underscoring the challenging nature of our dataset even in the fully-supervised setting. This difficulty may primarily stem from the expanded keyphrase set present in our dataset.

Figure 2 shows the category-wise F1 scores for each fine-grained type. There are some classes which have very low or zero F1 score. This may be attributed both to the nature of the phrases in those classes and the low count of samples available for those classes as seen in Figure 1.

7.2 Few-shot Setting

Tables 5 and 6 show the results of the top five few-shot NER models on our typed keyphrase dataset. We see that all state-of-the-art few-shot sequence labelling models have produced low performance on our dataset, FEW-TK. There have also been some unexpected findings. Since FEW-TK is a dataset in the scientific domain, we conducted experiments with SciBERT and BERT as the backbone language models for the few-shot settings. Surprisingly, we observed that in most cases, using BERT produced better results than using SciBERT. Another noticeable factor is the very low performance achieved by the MAML-ProtoNet (Ma et al., 2022) in almost all cases. Our analysis revealed that the span detection part of the model was giving extremely low results, which was reflected in the final results of the model. But the type detection mechanism performed relatively well, achieving F1 scores in the range of 60-80%. CONTAINER (Das et al., 2022) works best for the INTRA case while StructShot (Yang and Katiyar, 2020) works best for the INTER scenario in terms of F1 score. Figure 3 shows the deviation of the F1 scores for different seed values for the CONTAINER model through a box plot. The above findings comprehensively show that a lot of work needs to be done in the area of few-shot Typed Keyphrase Recognition.

Model	Backbone Model	Intra					
		5-way 1-shot			3-way 1-shot		
		Precision	Recall	F1	Precision	Recall	F1
Proto (Snell et al., 2017)	BERT	36.85	17.58	23.44	34.73	17.45	23.16
NNShot (Yang and Katiyar, 2020)	BERT	26.93	32.67	29.38	25.42	34.22	29.07
StructShot (Yang and Katiyar, 2020)	BERT	27.88	**36.65**	31.66	37.00	29.09	30.46
CONTaiNER (Das et al., 2022)	BERT	35.79	32.73	34.19	36.61	34.32	35.39
MAML-ProtoNet (Ma et al., 2022)	BERT	3.55	3.15	3.26	3.95	5.23	4.38
Proto (Snell et al., 2017)	SciBERT	24.06	27.03	25.11	13.44	07.30	09.40
NNShot (Yang and Katiyar, 2020)	SciBERT	27.05	26.04	26.53	25.91	26.82	26.35
StructShot (Yang and Katiyar, 2020)	SciBERT	24.06	27.03	25.11	26.71	26.72	26.71
CONTaiNER (Das et al., 2022)	SciBERT	**39.23**	33.18	**35.95**	**43.22**	**35.74**	**39.11**
MAML-ProtoNet (Ma et al., 2022)	SciBERT	6.23	3.65	4.69	4.62	3.16	3.71

Table 5: Performance of state-of-the-art models on Few-TK (intra).

Model	Backbone Model	Inter					
		5-way 1-shot			3-way 1-shot		
		Precision	Recall	F1	Precision	Recall	F1
Proto (Snell et al., 2017)	BERT	25.35	34.43	29.18	25.18	38.17	30.26
NNShot (Yang and Katiyar, 2020)	BERT	43.65	48.32	45.86	47.27	**53.45**	50.17
StructShot (Yang and Katiyar, 2020)	BERT	44.76	**49.42**	**46.95**	48.97	53.32	**51.05**
CONTaiNER (Das et al., 2022)	BERT	**47.55**	42.57	44.91	47.03	43.76	45.32
MAML-ProtoNet (Ma et al., 2022)	BERT	5.82	4.91	5.12	6.96	9.41	7.65
Proto (Snell et al., 2017)	SciBERT	12.96	15.44	14.08	16.41	17.99	17.09
NNShot (Yang and Katiyar, 2020)	SciBERT	38.45	43.13	40.65	38.45	43.13	40.65
StructShot (Yang and Katiyar, 2020)	SciBERT	39.53	42.06	40.75	38.99	43.07	40.92
CONTaiNER (Das et al., 2022)	SciBERT	49.51	42.91	45.95	**52.00**	48.82	50.35
MAML-ProtoNet (Ma et al., 2022)	SciBERT	8.18	3.29	4.69	7.67	5.35	6.29

Table 6: Performance of state-of-the-art few-shot models on Few-TK (inter).

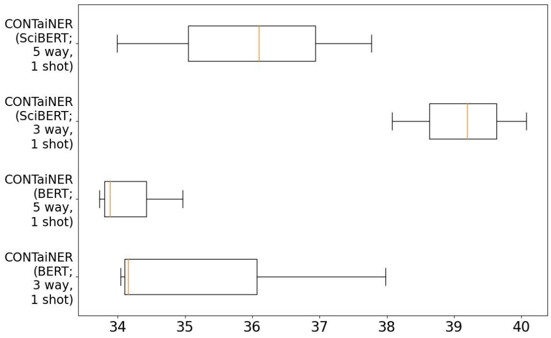


Figure 3: Box Plot for the CONTaiNER model in the Few-TK (intra) scenario for the 3-way 1-shot setting and the 5-way 1-shot settings respectively. The X-axis shows the F1 scores achieved by the model.

8 Error Analysis

We present our analysis for the NNShot model in Tables 7 and 8 for the span and type errors respectively. We use the 3-way 1-shot setting for our analysis. We consider two types of errors that occur when few-shot models try to classify Typed Keyphrases. If a model fails to detect the span of a Typed Keyphrase correctly, it is considered as a Span Error. If a token that must be included in a Typed Keyphrase is not done so by the model, it is called a False Negative (FN) case, while if a token is incorrectly included as part of a Typed Keyphrase, it is called a False Positive (FP) case.

When the span of a keyphrase has been correctly identified, if the model makes a misclassification while predicting the type of the keyphrase, it is termed a Type Error. If the model correctly predicts the coarse-grained type but fails to predict the fine-grained type accurately, it is termed a Within Error. On the other hand, if the model inaccurately predicts the coarse-grained type, it is referred to as an Outer Error.

Backbone Model	Intra		Inter	
	FP	FN	FP	FN
BERT-Base	4.50%	5.38%	2.72%	4.20%
SciBERT	3.20%	6.58%	2.74%	5.23%

Table 7: Span Error analysis of 3-way 1-shot setting using the NNShot model (Yang and Katiyar, 2020).

Backbone Model	Type Error	
	Within	Outer
BERT-Base	0.47%	0.95%
SciBERT	0.58%	1.72%

Table 8: Type Error analysis of 3-way 1-shot setting using the NNShot model (Yang and Katiyar, 2020).

9 Related Work

Automated IE from scientific literature has garnered significant interest from the NLP research community in recent years. (Gupta and Manning, 2011) introduce a method of extracting the Focus, Domain, and Techniques used in a scientific article. NLP-TDMS by (Hou et al., 2019) is a dataset containing the Task, Dataset, Metric and Score used in NLP papers, facilitating automated leaderboard construction. The ACL RD-TEC (QasemiZadeh and Schumann, 2016) dataset contains entities that are classified into 9 types. SciERC by Luan et al. (2018) contains entities of types Task, Method, Evaluation metric, Other-scientific-term, Material, and Generic. SciREX (Jain et al., 2020) uses both automatic and manual annotations to annotate entities including Method, Task, Metric, and Dataset as well as N-ary relations and co-references. NCG by (D'Souza et al., 2021) is the dataset used in a shared task to track scholarly contributions. ORKG-TDM (Kabongo et al., 2021) is a dataset to facilitate an approach for automated leaderboard extraction, encompassing Task, Method, and Metric entities. CL-Titles (D'Souza and Auer, 2021) is a dataset that was created based on lexico-syntactic patterns from titles in Computational Linguistics (CL) articles and contains entities identifying the Research problem, Resource, Tool, Language, Solution, and Method. ACL (D'Souza and Auer, 2022) is a part of the CS-NER dataset and contains 7 entities, namely, Language, Method, Research problem, Resource, Dataset, Solution, and Tool. PwC (D'Souza and Auer, 2022) was also introduced in the same work contains the research problem and method entities on `PapersWithCode`[2] data. In the context of few-shot learning, the work most closely related to ours is the Few-NERD dataset that was proposed by Ding et al. (2021), but it is for the general domain.

10 Discussion

The following points have come to our notice while creating the dataset. Generally, when considering both pure AI/ML literature and its sub-areas, we observe that abstracts from journals in allied AI fields, such as NLP or CV, often contain a considerable number of entities originating from the context of pure Artificial Intelligence, Machine Learning, or Deep Learning. However, in the reverse scenario, where abstracts from pure AI journals are examined, the presence of entities from allied AI areas is significantly less common. The TKDD journal was found to contain representations from all four domains, with the pure AI domain being the least dominant among them.

An inherent challenge we discovered with the annotation of scientific documents is that quite often a term is presented in a descriptive manner, which makes specifically demarcating the keyphrases quite a challenging task.

Dataset	Model	F1
SciERC (Span)	SpanBERT-Tagger	78.77
Few-TK (Span)	SpanBERT-Tagger	67.35 ↓
SciERC (Span)	SciBERT-Tagger	78.44
Few-TK (Span)	SciBERT-Tagger	69.15 ↓

Table 9: Performance of SciBERT and SpanBERT on SciERC and Few-TK datasets for detection of keyphrase spans.

In the fully supervised setting, we have additionally evaluated the ability of supervised models to detect span-level mentions by tasking the model with predicting only the keyphrase spans. We observe in Table 9 that both SpanBERT and SciBERT taggers perform similarly on each of the datasets, based on the span-level F1 scores. The performance of the Few-TK dataset is significantly lower than that of SciERC (Luan et al., 2018). Therefore, we infer that the detection of scientific spans in our dataset is more challenging and warrants greater attention from the community to enhance algorithms tailored for such scientific data.

11 Conclusion

We have developed a unique dataset tailored for the task of few-shot scientific keyphrase recognition within the scientific domain. We have also evaluated various models on it to assess its credibility as a challenging dataset. We hope that this dataset will be used as a cornerstone in research on scientific Typed Keyphrase recognition.

Acknowledgments

This work is partially supported by the SERB-DST Project CRG/2021/000803 sponsored by the Department of Science and Technology, Government of India at the Indian Association for the Cultivation of Science, Kolkata.

[2] `https://paperswithcode.com/`

Limitations

One challenge that remains is that we do not annotate discontinuous spans as a single keyphrase. For example, consider the sentence "...rule-based and neural models". One may wish to identify two separate keyphrases "rule-based models" and "neural models", but here we extract "rule-based" and "neural models" as the two keyphrases because including "models" in the first keyphrase makes the annotation process cumbersome and also introduces additional challenges for the learning algorithms. However, we aim to address this issue in a future work.

References

Isabelle Augenstein, Mrinal Das, Sebastian Riedel, Lakshmi Vikraman, and Andrew McCallum. 2017. SemEval 2017 task 10: ScienceIE - extracting keyphrases and relations from scientific publications. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 546–555, Vancouver, Canada. Association for Computational Linguistics.

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.

Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca Passonneau, and Rui Zhang. 2022. CONTaiNER: Few-shot named entity recognition via contrastive learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6338–6353, Dublin, Ireland. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. 2021. Few-NERD: A few-shot named entity recognition dataset. 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 3198–3213, Online. Association for Computational Linguistics.

Jennifer D'Souza and Sören Auer. 2021. Pattern-based acquisition of scientific entities from scholarly article titles. In *Towards Open and Trustworthy Digital Societies*, pages 401–410, Cham. Springer International Publishing.

Jennifer D'Souza and Sören Auer. 2022. Computer science named entity recognition in the open research knowledge graph. *arXiv preprint arXiv:2203.14579*.

Jennifer D'Souza, Sören Auer, and Ted Pedersen. 2021. SemEval-2021 task 11: NLPContributionGraph - structuring scholarly NLP contributions for a research knowledge graph. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 364–376, Online. Association for Computational Linguistics.

Jennifer D'Souza, Anett Hoppe, Arthur Brack, Mohmad Yaser Jaradeh, Sören Auer, and Ralph Ewerth. 2020. The STEM-ECR dataset: Grounding scientific entity references in STEM scholarly content to authoritative encyclopedic and lexicographic sources. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2192–2203, Marseille, France. European Language Resources Association.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR.

G.D. Forney. 1973. The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Sonal Gupta and Christopher Manning. 2011. Analyzing the dynamics of research by extracting key aspects of scientific papers. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1–9, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Yufang Hou, Charles Jochim, Martin Gleize, Francesca Bonin, and Debasis Ganguly. 2019. Identification of tasks, datasets, evaluation metrics, and numeric scores for scientific leaderboards construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5203–5213, Florence, Italy. Association for Computational Linguistics.

Yutai Hou, Wanxiang Che, Yongkui Lai, Zhihan Zhou, Yijia Liu, Han Liu, and Ting Liu. 2020. Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1381–1393, Online. Association for Computational Linguistics.

Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 216–223.

Sarthak Jain, Madeleine van Zuylen, Hannaneh Hajishirzi, and Iz Beltagy. 2020. SciREX: A challenge dataset for document-level information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7506–7516, Online. Association for Computational Linguistics.

Salomon Kabongo, Jennifer D'Souza, and Sören Auer. 2021. Automated mining of leaderboards for empirical ai research. In *Towards Open and Trustworthy Digital Societies*, pages 453–470, Cham. Springer International Publishing.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. SemEval-2010 task 5 : Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 21–26, Uppsala, Sweden. Association for Computational Linguistics.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232, Brussels, Belgium. Association for Computational Linguistics.

Tingting Ma, Huiqiang Jiang, Qianhui Wu, Tiejun Zhao, and Chin-Yew Lin. 2022. Decomposed meta-learning for few-shot named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1584–1596, Dublin, Ireland. Association for Computational Linguistics.

Rui Meng, Sanqiang Zhao, Shuguang Han, Daqing He, Peter Brusilovsky, and Yu Chi. 2017. Deep keyphrase generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 582–592, Vancouver, Canada. Association for Computational Linguistics.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30:3–26.

Georgios Petasis, Alessandro Cucchiarelli, Paola Velardi, Georgios Paliouras, Vangelis Karkaletsis, and Constantine D. Spyropoulos. 2000. Automatic adaptation of proper noun dictionaries through cooperation of machine learning and probabilistic methods. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, page 128–135, New York, NY, USA. Association for Computing Machinery.

Behrang QasemiZadeh and Anne-Kathrin Schumann. 2016. The ACL RD-TEC 2.0: A language resource for evaluating term extraction and entity recognition methods. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1862–1868, Portorož, Slovenia. European Language Resources Association (ELRA).

Tokala Yaswanth Sri Sai Santosh, Debarshi Kumar Sanyal, Plaban Kumar Bhowmick, and Partha Pratim Das. 2020. DAKE: Document-level attention for keyphrase extraction. In *Advances in Information Retrieval: 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14–17, 2020, Proceedings, Part II 42*, pages 392–401. Springer.

Tokala Yaswanth Sri Sai Santosh, Nikhil Reddy Varimalla, Anoop Vallabhajosyula, Debarshi Kumar Sanyal, and Partha Pratim Das. 2021. HiCoVA: Hierarchical conditional variational autoencoder for keyphrase generation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3448–3452.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

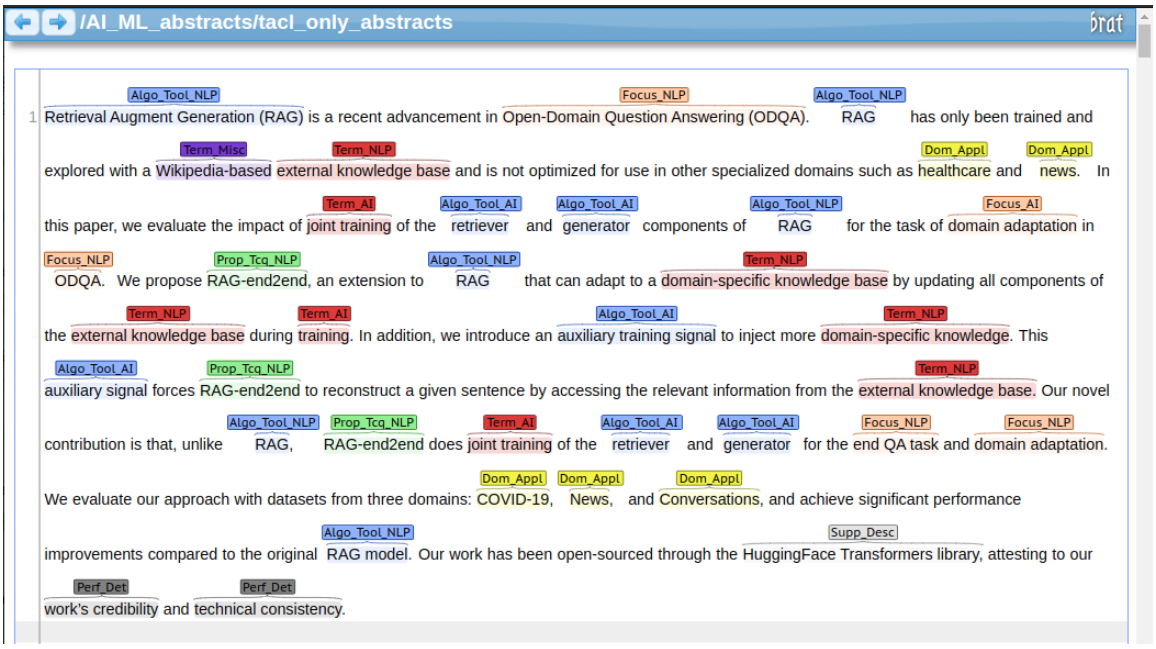
Yisheng Song, Ting Wang, Puyu Cai, Subrota K Mondal, and Jyoti Prakash Sahoo. 2023. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ACM Comput. Surv.* Just Accepted.

Santosh Tokala, Debarshi Kumar Sanyal, Plaban Kumar Bhowmick, and Partha Pratim Das. 2020. SaSAKE: Syntax and semantics aware keyphrase extraction from research papers. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5372–5383, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3).

Vikas Yadav and Steven Bethard. 2018. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yi Yang and Arzoo Katiyar. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6365–6375, Online. Association for Computational Linguistics.



Retrieval Augment Generation (RAG) is a recent advancement in Open-Domain Question Answering (ODQA). RAG has only been trained and explored with a Wikipedia-based external knowledge base and is not optimized for use in other specialized domains such as healthcare and news. In this paper, we evaluate the impact of joint training of the retriever and generator components of RAG for the task of domain adaptation in ODQA. We propose RAG-end2end, an extension to RAG that can adapt to a domain-specific knowledge base by updating all components of the external knowledge base during training. In addition, we introduce an auxiliary training signal to inject more domain-specific knowledge. This auxiliary signal forces RAG-end2end to reconstruct a given sentence by accessing the relevant information from the external knowledge base. Our novel contribution is that, unlike RAG, RAG-end2end does joint training of the retriever and generator for the end QA task and domain adaptation. We evaluate our approach with datasets from three domains: COVID-19, News, and Conversations, and achieve significant performance improvements compared to the original RAG model. Our work has been open-sourced through the HuggingFace Transformers library, attesting to our work's credibility and technical consistency.

Figure 4: Example of annotation of an abstract from FEW-TK in BRAT. This the same abstract as shown in Table 1.

A BRAT-based Annotation for FEW-TK

Figure 4 illustrates an annotated abstract (from our proposed dataset) in BRAT, a web-based tool designed for text annotation.

B Annotation Guidelines

The annotators were told to follow the keyphrase boundaries or spans following the annotation guidelines in ACL RD-TEC Annotation Guideline-ver 2 (`https://github.com/languagerecipes/acl-rd-tec-2.0/blob/master/distribution/documents/acl-rd-tec-guidelines-ver2.pdf`). We started with 45 fine-grained keyphrase types after brainstorming and discussions and after annotation merged the types that did not have a significant number of keyphrases.

B.1 Keyphrase Types

The description for the keyphrase types used for this dataset are as follows:

- **Focus**: This coarse-grained keyphrase type refers to the intent of the scientific document or article. Please note that a phrase is considered to be in this category only when it is the main theme of the paper or is a domain-specific task.

  AI/ML/DL focus refers to the main intent of the article that pertains to classical Artificial Intelligence or Machine Learning or Deep Learning.

  E.g.: continual learning, clustering

  Computer Vision focus refers to the focus of an article that is primarily related to Computer Vision.

  E.g.: visual identification, action recognition

  NLP focus is considered the focus of a paper that is primarily related to Natural language Processing.

  E.g.: text classification, sequence tagging

  Data Mining/Information Retrieval focus implies the main topic of the paper relates to Data Mining or Information Retrieval.

  E.g.: Graph-based Multi-View Clustering (GMVC), contextual bandit learning

  Miscellaneous focus refers to a theme or domain-specific task in an article that does not fit into any of the above-mentioned fine-grained categories.

  E.g.: Transportation demand forecasting, optimal online transportation

- **Proposed Technique**: This coarse-grained keyphrase type is used for those keyphrases which mention a method that has been proposed in the given document. This category especially refers to the name of the new method that is proposed, if any.

AI/ML/DL-based technique refers to a method put forward by the article that is used to solve a problem in classical Artificial Intelligence or Machine Learning or Deep Learning.

E.g.: Dual-MGAN, CoarsenRank

NLP-based technique is a technique presented in the article that is used to solve a Natural Language Processing task.

E.g.: Target-Guided Structured Attention Network (TG-SAN), Question Decomposition Meaning Representation (QDMR)

Computer Vision-based technique is a technique proposed in the article as a solution to a Computer Vision problem.

E.g.: SegNet, adaptive two-stream consensus network (A-TSCN)

Data Mining/Information Retrieval-based technique is a technique proposed in a article to solve a problem in Data Mining or Information Retrieval.

E.g.: dual subgraph-based pairwise graph neural network (DSGNN), Spatio-Temporal Heterogeneous graph Attention Network (STHAN)

- **Algorithm/Tool**: It refers to a pre-existing concept or algorithm that has been used in the research article.

  AI/ML/DL algorithm/tool is some algorithm that has been well established and is being used in almost all areas of Artificial Intelligence or Machine Learning or Deep Learning.

  E.g.: variational autoencoders, Bayesian PDE-constrained framework, logistic regression

  Statistical/Mathematical algorithm/tool is any existing statistical or mathematical tool or theorem or algorithm that has been referred to in the article.

  E.g.: Factorial hidden Markov models, symbolic Bayesian model

  Computer Vision algorithm/tool is any existing algorithm or tool that is solely used in the domain of Computer Vision.

  E.g.: 3D CNN model, Discriminative Correlation Filters (DCFs)

  NLP algorithm/tool is any existing algorithm or tool that is solely used in the domain of Natural Language Processing.

  E.g.: neural language generation models, Transformer language models

  Data Mining/Information Retrieval algorithm/tool is any existing algorithm or tool that is solely used in the domain of Data Mining/Information Retrieval.

  E.g.: structural neighbor aggregation. LBSNs

- **Study Domain**: This category includes mentions of the domain on which the article is based.

  AI/ML/DL domain is used when the domain name pertains very closely to Artificial Intelligence or Machine Learning or Deep Learning.

  E.g.: Geometric Deep Learning, machine Learning

  Computer Vision domain is used when domain name pertains to Computer Vision.

  E.g.:Computer Vision, image processing

  NLP domain refers specifically to the broad domain of Natural Language Processing.

  E.g.: Natural Language Understanding, NLP

  Data Mining/Information Retrieval domain

  E.g.: data mining, information retrieval

  Application domain refers to the applied domain for which the tool or algorithm or technique that has been proposed in the paper is presented.

  E.g.: COVID-19 News, sports competitions recommendations, healthcare, news

- **Supplementary Material**: This category contains the supplementary material that has been presented with the text.

  URL specifically refers to the URL to the code or dataset or any other material that is present in the article.

  E.g.: https://tpami.wmflabs.org, https://github.com/WayneWong97/CSDia

  Material Description is a phrase or word that describes the supplementary material provided in the article.

  E.g.: 170,000+ documents, 2–4 hop questions

Miscellaneous material alludes to references or any other material that has been presented with the paper.

E.g.: CRAN, DoubleML

- **Dataset**: It refers to the dataset name.

  AI/ML/DL dataset refers to a dataset that is primarily of generic use in Artificial Intelligence or Machine Learning or Deep Learning, and not meant for a specific use-case.

  E.g.: UniRef, BFD

  Computer Vision dataset alludes to a dataset that is used for a Computer Vision task.

  E.g.: ImageNet Large Scale Visual Recognition Challenge (ILSVRC), ImageNet 2012

  NLP dataset alludes to a dataset that is used for a Natural Language Processing task.

  E.g.: CFQ, FeTaQA

- **Metric**: This label refers to the keyphrases which represent different metrics.

  Classification metrics are the metrics that are used to measure the correctness of data classification.

  E.g.: accuracy, Macro F1

  Statistical/Mathematical metrics refer to quantitative metrics.

  E.g.: mIOU, Normalized Discounted Cumulative Gain (NDCG)

  NLP metrics refers to the metrics that are solely used in Natural Language Processing.

  E.g.: dialog act segmentation error rates (DSER), BLEU

  Miscellaneous metrics are those metrics that do not fall in any of the above metric categories.

  E.g.: signal-to-background ratio (SBR), human aggregate agreement

- **Allied Terms**: These are the terms which

  AI/ML/DL term refers to a term from the classical Artificial Intelligence or Machine Learning or Deep Learning domain. It refers to any term that is neither a task nor a technique nor a dataset in the present context.

  E.g.: model architecture, regularization parameter kernel

  Statistical/Mathematical term alludes to any technical term that belongs to Statistical or Mathematical domain. This is useful because AI/ML research articles generally refer to many Statistical/Mathematical terminologies.

  E.g.: probability, equivariance

  Computer Vision term refers to any term that does not belong to any of the above-mentioned categories but is an important terminology related to Computer Vision.

  E.g.: full-image convolutional features, coded exposure image

  NLP term alludes to any term that does not belong to any of the above-mentioned categories but is an important terminology related to Natural Language Processing.

  E.g.: entities, phonological

  Data Mining/Information Retrieval term alludes to any term that does not belong to any of the above-mentioned categories but is an important terminology related to Data Mining or Information Retrieval.

  E.g.: temporal nonlinear sparsity weak serial correlation, linkage quality

  Miscellaneous term is any term that does not fall under any of the above-mentioned categories but is still deemed important in the context of the paper.

  E.g.: model complexity, computational bottlenecks

- **Performance**: This category captures the performance-related information reported in the document.

  Numerical Performance alludes specifically to the results or such data that has been presented with numerical figures. It could be the quantitative value of any metric such as the F1 score.

  E.g.: 18.01, 63.69

  Performance Descriptor refers to any phrase that describes the performance in words.

  E.g.: top-1, inference time reduction

C Coarse-grained and Fine-grained Keyphrase Types

Table 10 shows the full list of proposed coarse-grained and fine-grained keyphrase types in FEW-TK dataset.

Coarse-grained Keyphrase Type	**Fine-grained Keyphrase Type**
Focus	AI/ML/DL focus Computer Vision focus NLP focus Data Mining/Information Retrieval focus Miscellaneous focus
Proposed Technique	AI/ML/DL-based technique Computer Vision-based technique NLP-based technique Data Mining/Information Retrieval-based technique
Algorithm/Tool	AI/ML/DL algorithm/tool Statistical/Mathematical algorithm/tool Computer Vision algorithm/tool NLP algorithm/tool Data Mining/Information Retrieval algorithm/tool Miscellaneous algorithm/tool
Study Domain	AI/ML/DL domain Computer Vision domain NLP domain Data Mining/Information Retrieval domain Application domain
Supplementary Material	URL Material Description Miscellaneous material
Dataset	AI/ML/DL dataset Computer Vision dataset NLP dataset
Metric	Classification metrics Statistical/Mathematical metrics NLP metrics Miscellaneous metrics
Allied Terms	AI/ML/DL term Statistical/Mathematical term Computer Vision term NLP term Data Mining/Information Retrieval term Miscellaneous term
Performance	Numerical Performance Performance Descriptor

Table 10: Combined list of all the coarse-grained keyphrase types and their corresponding fine-grained sub-types in FEW-TK.