Dataset mention extraction for AI-Related literature

Qi Zhang (qi.zhang@temple.edu), Zhuoan Zhou (zhuoan.zhou@temple.edu)

May 3, 2022

Abstract

In this project, we study the dataset names mention extraction task in AI-related literature as a Named Entity Recognition (NER) problem. Two research questions have been explored, which NER model has the best performance in dataset names NERT task and how will the amount of training data influence the models' performance. Many popular sequence labeling models including CRF, BiLSTM, embedding techniques, BERT and SciBERT have been conducted in our experiments. On all measures, the BERTs NER tagger performed best and most robustly. All the code of this project has been released, you can find in https://github.com/edzq/AI-Project.

1 Introduction

In the past decades, the research of Artificial Intelligence (AI) experienced a wave of high speed growth. Therefore, the number of scientific literature is also growing exponentially. Extracting scientific information from literature is very important for scientific text mining, which has many applications like building academic knowledge graph.

Dataset mentioned in papers is one of the most important information that researchers and government are interested in [2]. Firstly, understanding how dataset is can help the public and government improve the benefits of investment of creating dataset. Since we all know creating dataset is very time-consuming and expensive. Secondly, extracting dataset used in literature is critical for building dataset citation network, which could help researchers build work one others' foundation. Furthermore, dataset as one of the most important entities in

literature, the extraction work of it is very important for scientific text mining. In this wave of AI, data is one of the most important drivers [1]. So understanding how scientists use the dataset is critical for AI research.

This project we study the task of recognizing named dataset in AI-related papers as the Name Entity Recognition problem (NER). NER is a sub-task of information extraction that locate and classify named entities mentioned in text data like person, location and date [3]. In literature mining field, the entities can be methods, tasks, dataset and software and et al.. As the figure 1 shows, these entities extraction are all domain-specific NER problem. There are many challenges of dataset NER task. Firstly, dataset appears infrequently in documents and is only relevant in specific knowledge domains. Furthermore, entity resolution in dataset names is challenging since dataset references to the same dataset may vary widely across documents. For example, DBLP dataset could be referred as "DBLP Bibliography" or "DBLP dataset". Many dataset can also be used as abbreviation or full name.

In this project, in order to show which NER model fits the dataset NER task, we we compare the performance of many NER models based on one dataset mention corpus. The NER models we would like to explore including Conditional random field (CRF), bidirectional LSTM (BiLSTM) with word2vec embedding and without wor2vec embedding, BERT [4] and SciBERT [5]. In a word, we discuss two explore questions:

- EQ1: What is the performance of CRF, BiLSTM, BERT and SciBERT models on the dataset name extraction task?
- EQ2: How does the amount of training data impact the models' performance?

To answer these two questions, we trained these models on one large dataset name corpus. We also create one zero-shot set to evaluate these models performance. Then we vary the number of training data and test performance on zero-shot dataset.

All code used for this project can be found at https://github.com/edzq/AI-Project.

2 Related word

The work of this project is related to the NER in scientific. Scientific entity extraction and information extraction (IE) work can be found since early 1990s [6]. Since comparing to the general NER problem, the research in scientific NER is limited because of set of challenges. The main challenge is domain expertise is always needed for annotating to create available

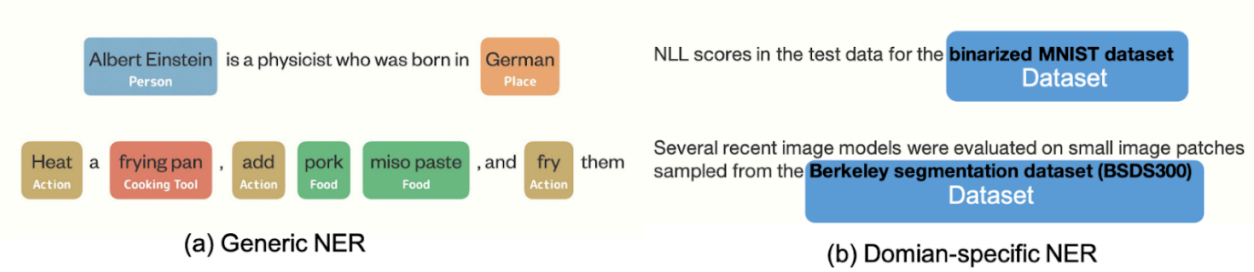


Figure 1: NER task in NLP

corpus. So for a long time, despite the growing interest in the scientific IE, research on this field is very narrow even now [7]. One famous work in scientific NER topic is GROBID [9], which is an open source software used widely in scientific IE. This work leverages the CRF model to extracte bibliographic data (such as like title, reference etc.) from scientific texts. However, in the medicine and biology fields, there are many researchers focusing on this scientific IE work [8].

For dataset extraction, many works use a great variety of methods can be found. But the research of this topic is unmanageable, which means the standards of the labeled dataset are vary from different papers. Heddes et al [1] released one mentioned dataset name corpus in 2020 and proposed one rule-based algorithm, which 0.72 exactly matching (this metric is the nervaluate [10]). But their dataset entities are very special. Ghavimi et al [11] studied the dataset extraction in social science field. They proposed one semi-automatic approach contains the BiLSTM-CRF model, which reached the 0.85 F1-score.

To date, many works using BERT to extact the dataset references but the performances differ greatly. Hou, Y et al [12] fine-tuned the BERT and reached 0.68 F1 score. However, Zhao, H et al [13] also leveraged the BERT and got the F1 score of 0.79. Heddes et al [1] tested both BERT and SciBERT models and reached 0.77 and 0.78 exact math score. Kaggle also held one competition in 2021 [14], but this challenge focused on the social science dataset.

Therefore, the research on dataset name recognition task uses many NER models including but not limited: CRF, rule-based, BiLSTM, BERT and SciBERT. But this task lack the public standard dataset. Different researchers also use different metrics.

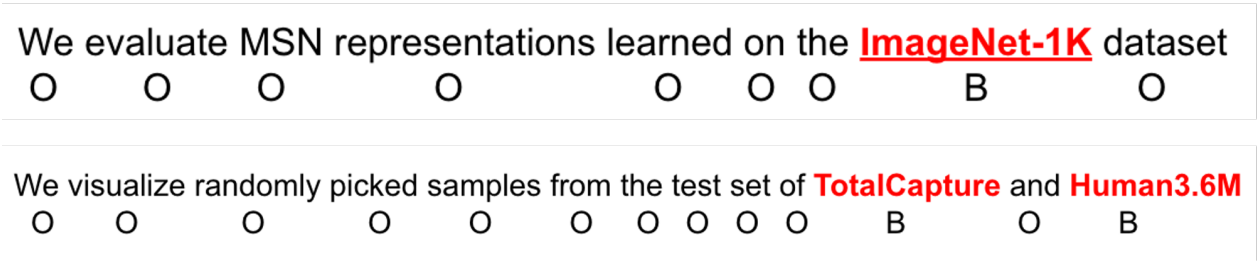


Figure 2: BIO tagging examples. Target entity will be labeled as "B" and "I", others will be labeled as "O".

3 Methodology

3.1 Dataset

3.1.1 Dataset overview

This dataset is provied by Qi's research group, now his research group works on the automatically generate large scale dataset mention detection corpus in AI-related literature. Please note that only the dataset is provided by the research group, all experiment designs and works in this project are finished by Qi and Zhuoan.

This dataset contains 210,185 sentences with BIO tagging from 25,450 papers, which are mainly from Arxiv (78%), CVPR (5%), ICCV (3%) and ICASSP (1%) et al. So most of these papers are from the top tier AI-related conferences and journals. BIO tagging means the target entity is labeled as "B" (if only one token) and "I" (if more than 2 tokens), others are labeled as "O". Please see the two examples in the figure 2.

In total there are 1,904 dataset names in the corpus. The figure 3 shows the distribution of the dataset names. 35% (677) are being mentioned more than 40 times in the papers. The the left figure in 3 shows most of the dataset has been mentioned 1 20 times in all the papers. The right figure in 3 shows most of the dataset names occurs less than 8 times one average in the papers mentioning them.

3.1.2 Dataset split strategies

Since we plan to test whether the trained models can be generalized engout even when dealing with unseen dataset names, we crated the zero-shot set. The dataset names in the zero-shot

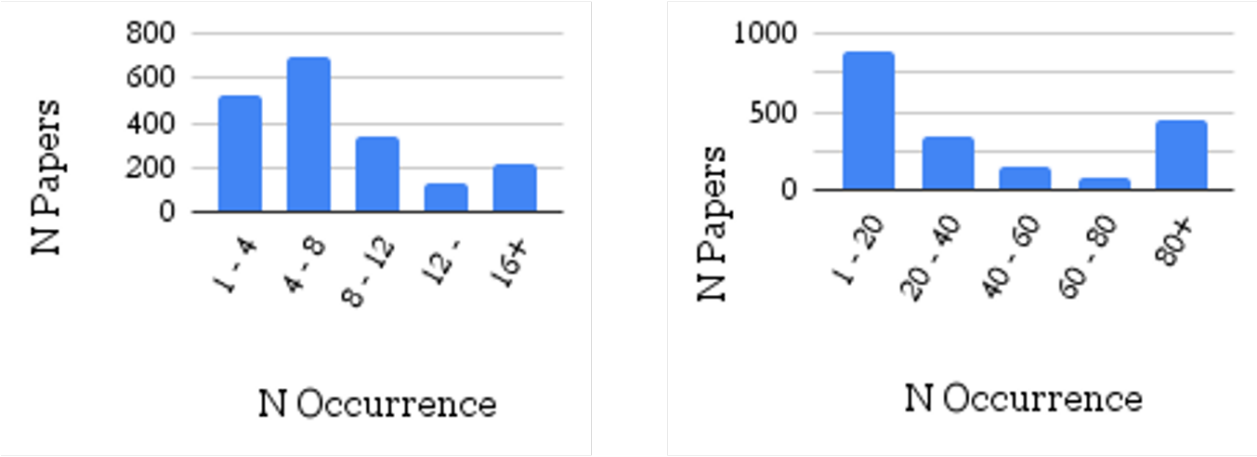


Figure 3: Dataset name occurrence distribution statistic.

set should never appear in other sets. We randomly selected 150 dataset names and then found all the papers (824 papers) mentioned these dataset. Finally, we got one zero-shot set contains 7,608 sentences.

Then we splitted the rest samples randomly. There are 146,580 (69.74%) sequences in training, 21,217 (10.09%) sequences in validation and 42,388 (20.17%) in test. So we get unseen group is the group of zero-shot set and seen group is the group of sequences with dataset names seen from the training set.

Please note: since the dataset we used in this project is from one corpus paper of Qi's research group, we cannot open source this dataset.

3.2 Models and experimental setup

NER can be done by using a number of sequence labeling methods. In this part, we go through all the models that used in our project and briefly describe how we implement them.

3.2.1 CRF

CRF, short for Conditional Random Fields, is the most prominent approach and used to be the state-of-the-art (SOTA) model in NER [15]. Since CRF considers the context unlike the traditional classifiers predicting labels without taking "neighbouring" samples into account, it has been widely used for structured prediction. In NER task, we use the linear chain CRF,

which is popular in NLP predicts sequences of labels for sequences of input samples.

Before using the CRF models, we need to add some features as the input to enhance the performance. Since context is taken into account, POS tags for each word have been added. Also, features like capitalizations, types of word (title, uppercase, lowercase, digital etc.) are added.

For implement, we use the sklearn-crfsuite from the scikit-learn library [16].

3.2.2 BiLSTM and embedding

LSTM, long short-term memory, is often used in NLP problems. BiLSTM, bi-directional LSTM, combines forward LSTM and backward LSTM. Compared to the traditional LSTM, BiLSTM is more stable and consistent. However, it has higher computational cost. LSTM is more used on text generation. BiLSTM usually performs better than LSTM on NER problems. We use two different word embedding techniques with BiLSTM. One of them is Glove, Global Vectors for word representation. Glove is count-based, and it calculates a co-occurrence probabilities matrix[20]. Word2Vec is predictive-based, and it vectorizes every word. However, each word can only have one vector. In our implement, we use the Keras framework. Used parameter setup were taken from the [17] and no parameter optimization was used in all BiLSTM models. All variations of BiLSTM are trained with 5 epochs. For fair comparison, the three models all use the 300 as the embedding dimension. BiLSTM-GLove, we used the embedding trained on Wikipedia and Gigaword and converted 36,332 tokens. However, 137,123 tokens are missed from the pretrained embeddings and thus initialized with zeros. We think this is because most of the dataset naes are missed from the pretrained embeddings. BiLSTM-Word2Vec, we used the embedding trained on Google news, and converted 47,772 tokens while 125,683 are missed.

3.2.3 BERT and SciBERT

BERT, short for Bidirectional Encoder Representations from Transformers, is a transformer-based pre-trained approach proposed by Google in 2018. BERT is praised for its context-aware word representations improving the prediction ability in may downstream tasks [4]. Even though the reasons of why the BERT reaches SOTA performance in many NLP tasks are not yet well illustrated, BERT has revolutionized classical NLP. SciBERT is trained on 1.14M scientific papers from Semantic Scholar, which consisting of 18% computer science papers [5].

In our implement, both BERT and SciBERT are based on a scikit-learn wrapper [18]. We choose case-sensitive model of BERT, since most of the dataset are referred as capitalized. For fair comparison, since SciBERT only has base model, BERT also used the BERTbase model. The configuration of the SciBERT is same as the paper [5]. Both of them use the following hyperparameters: 1) learning rate set as 5×10^{-5} for the Adam optimizer; 2) batch size set as 16; 3) max sequence length set as 178; 4) training epochs set as 3; 5) gradient clipping was used and set max gradient as 4.

4 Experiment results

We report the results grouped by the two explore questions mentioned in the beginning. For the evaluation metrics, we use the precision, recall and F1-score. In order to keep all the experiments in different models fair, we did not update the training parameters using the validation performance. It means we just used the validation set as an extra set for evaluation.

All of the experiment conducted in the high performance computing of Temple University, which is an interactive server provides 16 CPU cores and 512GB of RAM, and 4x NVIDIA Tesla V100 GPUs.

4.1 EQ1: Overall performance

The experiments of EQ1 are training and testing all the models and comparing their performance. Table 4.1 contains the six models' experiments result in validation set, test set and unseen (zero-shot) set. From these results, we observe that the best model for precision, recall and F1 Score of predicting the sequence label of dataset name in all of the tree test set is BERT. Specifically, BERT get 0.91 F1 score in validation set, 0.92 F1 score in test set and 0.80 in zero-shot set (unseen). It is very surprising that BERT is better than SciBERT, since SciBERT is trained on more scientific corpus. But the performances of the SciBERT and BERT are very close. For the BiLSTM, with pre-trained embedding models has better performance than without pre-trained embedding. And the overall performances of Google Wor2vec are better than the GLove.

Therefore, for the EQ1, we find that BERT and SciBERT are most suitable for the dataset names NER trask, and they also reach the best performance on unseen set testing, which reflects the generalization of them.

Table 1: The overall performance of different NER models

	Models	Precision	Recall	F1
Validation N=21,217	CRF	0.90	0.81	0.85
	BiLSTM	**0.92**	0.77	0.84
	BiLSTM-Word2vec	0.91	0.85	0.88
	BiLSTM-GLove	0.90	0.84	0.87
	BERT	0.91	**0.91**	**0.91**
	SciBERT	0.91	0.90	0.90
Test: Seen N=34,780	CRF	0.90	0.86	0.88
	BiLSTM	**0.93**	0.81	0.87
	BiLSTM-Word2vec	0.91	0.89	0.90
	BiLSTM-GLove	0.91	0.88	0.89
	BERT	0.92	**0.93**	**0.92**
	SciBERT	0.91	0.91	0.91
Unseen N=7,608	CRF	**0.89**	0.63	0.73
	BiLSTM	0.88	0.58	0.70
	BiLSTM-Word2vec	**0.89**	0.65	0.75
	BiLSTM-GLove	0.85	0.64	0.73
	BERT	0.86	**0.76**	0.81
	SciBERT	0.86	0.75	**0.80**

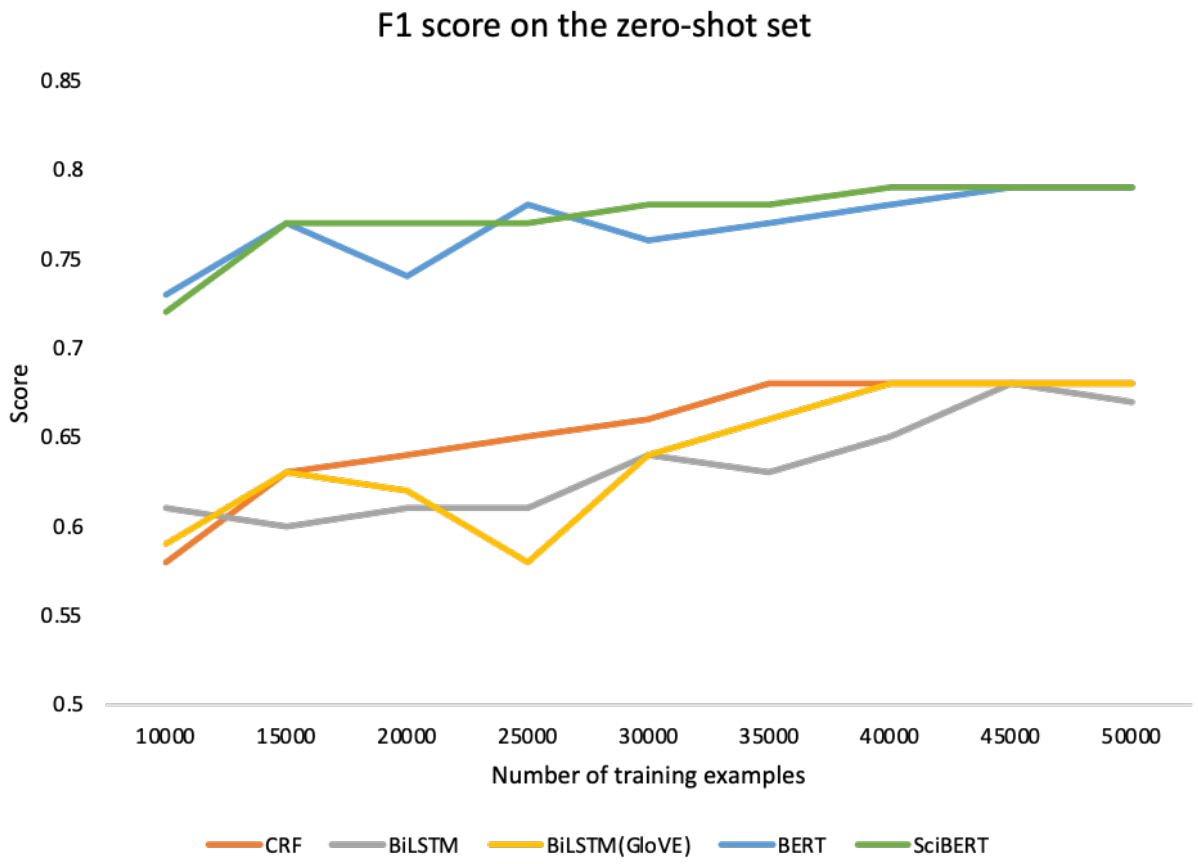
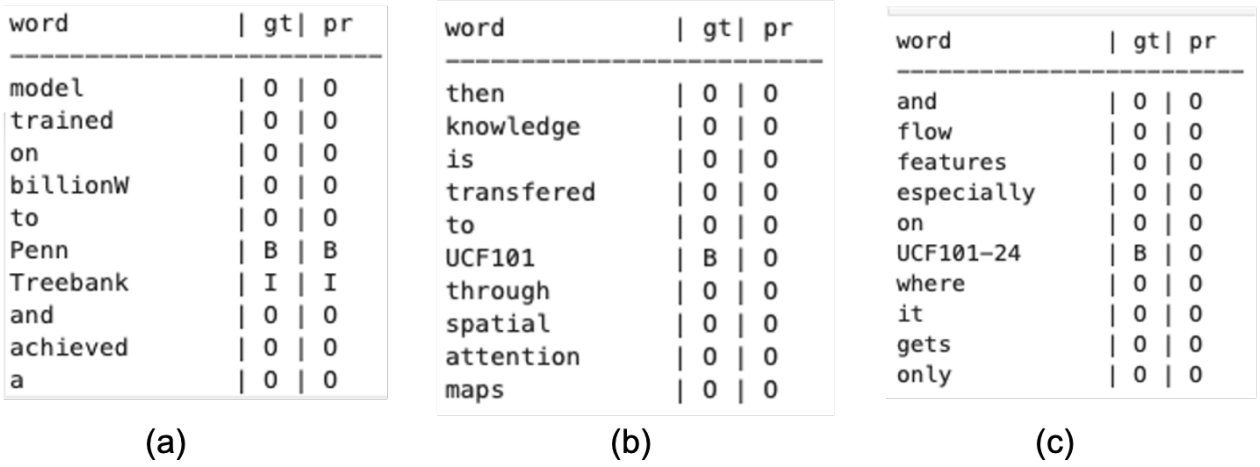


Figure 4: F1-score when changing the amount of training data.

4.2 EQ2:Amount of training data

In this explore problem, we consider the major factor of the performance: the training data size. To finish EQ2, we need to change the amount of the training samples and test the performance on zero-shot set (unseen part in the table 4.1). Since the training time of so many models is very long, we just used the training data size ranges from 10,000 to 50,000 and trained the model on each time add 5,000 samples, which means we will have 9 data points of each model. Totally, we trained CRF, BiLSTM, BiLSTM with GLove embedding, BERT and SciBERT.

Figure 4 shows how the F1 scores of each model changing as the number of training data changing. We can find clear gap between two BERT models and others. These are same as the results showed in table 4.1 unseen set testing. We also find that the performances of



word	gt	pr
model	O	O
trained	O	O
on	O	O
billionW	O	O
to	O	O
Penn	B	B
Treebank	I	I
and	O	O
achieved	O	O
a	O	O

(a)

word	gt	pr
then	O	O
knowledge	O	O
is	O	O
transfered	O	O
to	O	O
UCF101	B	O
through	O	O
spatial	O	O
attention	O	O
maps	O	O

(b)

word	gt	pr
and	O	O
flow	O	O
features	O	O
especially	O	O
on	O	O
UCF101-24	B	O
where	O	O
it	O	O
gets	O	O
only	O	O

(c)

Figure 5: Three testing examples with the SciBERT

BiLSTM with GLove embedding are not stable.

Therefore, the results of EQ2 show the robust behavior of the BERT and SciBERT on dataset names NERT task.

4.3 NER examples of the SciBERT

We further evaluate the example prediction. Since the predicted results are similar across models, we only show the results of the SciBERT. Detail please see the figure 5. The Figure 5 (a) is one positive example, which successfully prediction (pr in the figure) all the ground truth (gt in the figure) for both "B" and "I" tags. So in this example, the ground truth is "Penn" and "Trebank". The (b) and (c) in figure 5 are the negative examples. These two examples are very similar, one ground truth is "UCF101", another is "UCF101-24". The SciBERT. Our trained SciBERT fails in both of them. We will discuss the reason in the next section.

5 Conclusion and discussion

In this project, we study the dataset names extraction task as NER problem in literature. We explore two questions, one is which NER model fits this task best, another is how the amount of training dataset affect the models' performance. Not surprisingly, in both EQ1 and

EQ2, the BERT and SciBERT have the best performance in the dataset mention extraction task. We also find that the embedding has influence for the perfromance of BiLSTM models. Embedding improves the performance. From these two conclusions, we can infer that at least in this dataset and experiments, pre-train paradigm is better.

From this project, we also found some challenges of this task:

- Dataset version: like what we show in the examples in section 4.3, our trained models fail in many samples because the dataset version. For example, "UCF101" and "UCF101-24" are quite similar. The trained models seem be confused by these two named entities. It is very normal to name dataset with the numeric suffixes to represent the version.
- Ambiguation: dataset name as entities also have the problem of ambiguation. For example, ImageNet is not only one dataset, it is also one task (challenge), so the context to recognize them is very important. This is also very challenging for this task, since there are many other examples like question answer dataset and question answer task.
- Training time: the training processing at least costed us 60 hours. And, this is the training time on the university's high performance computing facility. So it is very hard to finish all of these experiments without GPU.

We also think that there are some interesting future works to explore more of this task:

- Exploring the domain adaption ability of these model. It means we can further split the papers in the dataset as computer vision (CV), natural language processing (NLP) and others. Then we training models in CV and testing in NLP domain.
- Exploring this task in the biological, chemical and medical fields, since there are more entities in these domains.
- Exploring more models like LSTM-CRF and BERT-CRF.

6 Acknowledge

This project thanks to the research group of Prof. Eduard Dragut, who provides this dataset for us. We also thanks to the Jo Pan, who provides the splitting strategies and some of the

utils code. Some code of this project references the [19], thanks to the authors. Furthermore, thanks to the High performance computing (HPC) of Temple university, without this HPC we cannot finish this project.

References

[1] Heddes, Jenny, et al. "The automatic detection of dataset names in scientific articles." Data 6.8 (2021): 84.

[2] Kratz, J.E.; Strasser, C. Researcher perspectives on publication and peer review of data. PLoS ONE 2015, 10, e0117619.

[3] https://en.wikipedia.org/wiki/Named-entity_recognition

[4] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[5] Beltagy, Iz, Kyle Lo, and Arman Cohan. "SciBERT: A pretrained language model for scientific text." arXiv preprint arXiv:1903.10676 (2019).

[6] Mohit, Behrang. "Named entity recognition." Natural language processing of semitic languages. Springer, Berlin, Heidelberg, 2014. 221-245.

[7] Luan, Yi, et al. "Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction." arXiv preprint arXiv:1808.09602 (2018).

[8] Tchoua, R.; Ajith, A.; Hong, Z.; Ward, L.; Chard, K.; Audus, D.; Patel, S.; de Pablo, J.; Foster, I. Towards hybrid human-machine scientific information extraction. In Proceedings of the 2018 New York Scientific Data Summit, New York, NY, USA, 6–8 August 2018; pp. 1–3.

[9] Lopez, P. GROBID. 2008–2022. Available online: https://github.com/kermitt2/grobid

[10] Github. Full Named-Entity (i.e., Not Tag/Token) Evaluation Metrics Based on SemEval'13. 2019. Available online: https:github.com/ivyleavedtoadflax/nervaluate

[11] Ghavimi, Behnam, et al. "Identifying and improving dataset references in social sciences full texts." arXiv preprint arXiv:1603.01774 (2016).

[12] Hou, Yufang, et al. "Identification of tasks, datasets, evaluation metrics, and numeric scores for scientific leaderboards construction." arXiv preprint arXiv:1906.09317 (2019).

[13] Zhao, He, et al. "A context-based framework for modeling the role and function of on-line resource citations in scientific literature." Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019.

[14] https://www.kaggle.com/c/coleridgeinitiative-show-us-the-data

[15] Sutton, Charles, and Andrew McCallum. "An introduction to conditional random fields." Foundations and Trends® in Machine Learning 4.4 (2012): 267-373.

[16] Sklearn. SklearnCrfsuite. Available online: https://sklearn-crfsuite.readthedocs.io/en/latest/

[17] Depends on the Definition Guide to Sequence Tagging with Neural Networks. 2017. Available online: https://www.depends-on-the-definition.com/guide-sequence-tagging-neural-networks-python/.

[18] Github. Scikit-Learn Wrapper to Finetune BERT 2019. Available online: https://github.com/charles9n/bert-sklearn

[19] https://github.com/xjaeh/ner_dataset_recognition

[20] Jeffrey Pennington, Richard Socher, Christopher D. Manning. "GloVe: Global Vectors for Word Representation." Available online: https://nlp.stanford.edu/projects/glove/