Pretraining Language- and Domain-Specific BERT on Automatically Translated Text

Tatsuya Ishigaki[†] **Yui Uehara**[†‡] **Goran Topić**[†] **Hiroya Takamura**[†]
[†]National Institute of Advanced Industrial Science and Technology, Japan,
[‡]Kanagawa University,
{ishigaki.tatsuya, goran.topic, takamura.hiroya}@aist.go.jp
yuiuehara@kanagawa-u.ac.jp

Abstract

Domain-specific pretrained language models such as SciBERT are effective for various tasks involving text in specific domains. However, pretraining BERT requires a large-scale language resource, which is not necessarily available in fine-grained domains, especially in non-English languages. In this study, we focus on a setting with no available domain-specific text for pretraining. To this end, we propose a simple framework that trains a BERT on text in the target language automatically translated from a resource-rich language, e.g., English. In this paper, we particularly focus on the materials science domain in Japanese. Our experiments pertain to the task of entity and relation extraction for this domain and language. The experiments demonstrate that the various models pretrained on translated texts consistently perform better than the general BERT in terms of F1 scores although the domain-specific BERTs do not use any human-authored domain-specific text. These results imply that BERTs for various low-resource domains can be successfully trained on texts automatically translated from resource-rich languages.

1 Introduction

Domain-specific pretrained language models (LMs), such as SciBERT (Beltagy et al., 2019), are known to perform better on many downstream tasks with texts in the specific domain, such as named entity recognition in biomedical (Li et al., 2016) and relation extraction in chemical domains (Kringelum et al., 2016). This trend has motivated researchers to release many domain-specific LMs for resource-rich domains and languages, specifically in medicine (Alsentzer et al., 2019), biomedicine (Lee et al., 2019), finance (Araci, 2019), and materials science (Gupta et al., 2021). Many of the domain-specific LMs have been trained on corpora consisting of academic papers or articles, which are usually open to the public. However, such open corpora are often not available in non-English languages. Meanwhile, there are a lot of documents that are not open such as internal corporate documents also in non-English languages, which still need to be processed with pretrained LMs.

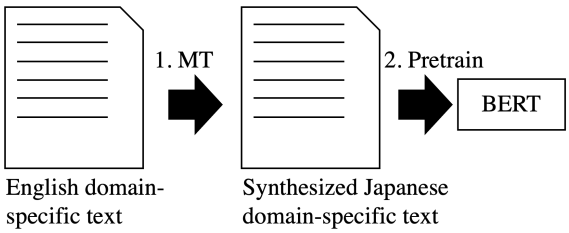

Figure 1: Framework for pretraining that uses language- and domain-specific texts obtained through machine translation.

We focus on a novel setup for pretraining domain-specific BERTs without the use of human-authored domain-specific text. As a solution to the problem, we pretrain LMs on domain-specific text automatically translated from a resource-rich language, i.e., English. As shown in Figure 1, journal papers are automatically translated from English to the target language, e.g., Japanese in this paper, then used in BERT pretraining in different configurations with or without general texts, e.g., Japanese Wikipedia, to investigate the viability on domain-specific Japanese text. Although this is a very simple approach with wide applicability to various domains and languages, the following two questions still need to be answered: 1) is the use of translated text effective in various strategies for pretraining BERT? and 2) does the vocabulary induced from the domain-specific corpus improve performance?

We evaluate our pretrained BERT models on named entity extraction and relation extraction for

Proceedings of Recent Advances in Natural Language Processing, pages 548–555
Varna, Sep 4–6, 2023
https://doi.org/10.26615/978-954-452-092-2_060

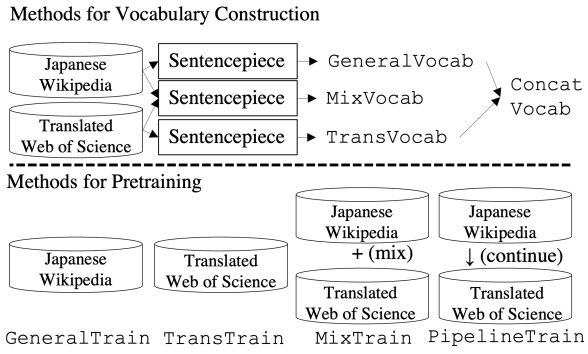

Figure 2: Setups of data and vocabulary used for pretraining the BERTs compared in this paper.

the materials science domain in Japanese due to its high demand. The results empirically show that all the models trained on the translated text consistently achieve better performance than the model trained only on the general text, despite the fact that noise may exist in translation (Artetxe et al., 2020). In addition, we found that the domain-specific vocabularies are effective when BERT is pretrained on a mixture of the two corpora.

Our contributions are: 1) we propose a new setup for pretraining domain-specific BERTs without any human-authored domain-specific text in the target language, 2) we show the effectiveness of the use of translated text for various pretraining strategies, 3) we release the Japanese BERT specific to the domain of materials science and a web-based application of information extractors where even non-NLP experts can benefit from our BERT[1].

2 Related Work

Different types of LLMs have different architectures. For example, BERT (Devlin et al., 2019) has only an encoder, GPT (Brown et al., 2020) adopts decoder-only model, and BART (Lewis et al., 2020) adopts an encoder-decoder architecture. Although GPT-like models are more actively studied recently, we focus on BERT because it is still fundamental to many entity and relation extractors in many domains (Nishida et al., 2023).

Various pretraining methods for BERT have been proposed. The original BERT uses only the general text (Devlin et al., 2019). SciBERT is trained on domain-specific text (Beltagy et al., 2019). Others adapt an LM, pretrained on the general domain, to specific domains by continuing the pretraining on domain-specific text (Wang et al., 2020; Lee et al., 2019; Zhang et al., 2020). Multilingual BERT (mBERT) (Devlin et al., 2019) is trained on a mixture of multiple corpora written in different languages. A domain-specific BERT can also be trained on a mixture of a general and a domain-specific corpus.

The methods above use different vocabularies consisting of only general domain tokens (Lee et al., 2019; Devlin et al., 2019), only domain-specific tokens (Beltagy et al., 2019), or tokens extracted from the union of the two (Wang et al., 2020). We examine the impact of different combinations of data usages and vocabularies.

Our approach is partly inspired by data augmentation techniques that benefit from machine translation (Bahdanau et al., 2014; Vaswani et al., 2017), whereby labelled data were augmented for reading comprehension (Yu et al., 2018), fake news detection (Amjad et al., 2020) and other tasks. Unlike those approaches, our focus is on augmenting unlabelled data for pretraining, which has not been well explored, compared with augmenting labelled data for finetuning.

3 Methodology

We show details about our collection of translated domain-specific texts, the data usage and vocabulary for pretraining.

3.1 Collecting Texts for Pretraining

In the materials science domain in Japanese, it is difficult to obtain a large-scale corpus. On the other hand, Web of Science[2], a database of journals in English, provides a large-scale corpus of scientific papers, including many on materials science.

We extract the English abstracts of the articles tagged with "Materials Science" from journals with IDs of "DSSHPSH" and "ESCI". We used Amazon Translate[3] in January of 2020 to translate articles from English to Japanese. The use of a commercial automatic translation service can be justified because even non-experts in NLP can make use of such a service when they want to apply our methodology to other domains and languages. Finally, we obtained 2,501,178 translated abstracts with 21,115,139 sentences.

In addition, we used the dump of Japanese Wikipedia as of April 1st, 2020, containing

[1] https://material-analyzer.airc.aist.go.jp

[2] https://www.webofscience.com

[3] https://aws.amazon.com/translate/

1,197,647 articles in the general domain with 21,584,456 sentences.

3.2 Vocabulary and Data Usage

There are at least four possible ways of constructing a vocabulary, as shown in the upper part of Figure 2. `GeneralVocab` learns subword segmentation only from the general text, while `TransVocab` learns from the translated text, both using SentencePiece (Kudo and Richardson, 2018). Devlin et al. (2019) use the former, and SciBERT (Beltagy et al., 2019; Gupta et al., 2021) uses the latter. `MixVocab` learns from the mixed corpus of general and the translated text, which relates to mBERT (Devlin et al., 2019). `ConcatVocab`, which is similar to exBERT (Wang et al., 2020), learns two vocabularies, one learned from the general text and the other learned from the translated text, and then the union of the two is used as the final vocabulary.

We categorize approaches for pretraining BERT in terms of data usage and vocabulary construction. There are at least four possible combinations of methods in terms of data usage, as shown at the bottom part of Figure 2. `GeneralTrain` uses only the general texts (Devlin et al., 2019). `TransTrain` uses only the translated texts. `MixTrain` and `PipelineTrain` use both the general and translated texts. `MixTrain` pretrains BERT on a mixture of general and translated texts (Gupta et al., 2021). `PipelineTrain` first pretrains BERT on the general text and then continues to pretrain it on the translated texts.

Ten models with different combinations of vocabulary construction and data usage were trained and further compared on downstream tasks.

4 Experiments

We explain tasks, models, and datasets used for evaluating the proposed BERTs.

4.1 Downstream Tasks

The pretrained models were compared on the entity and relation extraction from texts in the domain of materials science in Japanese, as shown in Figure 3. For entity extraction, we extract four types of entities: 1) material names such as "cellulose", 2) properties of materials such as "transition temperature", 3) numerical values, and 4) units. The relation extraction assigns a label to each semantically related pair of entities. For example, since "transition temperature" is an attribute of "cellulose", we assign the label "AttributeOf" between the corresponding entities. We show the full list of entity labels and relation labels in Tables 1 and 2, respectively.

Entity labels
B-Material
B-Property
B-Value
B-Unit
I-Material
I-Property
I-Value
I-Unit
O

Table 1: Labels for the entity extraction task.

Relation labels
AttributeOf
Value
Unit
Abbreviation
Synonym
Conjunction
Other

Table 2: Labels for the relation extraction task.

In our experiments, we use two settings: entity and relation extractors that target either "glass transition temperature" or "elasticity". We focus on these two targets because these are particularly important in the material science domain. For the first setting, we are constrained to extract only entities and relations related to the glass transition temperature, which is particularly important for researchers in the target domain. For example, for the Task1 example in Figure 3, we should extract 170°C but not 240°C, because the latter relates to "pyrolysis temperature" not "glass transition temperature". For the second setting, we constrain the model to extract entities and relations only related to the elasticity, which is another important factor in the domain. These constraints make the tasks more challenging because the models need to correctly comprehend the context and find only the entities that relate to "glass transition temperature" or "elasticity".

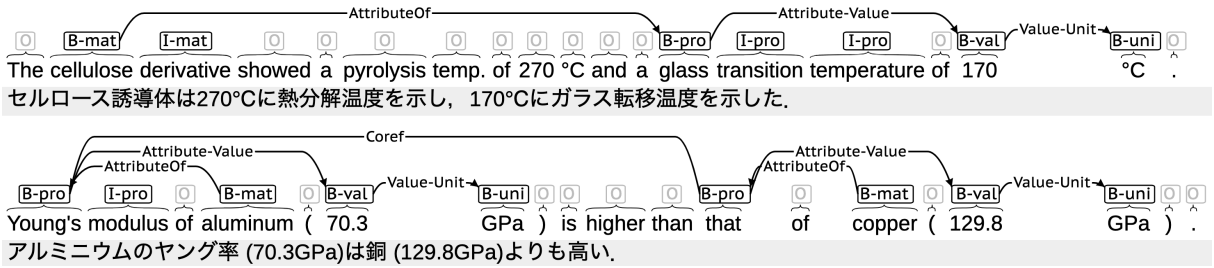

Figure 3: Examples of entity and relation extraction tasks on a text in the material science domain. We use two task settings for evaluating our proposed BERTs: the two sentences are from "glass transition temperature" and "elasticity" tasks, respectively. The shown annotation is a mock-up on the English translation; the actual input is in Japanese as shown in the line below.

4.2 Models for Downstream Tasks

We separately train the entity and relation extractors by the cross entropy losses. We use gold entities when finetuning relation extractors. The input sentence is tokenized by a Japanese morphological analyzer, MeCab (Kudo et al., 2004), and then segmented into subwords by a pretrained vocabulary described in Section 3.2. We add a special token [CLS] at the beginning of each sentence.

4.2.1 Entity Extractors

These subwords are encoded by BERT, and we obtain an embedding for each subword. We use BIO tagging scheme as shown in Figure 3. In addition to `O` (outside an entity), we use `B-material`, `I-material` and similarly for the other three entity types — 9 tags in total. For obtaining a score distribution over 9 tags, the embedding of the last subword in a token is passed to a classifier (multilayer perceptron (MLP) with one hidden layer) that assigns one of these tags.

4.2.2 Relation Extractors

The relation extractor predicts a relation label for each pair of entity spans. The representation of an entity span is obtained by the method of Trieu et al. (2020) that combines span representation (Sohrab et al., 2020) and the entity type representation. Then we concatenate the following four feature vectors: 1) representation of a head entity, 2) representation of a tail entity, 3) the element-wise product of the two entity representations (Luan et al., 2018; Lee et al., 2017), and 4) the embedding of the [CLS] token in the sentence. Given the concatenated feature vector, a classifier MLP followed by the softmax function returns probabilities of relation labels as a 7-dimensional vector.

4.3 Dataset for Finetuning and Evaluation

We use 27,053 sentences in 206 journal papers published in *Transactions of the Society of Polymer Science, Japan* for finetuning and evaluation. Experts manually annotated sentences with entities and relations. We use 60% of the dataset for training. The remaining data is equally divided into development and test data, where the former is used for selecting the model for evaluation. We conduct 5-fold cross-validation; the above data split is done five times. This dataset will be publicly available.

4.4 Parameters for Training Models for Downstream Tasks

When we induce subwords by SentencePiece (Kudo and Richardson, 2018), the sizes of `GeneralVocab`, `TransVocab`, and `MixVocab` are set to 32,104. For `ConcatVocab`, we use the union of `GeneralVocab` and `TransVocab`, resulting in the final vocabulary with 49,858 tokens. Each BERT was pretrained for 30 epochs by Adam (Kingma and Ba, 2015) with a learning rate of 10^{-4}. We finetune each extractor for 160 epochs by RAdam (Liu et al., 2020) with the learning rate 10^{-5}. We select the model with the highest macro-F1 score on the validation dataset. We report the averaged values of the five trials in 5-fold cross-validation.

4.5 Distributed Training of BERTs

We used distributed training for training BERTs to increase the speed of pretraining. We split the corpus for pretraining into four groups in terms of the length of the documents. A split contains the groups of texts with the lengths up to 128, 256, 384, or 512. We then calculated the cross entropy of each mini-batch in each split. We used one GPU for

each split, so we used four GPUs in total. Once we calculated cross entropy losses for every split, we averaged them and used them for backpropagation. We iteratively calculated losses and updated the parameters by using the averaged loss.

5 Results

Tables 3 and 4 show the respective scores for the two different settings: "glass transition temperature" and "elasticity". The span-based macro-Precision, Recall and F-score, which are commonly used, e.g., in Sohrab et al. (2020), are adopted as evaluation metrics. From top to bottom for both tables, we show the performances of the baseline (Model I) and nine proposed models (Model II to X). The proposed models are divided into three categories based on pretraining methods: 1) `TransTrain`, 2) `MixTrain`, and 3) `PipelineTrain`. For evaluating the relation extractors, we report performances on two settings; whether we use gold entities as input (Beltagy et al., 2019) or not in evaluation.

Do Translated Texts Improve the Performance?

All models trained on translated text (II to X) performed better than the model trained only on the general texts (I), the only exception being the precision of Model II on the relation extraction tasks for "glass transition tempreture". For Table 3, the baseline (Model I) trained only on the general text achieved an F-score of 90.24 for the entity extraction, while the BERT trained only from the translated text (Model II) achieved a higher F-score of 91.61, showing an improvement by 1.37 points. The F1 score on the relation task (gold) improved insignificantly (+0.04 points), and the score on the relation task (pred) showed minor improvements (+0.64), which can be attributed to improvement in entity extraction. However, the use of both types of texts does improve the performance, which reaches 78.76 and 72.36 at maximum. Thus, augmenting the general corpus by the translated corpus is more effective.

Similarly, in the task extracting "elasticity" shown in Table 4, the baseline entity extractor (Model I) achieved 92.64 in terms of F1 score, and all the models trained on the translated texts (Models II to X) achieved scores that are better than the baseline score.

How Do the Domain-specific Vocabularies Affect the Performance?

In `MixTrain`, `ConcatVocab` performs better than other vocabulary construction methods both for two settings: "glass transition temperature and elasticity". In entity extraction for two settings, we observed better F1 scores for Model IV with only the domain-specific vocabulary (91.66 for "glass transition temperature" and 94.12). Model V and VI, which construct vocabulary from both the general and translated texts, performed even better, i.e., 91.83 and 92.14 for the setting of "glass transition tempereture", respectively. Similar tendency can also be observed for the "elasticity" setting, i.e., 94.38 and 94.56 for Models V and VI, which are better than the F1 score 94.08 of Model III with only general vocabulary or the score 94.12 obtained by Model IV with only domain-specific vocabulary. We also observed a similar trend in relation extraction.

In contrast, for `PipelineTrain`, domain-specific vocabulary does not necessarily gain any performance. Even with the general-domain vocabulary (VII) alone, we obtained competitive or higher F1 scores (91.65, 78.58, 71.57, respectively for the three tasks) than most other models using the domain-specific vocabulary (VIII, IX, X). From the viewpoint of application, this is a favourable characteristic; we can expect high extraction performance by simply continuing pretraining a publicly available pretrained model on translated domain-specific text, instead of pretraining it on a huge general-domain text.

6 Conclusion

We showed that translated texts are beneficial for pretraining domain-specific BERTs in a low-resource language despite occasional translationese (Artetxe et al., 2020). Our approach can be applied to other languages and domains in which large-scale corpora are hard to obtain. In future work, our approach will be investigated on other pretrained models, e.g., GPT or BART, as well as other domains and languages. We leave investigations on the correlation between the translation qualities and downstream task performance as a future direction.

Acknowledgements

This paper is based on results obtained from a project JPNP20006, commissioned by the New En-

	Data for Pretrain	Vocab.	Entity (glass transition temperature)			Relation (gold)			Relation (pred)		
			P	R	F	P	R	F	P	R	F
Baseline (`GeneralTrain`)											
I	General data	`GeneralVocab`	89.42	91.12	90.24	78.16	77.51	77.53	69.93	71.06	70.19
Proposed Framework											
1) `TransTrain`											
II	Translated data	`TransVocab`	91.04	92.20	91.61	77.38	78.35	77.57	69.62	72.65	70.83
2) `MixTrain`											
III	Both data (mixed)	`GeneralVocab`	90.74	92.33	91.50	79.52	77.73	78.31	71.50	72.05	71.53
IV	Both data (mixed)	`TransVocab`	90.87	92.51	91.66	78.69	78.51	78.23	70.62	72.84	71.39
V	Both data (mixed)	`MixVocab`	90.94	92.78	91.83	79.15	79.03	**78.76**	70.51	73.67	71.81
VI	Both data (mixed)	`ConcatVocab`	91.03	93.30	**92.14**	**79.42**	78.65	78.74	71.70	73.52	**72.36**
3) `PipelineTrain`											
VII	Both data (pipeline)	`GeneralVocab`	90.85	92.51	91.65	79.15	78.49	78.58	70.78	72.75	71.57
VIII	Both data (pipeline)	`TransVocab`	91.46	92.39	91.91	79.04	78.90	78.69	71.55	73.24	72.17
IX	Both data (pipeline)	`MixVocab`	**91.17**	92.25	91.69	79.06	78.54	78.51	**71.84**	72.69	72.04
X	Both data (pipeline)	`ConcatVocab`	90.66	**92.45**	91.53	78.37	**79.64**	78.74	70.76	**73.95**	72.12

Table 3: Precision (P), Recall (R) and macro F1-score (F) on downstream tasks about glass transition temperature. The values better than the baselines are underlined. The proposed models, which use the translated texts, achieve better performances than the baseline.

	Data for Pretrain	Vocab.	Entity (elasticity)			Relation (gold)			Relation (pred)		
			P	R	F	P	R	F	P	R	F
Baseline (`GeneralTrain`)											
I	General data	`GeneralVocab`	92.64	93.15	92.87	77.99	78.51	78.36	71.04	71.86	71.00
Proposed Framework											
1) `TransTrain`											
II	Translated data	`TransVocab`	93.43	94.59	94.00	78.32	79.57	78.96	72.61	72.66	71.65
2) `MixTrain`											
III	Both data (mixed)	`GeneralVocab`	93.73	94.48	94.08	79.61	80.47	79.68	72.35	74.39	73.02
IV	Both data (mixed)	`TransVocab`	**94.45**	94.83	94.12	79.69	79.68	**80.71**	71.81	**75.13**	73.12
V	Both data (mixed)	`MixVocab`	93.78	95.01	94.38	79.02	79.03	79.98	71.91	74.70	72.91
VI	Both data (mixed)	`ConcatVocab`	94.11	95.04	**94.56**	79.58	79.64	80.55	73.05	74.86	73.42
3) `PipelineTrain`											
VII	Both data (pipeline)	`GeneralVocab`	93.62	94.69	94.13	79.60	80.43	79.60	72.25	74.54	73.04
VIII	Both data (pipeline)	`TransVocab`	94.21	94.22	94.18	**79.77**	**80.75**	79.86	**73.89**	74.49	**73.80**
IX	Both data (pipeline)	`MixVocab`	93.59	94.66	94.11	78.92	79.54	79.33	72.00	73.88	72.58
X	Both data (pipeline)	`ConcatVocab`	93.60	**95.25**	94.40	79.15	80.00	79.86	72.75	74.46	72.74

Table 4: Precision (P), Recall (R) and macro F1-score (F) on downstream tasks about elasticity. The values better than the baselines are underlined. The proposed models, which use the translated texts, achieve better performances than the baseline.

ergy and Industrial Technology Development Organization (NEDO). Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.

Ethics and Broader Impact

It is argued that existing machine translation systems are often biased in terms of some aspects such as gender. This may cause some biases in our translated dataset and our trained BERT model. However, our proposed BERT models are domain-specific and used only by experts not the general public. We believe the negative impact of such biases is limited if any.

Our proposed framework can be easily applied to various languages and domains. Our approach can have a significant impact on low-resource languages that have been difficult for researchers to train large language models due to the lack of large datasets. Our approach can also be applied to other architectures, such as decoder-only models, e.g., GPT, or encoder-decoder architectures, e.g., BART.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78.

Maaz Amjad, Grigori Sidorov, and Alisa Zhila. 2020. Data augmentation using machine translation for fake news detection in the Urdu language. In *Proceedings of the 12th Language Resources and Evaluation Con-*

ference (LREC2020), pages 2537–2542, Marseille, France.

Dogu Araci. 2019. FinBERT: Financial sentiment analysis with pre-trained language models. In *arXiv preprint (1908.10063, 2019)*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2020. Translation artifacts in cross-lingual transfer learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7674–7684, Online. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *Proceedings of the Third International Conference on Learning Representations (ICLR 2015)*, pages 1–15.

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-IJCNLP2019)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

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 (NAACL2019)*, pages 4171–4186, Minnesota, USA. Association for Computational Linguistics.

Tanishq Gupta, Mohd Zaki, N. M. Anoop Krishnan, and Mausam. 2021. Matscibert: A materials domain language model for text mining and information extraction.

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR2015)*.

Jens Kringelum, Sonny Kim Kjærulff, Søren Brunak, Ole Lund, Tudor I. Oprea, and Olivier Taboureau. 2016. Chemprot-3.0: a global chemical biology diseases mapping. *Database J. Biol. Databases Curation*, 2016.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to Japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP2004)*, pages 230–237, Barcelona, Spain. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *CoRR*, abs/1901.08746.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP2017)*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL2020)*, pages 7871–7880, Online. Association for Computational Linguistics.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, 2016.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2020. On the variance of the adaptive learning rate and beyond. In *Proceedings of the Eighth International Conference on Learning Representations (ICLR 2020)*, pages 1–13, Online.

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 (EMNLP2018)*, pages 3219–3232, Brussels, Belgium. Association for Computational Linguistics.

Kosuke Nishida, Naoki Yoshinaga, and Kyosuke Nishida. 2023. Self-adaptive named entity recognition by retrieving unstructured knowledge. In *The*

17th Conference of the European Chapter of the Association for Computational Linguistics (EACL2023).

Mohammad Golam Sohrab, Anh-Khoa Duong Nguyen, Makoto Miwa, and Hiroya Takamura. 2020. mgsohrab at WNUT 2020 shared task-1: Neural exhaustive approach for entity and relation recognition over wet lab protocols. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 290–298, Online. Association for Computational Linguistics.

Hai-Long Trieu, Thy Thy Tran, Khoa N A Duong, Anh Nguyen, Makoto Miwa, and Sophia Ananiadou. 2020. DeepEventMine: end-to-end neural nested event extraction from biomedical texts. *Bioinformatics*, 36(19):4910–4917.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS2017)*, pages 5998–6008.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656, Online. Association for Computational Linguistics.

Adams Wei Yu, David Dohan, Quoc Le, Thang Luong, Rui Zhao, and Kai Chen. 2018. Fast and accurate reading comprehension by combining self-attention and convolution. In *Proceedings of Sixth International Conference on Learning Representations (ICLR2018)*, Vancouver, Canada.

Rong Zhang, Revanth Gangi Reddy, Md Arafat Sultan, Vittorio Castelli, Anthony Ferritto, Radu Florian, Efsun Sarioglu Kayi, Salim Roukos, Avi Sil, and Todd Ward. 2020. Multi-stage pre-training for low-resource domain adaptation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP2020)*, pages 5461–5468, Online. Association for Computational Linguistics.