Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 2626–2633
Marseille, 20-25 June 2022

Re-train or Train from Scratch? Comparing Pre-training Strategies of BERT in the Medical Domain

Hicham El Boukkouri[1], Olivier Ferret[2], Thomas Lavergne[1], Pierre Zweigenbaum[1]
[1]Université Paris Saclay, CNRS, LISN, France
[2]Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
{elboukkouri,lavergne,pz}@lisn.fr, olivier.ferret@cea.fr

Abstract

BERT models used in specialized domains all seem to be the result of a simple strategy: initializing with the original BERT and then resuming pre-training on a specialized corpus. This method yields rather good performance (e.g. BioBERT (Lee et al., 2020), SciBERT (Beltagy et al., 2019), BlueBERT (Peng et al., 2019)). However, it seems reasonable to think that training directly on a specialized corpus, using a specialized vocabulary, could result in more tailored embeddings and thus help performance. To test this hypothesis, we train BERT models from scratch using many configurations involving general and medical corpora. Based on evaluations using four different tasks, we find that the initial corpus only has a weak influence on the performance of BERT models when these are further pre-trained on a medical corpus.

Keywords: word embeddings, contextualized embeddings, BERT, medical, biomedical, specialized domain, domain adaptation.

1. Introduction

Recent years have witnessed the widespread use of transfer learning techniques in Natural Language Processing (NLP) (Pan and Yang, 2010; Ruder, 2019). In fact, after being successful in speech processing (Wang and Zheng, 2015) and computer vision (He et al., 2017), transfer learning was applied in NLP as well with methods like ULMFiT (Howard and Ruder, 2018) showing that pre-trained language models can be successfully adapted through a sequential transfer learning process to bring performance gains to text classification. Shortly after, Radford et al. (2018) generalized these results to other tasks like natural language inference, text comprehension, etc.

Following the excitement around transfer learning in NLP, word embedding models moved from the *static embedding* paradigm (e.g. GloVe (Pennington et al., 2014), fastText (Bojanowski et al., 2017)) to the more dynamic *contextual embeddings* (ELMo (Peters et al., 2018), BERT (Devlin et al., 2019)) that are able to compute context-dependent word representations. Despite the performance gains that are brought by such models (see GLUE benchmark), using them in practice is significantly more costly than earlier static methods[1]. As a result, NLP practitioners tend to rely on pre-trained versions rather than training these models from scratch. Today, the most popular contextual embedding model seems to be BERT and the BERT-like models, for which there are pre-trained versions both for the *general domain* and for more *specialized domains* (e.g. BioBERT (Lee et al., 2020), BlueBERT (Peng et al., 2019), ClinicalBERT (Alsentzer et al., 2019a), and PubMedBERT (Gu et al., 2021) for the medical domain, SciBERT (Beltagy et al., 2019) for the scientific domain). Where general versions are pre-trained fully on general domain corpora, these specialized versions seem to all be the result of the same process: starting from a pre-trained general BERT and re-train it on a specialized corpus. This strategy brings undeniable improvements over using a general domain BERT (Alsentzer et al., 2019a; Si et al., 2019). However, the question remains: how do these re-trained models compare to models that are trained from scratch on specialized corpora without going through a general domain pre-training?

In this work, we will focus on the medical domain for which we will study the impact of three parameters on the downstream performance of BERT: the domain of its reference vocabulary (general vs. medical), the initial pre-training corpus (general vs. medical vs. both) and the second *specialization* corpus (none vs. medical). For a fairer comparison, we pre-train all the models ourselves using exactly the same hyper-parameters, then we evaluate these models on a variety of medical (clinical and biomedical) tasks: clinical concept detection (i2b2/VA 2010 (Uzuner et al., 2011)), clinical language inference (MEDNLI (Romanov and Shivade, 2018)), and biomedical relation extraction (ChemProt (Krallinger et al., 2017) and DDI (Herrero-Zazo et al., 2013)). This work focuses on the English language.

Our contributions are the following:

- we conduct a preliminary analysis of the impact of the domain of BERT's WordPiece vocabulary and show notable differences between a general and medical vocabulary when handling medical terms;
- we compare multiple BERT models with varying

[1]Pre-training BERT from scratch may require multiple GPUs for multiple weeks. The benchmarks performed by NVIDIA and available here give a good idea about the cost of the training and the fine-tuning of BERT models.

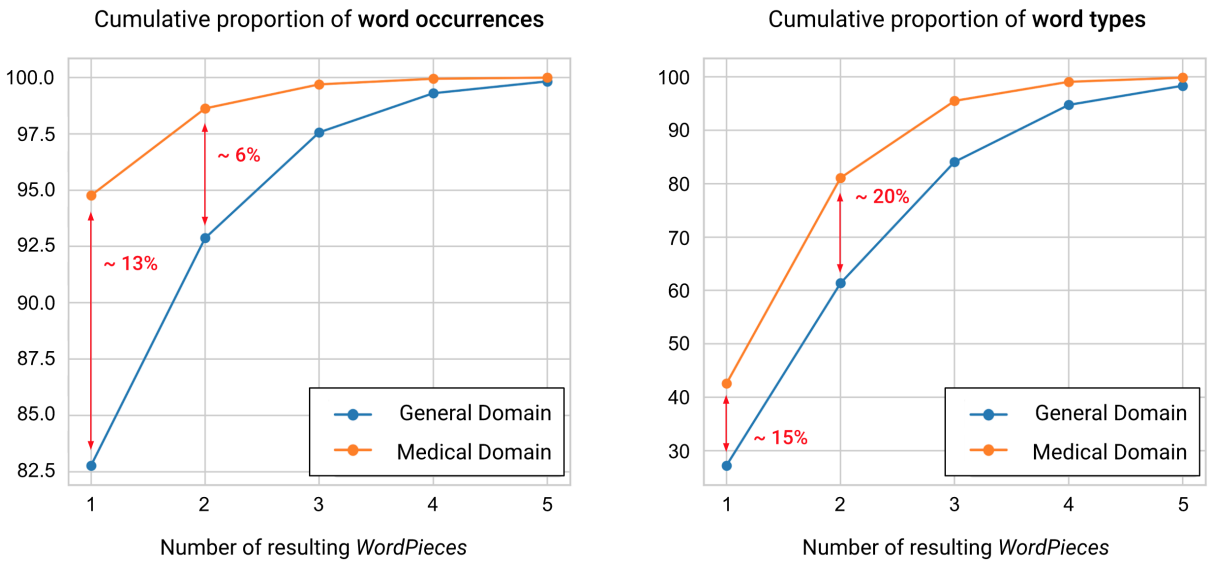

Figure 1: Comparison of the tokenization of a medical corpus by vocabularies from different domains.

degrees of specialization and observe that the standard strategy that consists in re-training a general model on specialized texts performs at the same level as a model that is trained from scratch on a specialized corpus using a specialized vocabulary;

- we share our code and pre-trained models for the benefit of the NLP community and to help the reproducibility of our experiments.

We will first introduce the principles of BERT (Section 2), in particular, those relative to the hypotheses we would like to test (Section 3), then we present our experiments (Section 4) and their results (Section 5) before a final conclusion (Section 6).

2. The Problem of BERT's Tokenization

BERT and BERT-like models such as RoBERTa (Liu et al., 2019), XLNet (Yang et al., 2019), or ALBERT (Lan et al., 2019), rely on a tokenization method that leverages subwords in order to keep the vocabulary small and handle potential out-of-vocabulary tokens by decomposing them into a sequence of known WordPieces (Wu et al., 2016). This allows striking a good balance between the efficiency of full words and the flexibility of characters, especially when constructing a model for a domain that is known in advance such as the default general domain. This WordPiece vocabulary is generally learned using a variant of BPE[2] (Gage, 1994), a compression algorithm that was adapted for tokenization purposes (Sennrich et al., 2016) such that the most frequent symbols are iteratively merged—forming character bi-grams, tri-grams, etc—and added to a subword vocabulary until a target size is reached. As a result, the output vocabulary is highly dependent on the corpus it was trained on—more specifically, its domain—which may entail some issues when the downstream applications cannot be expected to remain within the same original domain.

In practice, the subword vocabulary is learned using the exact same corpus that is used for pre-training, which ensures this vocabulary is a perfect match for the pre-training domain. However, a mismatch occurs when the pre-trained model is used on a task from a different domain, which, given the cost of training domain-specific versions of such large models, is likely in practice. Furthermore, when the resources for training such models for the target domain are available, the most popular approach seems to be re-training existing general-domain systems on specialized corpora (e.g. SciBERT[3] (Beltagy et al., 2019), ClinicalBERT (Alsentzer et al., 2019b)), probably in an attempt to leverage pre-existing knowledge within the model and speed up convergence. As a result, these models also re-use the original subword vocabulary, which could affect the pre-training procedure in a way that may be harmful, especially in specialized domains such as the medical domain where technical terms are very present.

3. Effect of BERT's Vocabulary on Tokenization

To study the effect of using general-domain vocabularies in specialized domains, we first look into the tokenization results. Here, we assume that we have access to the

[2]The exact implementation used for BERT seems to be internal to Google. However, most subsequent iterations like RoBERTa rely on the supposedly comparable BPE algorithm (see `https://github.com/google/sentencepiece#comparisons-with-other-implementations`).

[3]There are versions of SciBERT that use a custom scientific vocabulary as well.

original BERT model—which uses a general-domain WordPiece vocabulary—and investigate how this general vocabulary holds against specialized texts in the medical domain. In practice, after learning a medical WordPiece vocabulary, we randomly select a large sample ($\sim$ 1 million tokens) from the same medical corpus[4] and tokenize it using either BERT's original vocabulary (general domain) or our custom specialized vocabulary (medical domain). Figure 1 shows the cumulative proportion of word types (i.e. distinct tokens) and word occurrences (i.e. counting each token as many times as they occur) against the number of resulting subwords after tokenization.

If we look at the frequency of splitting an unknown token into multiple WordPieces, we realize that the medical vocabulary produces overall fewer WordPieces than the general version. Moreover, we see that $\approx 13\%$ of occurrences are never split as they are already part of the medical vocabulary but are decomposed into two or more WordPieces by the general vocabulary.

Reference	Vocabulary	Tokenization
paracetamol	medical	[paracetamol]
paracetamol	general	[para, ce, tam, ol]
choledocholithiasis	medical	[choledoch, olithiasis]
choledocholithiasis	general	[cho, led, och, oli, thi, asi, s]
borborygmi	medical	[bor, bor, yg, mi]
borborygmi	general	[bo, rb, ory, gm, i]

Table 1: Comparison of the tokenization of specific medical terms by vocabularies from different domains.

When looking closer at the quality of the produced WordPieces (see Table 1), we see that in addition to producing fewer subwords, the specialized vocabulary also seems to produce more meaningful units (e.g. *choledoch* and *olithiasis*).

4. Experiments

When training a specialized version of BERT, the usual approach consists in using a specialized corpus to resume the pre-training of the original BERT, which uses a general-domain vocabulary and was already pre-trained on a corpus of the general domain. In order to compare this approach with the other natural method consisting in pre-training a specialized version from scratch on a specialized corpus using a specialized vocabulary, we pre-train multiple BERT models with different vocabularies (general vs. medical), initial pre-training corpora (general vs. medical vs. mix of both) and specialization[5] corpora (none vs. medical).

In what follows, we will be using the BERT (base, uncased) architecture that consists of $L = 12$ Transformer layers, each using $H = 12$ attention heads and producing 768-dimensional representations. All our models are trained on lowercase English texts.

4.1. Different Configurations of BERT

Domain	Corpora	# of documents	# of words
General	Wikipedia (EN)	11.9 million	2.14 billion
	OpenWebText	3.15 million	1.28 billion
Medical	MIMIC-III	4.17 million	0.5 billion
	PubMed	4.65 million	0.5 billion

Table 2: Detail of the pre-training corpora.

We represent each configuration with a tuple corresponding to the values of the studied parameters: (V = vocabulary, C_1 = pre-training corpus, C_2 = specialization corpus).

($\mathrm{V} =$ **general,** $\mathrm{C}_1 =$ **general,** $\mathrm{C}_2 = \varnothing$) For a fairer comparison, we pre-train our own general domain model. Despite the redundancy with the open-source models from (Devlin et al., 2019), pre-training our own general-domain model guarantees that all the models we compare were trained in the same conditions. However, we use the same vocabulary as the original BERT: a vocabulary built from English Wikipedia and BooksCorpus (Zhu et al., 2015). During pre-training, we use a general corpus (see Table 2) made of English Wikipedia and a part of the OpenWebText[6] corpus (Gokaslan and Cohen, 2019). We sample enough data from OpenWebText to achieve a final corpus that is comparable in size to the one used in Devlin et al. (2019).

($\mathrm{V} =$ **general,** $\mathrm{C}_1 =$ **general,** $\mathrm{C}_2 =$ **medical**) Here we aim to reproduce the usual approach that re-trained a model from the general domain on a set of specialized texts. More precisely, while keeping the general vocabulary, we resume the pre-training of the previous model on a medical corpus made of clinical notes from MIMIC-III (Johnson et al., 2016) and biomedical paper abstracts from PubMed (Fiorini et al., 2018).

($\mathrm{V} =$ **medical,** $\mathrm{C}_1 =$ **medical,** $\mathrm{C}_2 = \varnothing$) Contrary to previous models, this version is directly trained on medical texts. Moreover, here we use a medical vocabulary that we build from the medical corpus (see Table 2) using the SentencePiece library that implements the BPE algorithm (Sennrich et al., 2015)[7].

[4]The sample is taken from the medical corpus described in Section 4.1 of this paper. More details about the vocabularies are also available in the same section.

[5]The specialization corpus is used to resume the pre-training of an already pre-trained model.

[6]Given that the BooksCorpus is not available anymore, we replaced it with OpenWebText, which aims to reproduce the WebText corpus that was used for training the GPT-2 model from Radford et al. (2019).

[7]It is important to note that the algorithm that was used to build the original BERT vocabulary is not available. Therefore, we need to use similar algorithms to build new vocabularies.

(V = **medical,** C_1 = **medical,** C_2 = **medical**) From the model that is pre-trained directly on a medical corpus, we pre-train a second time on the same corpus, which is equivalent to performing the pre-training on the medical corpus for twice as many epochs. This is done to achieve a model that is trained as long as (V = general, C_1 = general, C_2 = medical).

4.2. Evaluation Tasks

Pre-trained models are evaluated on five medical tasks after adding task-specific output layers as detailed in Devlin et al. (2019).

Medical Entity Recognition We evaluate on the i2b2/VA 2010 (Uzuner et al., 2011) clinical concept extraction task, which aims to extract three types of medical concepts: PROBLEM (e.g. *headache*), TREATMENT (e.g. *oxycodone*), and TEST (e.g. *MRI*). An example is given in Figure 2.

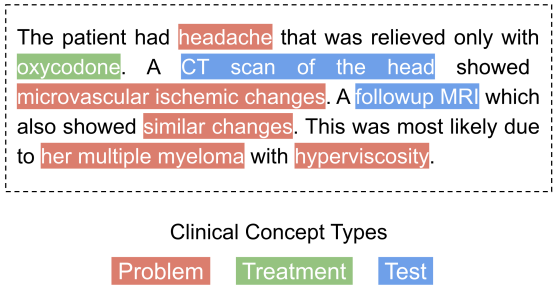

Figure 2: Example from i2b2/VA 2010.

Natural Language Inference We also evaluate on the clinical natural language inference task MEDNLI (Romanov and Shivade, 2018), which aims to classify sentence pairs into three categories: CONTRADICTION, ENTAILMENT, and NEUTRAL. Examples are given in Figure 3.

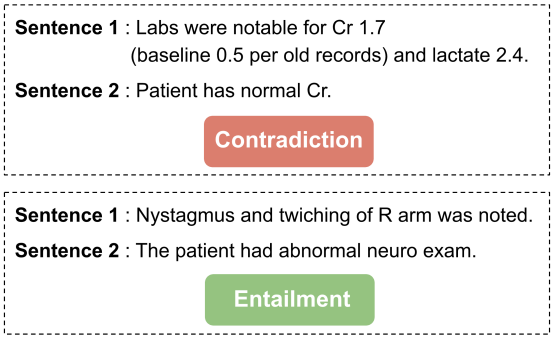

Figure 3: Examples from MEDNLI.

Relation Classification For more variety, we also evaluate on two biomedical relation classification tasks: ChemProt (Krallinger et al., 2017) from the BioCreative VI challenge and DDI (Herrero-Zazo et al., 2013) from SemEval 2013 - Task 9.2. The goal of ChemProt is to detect and classify chemical-protein interactions as ACTIVATOR (CPR:3), INHIBITOR (CPR:4), AGONIST (CPR:5), ANTAGONIST (CPR:6), or SUBSTRATE (CPR:9). The goal of DDI is to detect and classify drug-drug interactions into the following categories: ADVISE (DDI-advise), EFFECT (DDI-effect), MECHANISM (DDI-mechanism), and INTERACTION (DDI-int). Examples are given in Figure 4.

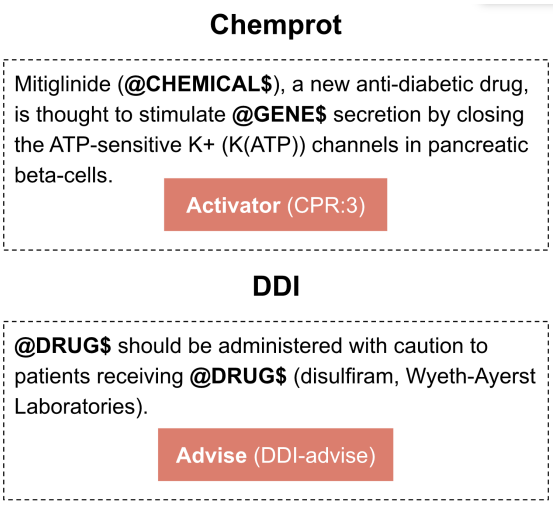

Figure 4: Examples from ChemProt and DDI.

The number of training, validation, and test examples of each task is reported in Table 3.

	i2b2	MEDNLI	ChemProt	DDI
Train	24.757	11.232	19.460	18.779
Val.	6.189	1.395	11.820	7.244
Test	45.404	1.422	16.943	5.761

Table 3: Number of examples of each evaluation task.

4.3. Implementation Details

In order to help reproduce our results, we share our pre-training and fine-tuning parameters. Moreover, we provide all of our pre-trained models as well as the code we used[8].

4.3.1. Pre-training Parameters

We train each model using 16 Tesla V100-SXM2-16GB GPUs and following the implementation and parameters in the NVIDIA codebase[9]. Each complete pre-training phase consists of two steps:

[8] https://github.com/helboukkouri/recital_2020

[9] More specifically, we adapt these scripts to our needs.

Model			Evaluation task			
V	**C_1**	**C_2**	**i2b2/VA 2010**	**MEDNLI**	**ChemProt**	**DDI**
general	general	∅	85.66 ± 0.18	77.31 ± 0.71	67.47 ± 0.99	75.81 ± 1.02
general	general	medical	<u>89.00</u> ± 0.17	**84.91** ± 0.46	<u>72.29</u> ± 0.58	<u>78.82</u> ± 1.11
medical	medical	∅	88.80 ± 0.10	83.54 ± 0.43	71.30 ± 0.51	79.40 ± 1.15
medical	medical	medical	**89.20** ± 0.20	84.32 ± 0.73	**72.97** ± 0.46	**80.11** ± 0.79
BERT (base) (Devlin et al., 2019)			86.42 ± 0.31	77.85 ± 0.63	69.22 ± 0.56	77.89 ± 0.92
BlueBERT (base) (Peng et al., 2019)			88.70 ± 0.21	<u>84.53</u> ± 0.76	68.35 ± 0.61	77.89 ± 0.65

Table 4: Evaluation results. The performance of i2b2/VA 2010 is computed in terms of strict F1 over the entities to detect, the performance of MEDNLI is given in terms of accuracy, and the performances of ChemProt and DDI are expressed in terms of micro-F1 measure. The best performance is shown in **bold** and the second best is <u>underlined</u>.

Step 1 3,519 updates with a total batch size[10] of 8,192 on sequences of size 128 with a learning rate of 6.10^{-3}.

Step 2 782 more updates with a batch size of 4,096 on sequences of size 512 with a lower learning rate of 4.10^{-3}.

During pre-training, we use the LAMB optimizer (You et al., 2020) as this has been shown to speed up convergence for large language models. We also use a linear schedule where the learning rate grows linearly during a certain number of training updates—i.e. 1000 steps for phase 1 and 100 steps for phase 2—before reaching its desired value and decreasing linearly to zero. Finally, we use a weight decay of 0.01 as is often the case with these models to further regularize training.

4.3.2. Fine-tuning Parameters

Each model is fine-tuned for a maximum of 15 epochs on the training data using a batch size of 32. After each iteration, we evaluate the model on the validation set and save the version with the best performance. After the 15 epochs are complete, we load the model from the best epoch and evaluate it on the test set.

5. Results and Discussion

To account for the randomness in the training and evaluation procedures, we run each fine-tuning with 10 different random seeds and compute a final model performance on each task as (mean± std). The results are shown in Table 4.

Pre-training Sanity Check Given the complexity of BERT's pre-training procedure, it is useful to compare our general-domain model to the original BERT: BERT (base). We observe that both models have similar performances with an advantage to the original BERT. However, this difference may be explained by either the different pre-training corpora (OpenWebText in our case instead of BooksCorpus) or the different pre-training parameters[11]. Therefore, given the similarity of the results, we can suppose that our pre-training procedure is correct.

One pre-training step: General BERT $<$ Medical BERT First, we compare the models that are only pre-trained once ($C_2 = \varnothing$). As expected, we can see that the model that is trained on a medical corpus using a medical vocabulary (V = medical, C_1 = medical, $C_2 = \varnothing$) gets systematically better results that the general domain version: an average of +3.14 F1, +6.23 Acc, +3.83 F1, and +3.59 F1 on i2b2, MEDNLI, ChemProt, and DDI respectively. Incidentally, it seems that using a medical vocabulary and training a model directly on a medical corpus (V = M, C_1 = M, $C_2 = \varnothing$) leads to better results than BlueBERT (except for MEDNLI), which re-trains the original BERT on a similar medical corpus[12]. This could hint at using a specialized vocabulary and training directly on in-domain data as being possibly better than the plain re-training of a general model on a specialized corpus.

Two pre-training steps When we look at the performance of the same models after being re-trained on a medical corpus, both models get similar results: on average +0.20 F1, -0.59 Acc, +0.68 F1 and +1.29 F1 on i2b2, MEDNLI, ChemProt and DDI respectively for the purely medical model (V = medical, C_1 = medical, C_2 = medical), which seems to improve slightly on (V = general, C_1 = general, C_2 = medical) when it comes to tasks in the biomedical domain (ChemProt and DDI).

Finally, we observe that our re-trained general model

[10] We use gradient accumulation for larger batch sizes.

[11] Due to server limitations, we trained our models for half the number of steps suggested in NVIDIA's implementation. This could mean that our models are overall under-trained.

[12] Their corpus, however, includes 8 times as many biomedical texts as we do. Interestingly, we outperform this model on ChemProt and DDI, both of which are from the biomedical domain.

gets better results than BlueBERT[13] on the biomedical tasks, especially on Chemprot (+3.07 F1), despite being pre-trained on similar corpora (MIMIC-III and PubMed).

Is Training from Scratch Really Better? Even though our medical model performs better than BlueBERT, a version that retrains the original BERT on a medical corpus, we see that training from scratch with a specialized vocabulary is not necessarily better, since re-training our general model on a specialized corpus (V = general, C_1 = general, C_2 = medical) eventually leads to better results. This reaffirms the importance of training models in similar conditions and warrants the inclusion of a medical model that is trained a second time on the same medical corpus. In fact, by re-training the purely medical model a second time (V = medical, C_1 = medical, C_2 = medical), we manage to improve over the previously mentioned re-trained version on most evaluation tasks, except for MedNLI. This goes to show that the overall training time of these models does matter, and to a point where a purely specialized model may perform worse than a model relying on general-domain representations if the former is not trained properly.

6. Conclusion

In a context where BERT-like representation models are more and more popular, we evaluate the usual way such models are used in a specialized domain: re-training the original BERT on a specialized corpus before adapting it to a task of interest. While focusing on the medical domain, we compared two methods where the model is first pre-trained on an initial corpus (general domain vs. medical domain) using an appropriate vocabulary (general / medical) before being re-trained on a medical corpus. We observe that despite the initial differences between the two models both of them perform at a similar level when re-trained on a specialized corpus. We conclude, after taking into account the resource cost of each method, that it is preferable to re-train the original BERT on a specialized corpus instead of training a model from scratch with a specialized vocabulary.

Another strategy adopted by several recent studies (Hong et al., 2021; Tai et al., 2020; Diao et al., 2021) for dealing with the discrepancy between the vocabulary of the general domain and the vocabulary of a more specialized domain like the medical domain is to propose methods for extending the vocabulary of the general domain model to include the vocabulary of the specialized domain or a part of it. The evaluation of these methods in the same framework as our work would be an interesting extension of the results presented in this article.

[13]This is the version of BlueBERT trained on PubMed and MIMIC-III.

7. Acknowledgements

This work has been funded by the French National Research Agency (ANR) under the ADDICTE project (ANR-17-CE23-0001). We gratefully acknowledge Junichi Tsujii and the Artificial Intelligence Research Center (AIRC)[14] for allowing us to use the ABCI cluster[15] to run our experiments.

8. Bibliographical References

Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jindi, D., Naumann, T., and McDermott, M. (2019a). Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jindi, D., Naumann, T., and McDermott, M. (2019b). Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Beltagy, I., Lo, K., and Cohan, A. (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 3606–3611.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (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, June. Association for Computational Linguistics.

Diao, S., Xu, R., Su, H., Jiang, Y., Song, Y., and Zhang, T. (2021). Taming Pre-trained Language Models with N-gram Representations for Low-Resource Domain Adaptation. 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 3336–3349, Online, August. Association for Computational Linguistics.

Gage, P. (1994). A new algorithm for data compression. *C Users Journal*, 12(2):23–38.

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H.

[14]https://www.airc.aist.go.jp/en/intro/

[15]https://abci.ai/

(2021). Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1), oct.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. B. (2017). Mask R-CNN. *CoRR*, abs/1703.06870.

Hong, J., Kim, T., Lim, H., and Choo, J. (2021). AVocaDo: Strategy for Adapting Vocabulary to Downstream Domain. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4692–4700, Online and Punta Cana, Dominican Republic, November. Association for Computational Linguistics.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). ALBERT: A lite BERT for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Pan, S. J. and Yang, Q. (2010). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, October.

Peng, Y., Yan, S., and Lu, Z. (2019). Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 2019 Workshop on Biomedical Natural Language Processing (BioNLP 2019)*, pages 58–65.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June. Association for Computational Linguistics.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training. *Technical report, OpenAI*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Ruder, S. (2019). *Neural transfer learning for natural language processing*. Ph.D. thesis, NUI Galway.

Sennrich, R., Haddow, B., and Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August. Association for Computational Linguistics.

Si, Y., Wang, J., Xu, H., and Roberts, K. (2019). Enhancing clinical concept extraction with contextual embeddings. *Journal of the American Medical Informatics Association*, 26(11):1297–1304.

Tai, W., Kung, H. T., Dong, X., Comiter, M., and Kuo, C.-F. (2020). exBERT: Extending Pre-trained Models with Domain-specific Vocabulary Under Constrained Training Resources. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1433–1439, Online, November. Association for Computational Linguistics.

Wang, D. and Zheng, T. F. (2015). Transfer learning for speech and language processing. In *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1225–1237. IEEE.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J. G., Salakhutdinov, R., and Le, Q. V. (2019). XLNet: Generalized autoregressive pretraining for language understanding. In Hanna M. Wallach, et al., editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

You, Y., Li, J., Reddi, S. J., Hseu, J., Kumar, S., Bhojanapalli, S., Song, X., Demmel, J., Keutzer, K., and Hsieh, C. (2020). Large batch optimization for deep learning: Training BERT in 76 minutes. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

9. Language Resource References

Fiorini, N., Leaman, R., Lipman, D. J., and Lu, Z. (2018). How user intelligence is improving PubMed. *Nature biotechnology*, 36(10):937–945.

Gokaslan, A. and Cohen, V. (2019). OpenWebText corpus. `http://Skylion007.github.io/OpenWebTextCorpus`.

Herrero-Zazo, M., Segura-Bedmar, I., Martínez, P., and Declerck, T. (2013). The DDI corpus: An annotated

corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 46(5):914–920.

Johnson, A. E., Pollard, T. J., Shen, L., Li-wei, H. L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L. A., and Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. *Scientific data*, 3:160035.

Krallinger, M., Rabal, O., Akhondi, S. A., et al. (2017). Overview of the BioCreative VI chemical-protein interaction track. In *Proceedings of the sixth BioCreative challenge evaluation workshop*, volume 1, pages 141–146.

Romanov, A. and Shivade, C. (2018). Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium, October-November. Association for Computational Linguistics.

Uzuner, Ö., South, B. R., Shen, S., and DuVall, S. L. (2011). 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.