Article

Patterns

Fine-tuning large neural language models for biomedical natural language processing

Highlights

- Systematic exploration of fine-tuning stability in biomedical NLP
- Domain-specific vocabulary and pretraining facilitate robust models for fine-tuning
- PubMedBERT-large and PubMedELECTRA models advance state-of-the-art in biomedical NLP

Authors

Robert Tinn, Hao Cheng, Yu Gu, ..., Tristan Naumann, Jianfeng Gao, Hoifung Poon

Correspondence

hoifung@microsoft.com

In brief

Large neural language models have transformed modern natural language processing (NLP). However, fine-tuning these models for specific tasks remains challenging as model size increases, especially with small labeled datasets, which are common in biomedical NLP. This systematic exploration of fine-tuning stability in biomedical NLP highlights the importance of domain-specific vocabulary and pretraining for creating robust models and establishes a new state of the art on a wide range of biomedical NLP applications.

Tinn et al., 2023, Patterns *4*, 100729
April 14, 2023 © 2023 The Authors.
https://doi.org/10.1016/j.patter.2023.100729

CellPress

Patterns

CellPress
OPEN ACCESS

Article

Fine-tuning large neural language models for biomedical natural language processing

Robert Tinn,[1,2] Hao Cheng,[1,2] Yu Gu,[1] Naoto Usuyama,[1] Xiaodong Liu,[1] Tristan Naumann,[1] Jianfeng Gao,[1] and Hoifung Poon[1,3,*]

[1]Microsoft Research, Redmond, WA, USA
[2]These authors contributed equally
[3]Lead contact
*Correspondence: hoifung@microsoft.com

https://doi.org/10.1016/j.patter.2023.100729

THE BIGGER PICTURE Large neural language models have transformed modern natural language processing (NLP) and have recently become a focus of public attention. However, fine-tuning these models for specific tasks of interest remains challenging as model size increases, especially with small labeled datasets, which are common in biomedical NLP.

This study conducts a systematic exploration of fine-tuning stability in biomedical NLP and identifies techniques that address instability and improve performance. The findings highlight the importance of domain-specific vocabulary and pretraining for creating robust models and establish a new state of the art on a wide range of biomedical NLP applications in the Biomedical Language Understanding and Reasoning Benchmark (BLURB).

1 2 3 4 5 **Proof-of-Concept:** Data science output has been formulated, implemented, and tested for one domain/problem

SUMMARY

Large neural language models have transformed modern natural language processing (NLP) applications. However, fine-tuning such models for specific tasks remains challenging as model size increases, especially with small labeled datasets, which are common in biomedical NLP. We conduct a systematic study on fine-tuning stability in biomedical NLP. We show that fine-tuning performance may be sensitive to pretraining settings and conduct an exploration of techniques for addressing fine-tuning instability. We show that these techniques can substantially improve fine-tuning performance for low-resource biomedical NLP applications. Specifically, freezing lower layers is helpful for standard BERT-BASE models, while layerwise decay is more effective for BERT-LARGE and ELECTRA models. For low-resource text similarity tasks, such as BIOSSES, reinitializing the top layers is the optimal strategy. Overall, domain-specific vocabulary and pretraining facilitate robust models for fine-tuning. Based on these findings, we establish a new state of the art on a wide range of biomedical NLP applications.

INTRODUCTION

Biomedical text is growing at an explosive rate. PubMed[1] adds thousands of scientific papers every day and more than a million every year. Simultaneously, digitization of patient records has created steadily growing resources of clinical text. For example, every year there are about 2 million new cancer patients in the United States alone, each with hundreds of clinical notes, such as pathology reports and progress notes.[2] Curating knowledge and longitudinal patient information from this text stands to accelerate clinical research and improve clinical care. Manual curation, however, does not scale to the rapid growth of biomedical text because manual curation often requires hours for each paper or patient and may require domain-specific expertise, such as clinical knowledge, that precludes common techniques in crowd-sourcing.

Natural language processing (NLP) has emerged as a promising direction to accelerate curation by automatically extracting candidate findings for human experts to validate.[3,4] However, standard supervised learning often requires a large amount of

Table 1. Summary of techniques for fine-tuning stabilization in recent studies and our investigation: conducting longer training, adopting bias correction in the ADAM algorithm, freezing pretrained parameters in the lower layers, adopting layerwise learning-rate decay, and reinitializing parameters in the top layers

	Domain	Longer training	ADAM debiasing	Layer freeze	Layerwise decay	Layer reinit
Grieβhaber et al.[7]	General (GLUE)			✓		
Mosbach et al.[8]	General (GLUE)	✓	✓			
Zhang et al.[9]	General (GLUE)	✓	✓		✓	✓
Ours	Biomedical (BLURB)	✓	✓	✓	✓	✓

training data. Consequently, task-agnostic self-supervised learning is rapidly gaining traction. By pretraining on unlabeled text, large neural language models facilitate transfer learning and have demonstrated spectacular success for a wide range of NLP applications.[5,6]

Fine-tuning these large neural models for specific tasks, however, may be unstable and prone to overfitting, as has been shown in the general domain.[7–9] For biomedicine, the challenge is further exacerbated by the scarcity of task-specific training data because annotation requires domain expertise and crowd-sourcing is harder to apply. For example, the Biomedical Semantic Similarity Estimation System (BIOSSES)[10] dataset, a semantic similarity task in the biomedical domain, contains only 100 annotated examples in total. By contrast, STS,[11] a similar dataset in the general domain, contains 8,628 examples.

In this paper, we conduct a systematic study on fine-tuning stability in biomedical NLP. We focus this effort on two popular models, Bidirectional Encoder Representations from Transformers (BERT) and Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA). We ground our study in BLURB, a recently proposed comprehensive benchmark for biomedical NLP comprising six tasks and 13 datasets.[12]

We first studied how pretraining settings impact fine-tuning performance. We show that for all applications, skipping next-sentence prediction (NSP) in pretraining has negligible effect, thus saving significant compute time, a finding consistent with general-domain observations by Liu et al.[6] and Aroca-Ouellette and Rudzicz.[13] However, modeling segment IDs during pretraining may have a large impact on certain semantic tasks, such as text similarity and question answering, especially when training data are scarce. Larger models (e.g., BERT-LARGE) significantly increase fine-tuning instability, and their use often hurts downstream performance. Interestingly, changing the pretraining objective from the masked language model (MLM) to ELECTRA has demonstrated improved performance in general-domain applications,[14] but it may exacerbate fine-tuning instability in low-resource biomedical applications.

We then conducted a comprehensive exploration of stabilization techniques to establish the best practice for biomedical fine-tuning. We show that conventional general-domain techniques, such as longer training and gradient debiasing, help but layerwise adaptation methods are key to restoring fine-tuning stability in biomedical applications. Interestingly, their efficacy may vary with pretraining settings and/or end tasks. For example, freezing lower layers is helpful for standard BERT-BASE models, whereas layerwise decay is more effective for BERT-LARGE and ELECTRA models. For low-resource text similarity tasks, such as BIOSSES, reinitializing the top layers is the optimal strategy. Overall, we find that domain-specific vocabulary and pretraining produce more robust language models. Based on these findings, we attain new state-of-the-art performance on a wide range of biomedical NLP tasks.

Finally, we showed that the best biomedical language models not only cover a much wider range of applications, but also substantially outperform off-the-shelf biomedical NLP tools on their currently available tasks. To facilitate biomedical research and applications, we released our state-of-the-art pretrained and task-specific fine-tuned models.

RESULTS

We conduct a systematic study on fine-tuning stability and mitigation methods in the presence of various pretraining settings and large models. Prior work studying fine-tuning stability and mitigation methods tends to focus on general domain—e.g., using BERT models pretrained on general-domain corpora and evaluating on GLUE[15] or SuperGLUE.[16] Table 1 summarizes representative recent work and common stabilization techniques.

We ground our study on the Biomedical Language Understanding & Reasoning Benchmark (BLURB).[12] BLURB is a comprehensive benchmark for biomedical NLP, spanning six tasks and 13 datasets, including applications with very small training datasets, such as text similarity and question answering. To facilitate a head-to-head comparison, we followed the train/dev/test setup from BLURB in all our experiments.

Instability with alternative pretraining settings

We first conducted an ablation study to evaluate the impact of pretraining settings on fine-tuning stability. Prior work on fine-tuning stability focuses almost exclusively on LARGE models; we showed that BASE models also suffer instability if we deviate from standard BERT pretraining settings.

Specifically, we experimented with skipping NSP during pretraining. Standard BERT pretraining inputs two text sequences (with two distinct segment IDs). We also experimented with inputting a single sequence at a time (with same segment ID). For a head-to-head comparison, we pretrained all language models from scratch on PubMed abstracts (i.e., using the same settings as PubMedBERT) and adopted the same fine-tuning settings as in Gu et al.[12]

Table 2 shows the results. In general, NSP has relatively little impact on end task performance. However, pretraining with single sequences leads to a substantial performance drop in the sentence similarity task (BIOSSES). Presumably performance degrades because this task requires comparison of two sentences and the training set is very small, therefore pretraining with

Table 2. Comparison of BLURB test performance with various pretraining settings: standard BERT pretraining; BERT pretraining without NSP (i.e., MLM only); BERT pretraining with MLM only and single-sequence (single-segment ID); ELECTRA

	BERT	BERT (no NSP)	BERT (no NSP, single seq)	ELECTRA
BC5-chem	93.33*	93.21	93.20	93.00
BC5-disease	85.62*	85.29	85.44	84.84
NCBI-disease	87.82	88.29	88.68*	87.17
BC2GM	84.52	84.41	84.63*	84.03
JNLPBA	79.10*	79.01	79.10*	78.57
EBM PICO	73.38	73.87*	73.64	73.57
ChemProt	77.24*	76.82	76.88	76.34
DDI	82.36	82.64*	82.45	80.58
GAD	83.96*	82.30	83.24	83.40
BIOSSES	93.46*	93.12	75.50	80.24
HoC	82.32	82.37*	81.91	81.28
PubMedQA	55.84	56.40	66.66*	64.96
BioASQ	87.56	83.57	85.64	88.93*
BLURB score	81.35*	81.00	79.04	79.61

*Highest performance for task (row).

two text segments helps. Surprisingly, pretraining with single sequences substantially improves test performance on PubMedQA, even though the task also inputs two text segments. Interestingly, even with the original pretraining setting (with NSP and two segments), simply using a single-segment ID in fine-tuning for PubMedQA would result in a similarly large gain in test performance (F1 63.92, not shown in the table). However, the standard setting (using separate segment ID) is still better for BIOSSES and BioASQ.

We also evaluated using the ELECTRA objective. Unlike in the general domain, ELECTRA does not show clear improvements over the MLM objective, when fine-tuned in an identical manner, in the biomedical domain. In fact, ELECTRA performs worse on most tasks and suffers a catastrophic performance drop in text similarity. We note that these tasks also happen to have relatively small training sets. This may appear contradictory with some recent work that demonstrates superior results using ELECTRA in biomedical NLP.[17] Later, we showed that it is indeed possible to attain higher performance with ELECTRA, but doing so requires various techniques to stabilize and improve fine-tuning. Compared with BERT with the MLM objective, ELECTRA is generally more difficult to fine-tune and demonstrates no significant advantage for biomedical NLP.

Stabilization by adjusting standard optimization

As previously mentioned, prior studies conclude that small optimization adjustments often suffice to restore fine-tuning stability in LARGE models. In biomedical NLP, however, we found that such adjustments are necessary to prevent catastrophic performance drops, but are not always sufficient for stabilizing fine-tuning, even with BASE models. Table 3 shows an ablation study on BIOSSES. In this case, forgoing either adjustment leads to a significant performance drop. But, as noted in the last subsection, even if both are used, fine-tuning remains unstable with alternative pretraining settings, which requires more advanced stabilization techniques.

Table 3. Ablation study on optimization adjustments in fine-tuning by comparing BIOSSES test performance under various pretraining settings

Pretraining setting	Improved optimization	Standard epochs	No bias correction
BERT	93.46*	92.64	91.75
BERT (no NSP)	93.12*	91.31	92.35
BERT (no NSP, single seq)	75.50*	0.65	70.50
ELECTRA	80.24	49.87	80.41*

Improved optimization used bias correction in ADAM and up to 100 epochs in fine-tuning (vs. up to five epochs in standard setting), all with BASE models.

*Highest performance for model (row).

Stabilization by layer-specific adaptation

Next, we studied various layer-specific adaptation methods in fine-tuning. Given that most models suffer from high instability on sentence similarity (BIOSSES) and question answering (BioASQ and PubMedQA), we focused on those tasks. For question answering, we reported the mean performance. Table 4 shows the results. All three methods are broadly beneficial, but their effects vary substantially with tasks and pretraining settings. Freezing lower layers is helpful for BERT models with the standard MLM objective, whereas layerwise decay is more effective for ELECTRA models. For sentence similarity, reinitializing the top layers is the optimal strategy. We focused our study on sentence similarity and question answering tasks, as other datasets in BLURB are relatively large and do not suffer from stability issues. We explored a combination of layer-specific adaptation methods but found little gain in preliminary experiments.

In addition, we consider the task of relation extraction and simulate low-resource settings by subsampling training instances (100/500/1000 from ChemProt and DDI that contain 18,035 and 25,296 training instances, respectively). Table 5 shows the results. Not surprisingly, test performance is lower with fewer training instances. However, the simulated results confirm that layer-specific adaptation generally increases fine-tuning stability and test performance (except in the extremely low-resource setting of 100 training instances).

Stabilization for larger models

It is well known that larger models can be finicky to fine-tune.[5] Again, we focused on sentence similarity (BIOSSES) and question answering (BioASQ and PubMedQA). Indeed, we observed a substantial drop in test performance on sentence similarity and question-answering tasks for most large models (see Table 6). Note that to avoid clutter, we only show the average scores for the question-answering tasks.

Surprisingly, PubMedBERT-LARGE is a notable exception because it does not suffer any catastrophic performance drop. In fact, it actually gains slightly on the question-answering tasks. This stands in stark contrast with other models such as BioBERT[18] and BlueBERT.[19] We hypothesize that its robustness

Table 4. Comparison of test performance (and standard deviation) on the BIOSSES and BioASQ tasks with major layer-specific adaptation methods, all with BASE models

Pretraining setting	Baseline	Laye freeze	Layerwise decay	Layer reinit
BIOSSES				
BERT	93.46 (0.96)	92.86 (0.88)	93.35 (0.78)	94.49* (0.88)
BERT (no NSP)	93.12 (1.04)	94.01* (0.99)	93.01 (1.07)	92.89 (0.91)
BERT (no NSP, single seq)	75.50 (3.00)	72.09 (2.86)	74.11 (3.51)	85.04* (2.69)
ELECTRA	80.24 (5.92)	83.06 (3.68)	83.55 (3.25)	88.74* (2.29)
BioASQ				
BERT	87.56 (2.43)	90.50* (1.51)	88.29 (2.76)	81.28 (3.72)
BERT (no NSP)	83.57 (3.60)	87.07* (3.15)	86.07 (2.29)	83.64 (3.70)
BERT (no NSP, single seq)	85.64 (2.48)	88.64* (1.76)	88.50 (1.63)	80.79 (2.51)
ELECTRA	88.93 (3.87)	90.00 (4.16)	90.64* (2.60)	88.14 (2.21)

*Highest performance for model (row).

stems from domain-specific vocabulary and pretraining. Interestingly, although PubMedELECTRA-LARGE is also pretrained in the same domain-specific fashion, it suffers a similar performance drop, which provides further evidence that the ELECTRA pretraining objective may exacerbate fine-tuning instability.

Optimization adjustments (longer training time and ADAM bias correction) have been used in all these experiments. Unlike in the general domain, they are not sufficient to restore stability. As in the case of BASE models, layer-specific adaptation methods can substantially reduce fine-tuning instability, in some cases enabling LARGE models to attain even higher performance than BASE (e.g., PubMedBERT-LARGE on QA and PubMedELECTRA-LARGE on SS). See Table 6.

Like Gu et al.,[12] we also observed that domain-specific vocabulary and pretraining are far superior, as PubMedBERT-LARGE substantially outperforms BioBERT-LARGE[18] and BlueBERT-LARGE[19]. Again, while ELECTRA models can perform reasonably well with advanced stabilization techniques, they are still finicky to fine-tune and are not superior over BERT models with the standard MLM pretraining objective. As with BASE models, reinitializing the top layers is still the optimal strategy for sentence similarity. However, for question answering, layerwise decay is superior for LARGE models.

Table 7 compares overall BLURB test performance for LARGE models with both improved optimization and layer-specific adaptation. They help stabilize fine-tuning, with no LARGE model suffering significant instability issues. With domain-specific vocabulary and pretraining, PubMedBERT-LARGE and PubMed Electra-LARGE benefit the most and attain significant gain over BASE.

Ablation study on layer removal

Rising concerns about computation cost of large pretrained models have spawned research in model pruning, such as removing top layers of a BERT model.[20] We thus conducted an ablation study on BLURB tasks to assess the impact of removing top layers from PubMedBERT. Table 8 shows the results. Indeed, pruning barely impacts fine-tuning efficacy for many tasks, such as named entity recognition (NER), evidence-based medical information extraction, sentence similarity, and document classification. Test performance does not substantially drop even when the top half of the layers were removed, suggesting that these tasks are relatively easy and do not require deep semantic modeling. By contrast, test performance in relation to extraction and question answering was substantially impacted by layer removal, dropping up to 3 to 4 absolute points for the former and up to 6 to 13 points for the latter. This suggests model pruning may make sense for simpler tasks, but not for semantically more challenging tasks. Further, this study suggests that the upper encoder layers are crucial for semantically challenging tasks, such as question answering. This illustrates why stabilization techniques, such as layerwise decay and layer freezing, are particularly beneficial for question-answering tasks—both techniques emphasize retraining these upper layers, which may have overfit to the pretraining objective.

New state-of-the-art in biomedical NLP

To further improve test performance for low-resource tasks, a common technique is to combine the training set and development set to train the final model—after hyperparameter search is done. We found that for most biomedical NLP tasks, this was not necessary, but it had a significant effect on BIOSSES. This is not surprising given that this dataset is the smallest.

By combining our findings on optimal fine-tuning strategy, we establish a new state-of-the-art in biomedical NLP. Table 9 shows the results. PubMedBERT with the MLM pretraining objective remains the best model, consistently outperforming ELECTRA in most tasks, although the latter does demonstrate some advantage in question-answering tasks, as can be seen in its superior performance with BASE models. With more extensive hyperparameter tuning, the gap between BASE and LARGE is smaller, compared with more standard fine-tuning (Table 6), which is not surprising. Overall, we were able to significantly improve the BLURB score by 1.6 absolute points, compared with the original PubMedBERT results in Gu et al.[12] (from 81.35 to 82.91).

Comparison with off-the-shelf tools

While there are many off-the-shelf tools for general-domain NLP tasks, there are few available for the biomedical domain. Two recent exceptions are scispaCy[21] and Stanza,[22] both with a limited scope focusing on NER. Table 10 compares sciSpaCy and Stanza performances with PubMedBERT on BLURB NER tasks. scispaCy comes with two versions, trained on JNLPBA and BC5CDR, respectively. Stanza comes with eight pretrained biomedical models, among which four overlap with or are related to BLURB NER tasks, namely JNLPBA, BC5CDR, NCBI-disease, and BC4CHEMD. We compare individually and to an oracle version of sciSpaCy and huggingface versions of Stanza that pick the optimal between the three for each evaluation dataset. As Stanza does not provide any gene/protein extraction

Table 5. Comparison of test performance (and standard deviation) on the ChemProt and DDI relation extraction tasks with major layer-specific adaptation methods, for reduced numbers of training instances

No. Training instances	Baseline	Layer freeze	Layerwise decay	Layer reinit
ChemProt				
100	22.45* (3.22)	20.39 (2.53)	20.50 (2.44*)	19.33 (3.18)
500	44.77 (3.71)	48.40 (2.85)	48.55* **(1.78*)**	43.79 (3.29)
1000	56.62 (1.93)	59.91* **(1.26*)**	59.67 (1.39)	55.31 (2.65)
DDI				
100	10.72 (2.93)	11.13* (3.73)	10.34 (2.50*)	9.83 (2.64)
500	34.36 (5.46)	39.78 (4.39)	40.15* **(3.34*)**	36.67 (5.50)
1000	58.71 (2.87)	61.40 (2.53)	61.54* **(1.46*)**	58.67 (3.54)

*Highest performance and lowest standard deviation.

model, its performances on BC2GM task are empty. While scispaCy and Stanza perform well, PubMedBERT fine-tuned models attain substantially higher scores. We note that many scispaCy errors stem from imperfect entity boundaries. We thus further compare the two using a lenient score that regards overlapping predictions as correct (Table 11). As expected, the gap shrinks but PubMedBERT models still demonstrate overwhelming improvement, raising the average score by more than 10 points compared with sciSpacy. In both settings, PubMedBERT models outperform Stanza across tasks.

DISCUSSION

Studies on pretraining and fine-tuning large neural language models originated in the general domain, such as newswire and the web. Recently, there has been increasing interest in biomedical pretraining[12,18,19,23] and applications.[24,25] In particular, Gu et al.[12] conducted an extensive evaluation of pretrained models on wide-ranging biomedical NLP tasks. However, they focus on domain-specific pretraining, whereas we study fine-tuning techniques and explore how they might interact with tasks and pretraining settings.

Prior studies on fine-tuning stability focused on the general domain and LARGE models, and often conclude that simple optimization adjustments, such as longer training time and ADAM debiasing, suffice for stabilization. By contrast, we show that in biomedical NLP, even BASE models may exhibit serious instability issues and simple optimization adjustments are necessary, but not sufficient, to restore stabilization. We systematically study how fine-tuning instability may be exacerbated with alternative pretraining settings such as using single sequences and the ELECTRA objective. We show that layer-specific adaptation methods help substantially in stabilization and identify the optimal strategy based on tasks and pretraining settings.

Limitations of the study

In this work, we identify several strategies that are effective for biomedical applications, but we have not found a single strategy or combination that works well across all models and tasks. Our exploration is also bounded by computational resources and time. While our study is relatively large and thorough, we have not exhaustively explored all possible settings or methods. Below we list some relevant directions that are beyond the scope of our study.

Multi-task learning can also mitigate the challenge presented by low-resource biomedical tasks.[26] This process generally requires applications with multiple related datasets,

Table 6. Comparison of test performance (and standard deviation) on the BIOSSES and BioASQ tasks with major layer-specific adaptation methods, all with LARGE models

	Baseline	Layer reeze	Layerwise decay	Layer reinit
BIOSSES				
BioBERT-LARGE	84.92 (10.2)	88.65 (6.60)	90.13 (1.70)	91.53* (4.09)
BlueBERT-LARGE	82.35 (2.21)	84.56 (1.95)	84.80 (2.58)	86.18* (1.21)
PubMedBERT-LARGE	91.06 (1.51)	91.19 (1.12)	90.87 (0.92)	92.73* (0.96)
PubMedELECTRA-LARGE	71.61 (4.91)	86.42 (3.33)	86.17 (1.21)	90.33* (1.04)
BioASQ				
BioBERT-LARGE	67.79 (6.59)	74.57 (3.18)	78.93* (3.39)	74.43 (9.76)
BlueBERT-LARGE	70.43 (3.91)	70.21 (2.99)	72.21* (2.68)	70.86 (2.15)
PubMedBERT-LARGE	92.36 (1.36)	93.14 (2.35)	93.36* (1.22)	91.21 (1.54)
PubMedELECTRA-LARGE	79.93 (5.04)	88.64 (4.09)	93.14* (1.71)	88.07 (3.21)

*Highest performance for model (row).

Table 7. Comparison of BLURB test performance using LARGE models (24 layers, 300M+ parameters), with optimal layer-specific stabilization methods

	BioBERT-LARGE	BlueBERT-LARGE	PubMedBERT-LARGE	PubMedELECTRA-LARGE
BC5-chem	93.05	90.24	93.23*	92.90
BC5-disease	84.97	82.93	85.77*	84.82
NCBI-disease	88.76*	86.44	88.25	87.93
BC2GM	84.21	80.86	84.72*	83.87
JNLPBA	78.83	77.59	79.44*	78.77
EBM PICO	73.81	72.43	73.61	73.95*
ChemProt	77.79	71.31	78.77*	76.80
DDI	81.53	78.99	82.39*	78.92
GAD	82.47	75.80	83.57	83.93*
BIOSSES	91.53	86.18	92.73*	90.33
HoC	81.57	81.35	82.57*	82.37
PubMedQA	55.16	55.24	67.38*	65.02
BioASQ	78.93	72.21	93.36*	93.14
BLURB score	80.09	77.11	82.86*	81.88
Δ BASE model	−0.59	+0.15	+0.58	+0.37

Note: the optimal strategies were all layer-reinit for BIOSSES and layerwise decay for QA.
*Highest performance for task (row).

such as NER. As discussed previously, NER tasks are relatively easy, and domain-specific pretrained models can already attain high performance without specific adaptation.

Other relevant methods include Mixout,[27] which has not been found to consistently improve performance, and fine-tuning on intermediate tasks,[28] which is not always applicable and incurs substantial computation. Instead, we focus on layer-specific adaptation techniques that are generalizable and easily implemented.

Finally, adversarial training can also help instability issues[29–31] and prompt-based learning has been shown to work well in low-resource settings.[32,33] They are worth exploring in future work.

EXPERIMENTAL PROCEDURES

Resource availability

Lead contact

Further information and requests for resources should be directed to the lead contact, Hoifung Poon (hoifung@microsoft.com).

Materials availability

This study did not generate any physical materials.

Data and code availability

Models are publicly available via Hugging Face (https://aka.ms/huggingface) and have been archived to Zenodo where not described by prior work. Specifically, the following models are made available.

- PubMedBERT[12]: https://aka.ms/pubmedbert
- PubMedBERT-*LARGE*[34]: https://aka.ms/pubmedbert-large

Table 8. Ablation study on the impact of model pruning by comparing test performance on BLURB tasks after removing top layers of PubMedBERT

	Layers removed				
	0	2	4	6	Performance drop
BC5-chem	93.33	93.22	92.96	92.40	−0.93
BC5-disease	85.62	85.35	85.17	84.29	−1.33
NCBI-disease	87.82	88.38	87.99	87.38	−0.44
BC2GM	84.52	84.32	83.46	82.05	−2.47
JNLPBA	79.10	78.94	78.88	78.07	−1.03
EBM PICO	73.38	73.38	73.33	73.35	−0.05
ChemProt	77.24	76.11	73.40	72.74	−4.50
DDI	82.36	82.16	79.71	79.30	−3.06
GAD	83.96	82.33	80.23	79.21	−4.75
BIOSSES	92.30	92.66	92.80	92.12	−0.18
HoC	82.32	82.46	82.43	82.01	−0.31
PubMedQA	55.84	51.22	49.76	50.08	−6.08
BioASQ	87.56	83.73	77.50	74.00	−13.56

Table 9. Comparison of BLURB test performance: standard fine-tuning vs. optimal fine-tuning with advanced stabilization methods PLUS extensive hyperparameter search

	PubMedBERT-BASE		PubMedBERT-LARGE		PubMedELECTRA-LARGE	
Fine-tuning	Standard	Optimal	Standard	Optimal	Standard	Optimal
BC5-chem	93.33	93.33	93.23*	93.23*	92.90	93.25
BC5-disease	85.62	85.62	85.77*	85.77*	84.82	85.23
NCBI-disease	87.82	88.21	88.25*	88.25*	87.93	88.19
BC2GM	84.52	84.55	84.72*	84.72*	83.87	84.47
JNLPBA	79.10	79.16	79.44*	79.44*	78.77	78.77
EBM PICO	73.38	73.45	73.61	73.61	73.95	74.02*
ChemProt	77.24	77.41	78.77*	78.77*	76.80	77.26
DDI	82.36	83.17	82.39	82.78*	78.92	80.37
GAD	83.96	84.01*	83.57	83.76	83.93	83.93
BIOSSES	92.30	94.49*	90.29	92.73	86.17	92.69
HoC	82.32	83.02*	82.57	82.70	82.37	82.37
PubMedQA	55.84	63.92	63.18	67.38*	60.18	65.02
BioASQ	87.56	82.75	92.36	93.36*	81.71	93.14
BLURB score	81.16	82.75	82.02	82.91*	79.83	82.44

Note: the addition of extensive hyperparameter tuning results in optimal results that may be higher than those reported without extensive hyperparameter tuning, as in Table 7.
*Highest performance for task (row).

- PubMedELECTRA[35]: https://aka.ms/pubmedelectra
- PubMedELECTRA-*LARGE*[36]: https://aka.ms/pubmedelectra-large

Additional details regarding the Biomedical Language Understanding & Reasoning Benchmark (BLURB), including data sizes and evaluation metrics used can be found in Gu et al.[12] and on the benchmark site: https://aka.ms/BLURB. BLURB comprises 13 datasets across six tasks that are made available by the dataset owners. Links to each of these datasets also can be found on the benchmark site. For reference, these are as follows:

- Named entity recognition (NER): BC5-chem,[37] BC5-disease,[37] NCBI-disease,[38] BC2GM,[39] and JNLPBA[40]
- Evidence-based medical information extraction (PICO): EBM PICO[41]
- Relation extraction: ChemProt,[42] DDI,[43] GAD[44]
- Sentence similarity: BIOSSES[10]
- Document classification: HoC[45]
- Question answering: PubMedQA,[46] BioASQ[47]

Any additional information is available from the lead contact upon request.

Methods

In this paper, we focus our study on BERT[5] and its variants, which have become a mainstay of neural language models in NLP applications. Here, we begin with a brief review of the evaluation metrics for each of the datasets comprising BLURB, and then review core technical aspects in neural language model pretraining and fine-tuning, providing a basis for the key research questions of our fine-tuning study.

Evaluation metrics

The six tasks in BLURB use evaluation metrics that are appropriate for each task. For reference, we identify each evaluation metric here, but refer to readers to Gu et al.[12] and the corresponding works describing each task for additional details.

Named-entity recognition (NER), including BC5-chem,[37] BC5-disease,[37] NCBI-disease,[38] BC2GM,[39] and JNLPBA,[40] use F1 score at the entity-level. *Evidence-based medical information extraction (PICO)*, including EBM PICO,[41] uses macro F1 score at the word-level. *Relation extraction*, including ChemProt,[42] DDI,[43] and GAD,[44] use micro F1 score. *Sentence similarity*, including BIOSSES,[10] use Pearson correlation between the gold standard scores and the scores produced by the model. *Document classification*, including HoC,[45] uses micro F1 score. *Question answering*, including PubMedQA[46] and BioASQ,[47] uses accuracy.

The BLURB score is the macro average of average tests results for each of the six tasks (NER, PICO, relation extraction, sentence similarity, document classification, and question answering).

Neural language models

The input to a neural language model consists of text spans, such as sentences, separated by special tokens [SEP]. To address the problem of out-of-vocabulary words, neural language models generate a vocabulary from subword

Table 10. Comparison of PubMedBERT fine-tuned models, scispaCy, Stanza on BLURB named-entity recognition tasks (standard entity-level test F1 score)

	scispaCy			Stanza					PubMedBERT	
	JNLPBA	bc5cdr	max	jnlpba	bc5cdr	ncbi-disease	bc4chemd	max	BASE	LARGE
BC5-chem	3.60	86.49	86.49	–	91.47	–	89.84	91.47	93.33*	93.23
BC5-disease	1.35	80.03	80.03	–	83.78	63.54	–	83.78	85.62	85.77*
NCBI-disease	1.77	57.18	57.18	–	68.00	86.89	–	86.89	87.82	88.25*
BC2GM	51.98	6.73	51.98	–	–	–	–	–	84.52	84.72*
JNLPBA	77.31	10.28	77.31	77.32	–	–	–	77.32	79.10	79.44*
Mean Score	27.20	48.14	70.60	77.32	81.08	75.22	89.84	84.86	86.08	86.28*

*Highest performance for task (row).

Table 11. Comparison of PubMedBERT fine-tuned models, scispaCy and Stanza on BLURB named-entity recognition tasks (relaxed entity-level test F1 score—overlap counted as correct)

	scispaCy			Stanza					PubMedBERT	
	jnlpba	bc5cdr	max	jnlpba	bc5cdr	ncbi-disease	bc4chemd	max	BASE	LARGE
BC5-chem	7.70	91.42	91.42	–	94.31	–	92.61	94.31	95.18	95.37*
BC5-disease	2.09	88.88	88.88	–	92.10	77.79	–	92.10	93.34	93.74*
NCBI-disease	12.94	74.64	74.64	–	81.83	93.37	–	93.37	95.22	95.24*
BC2GM	68.87	15.92	68.87	–	–	–	–	–	95.56	96.05*
JNLPBA	87.25	20.50	87.25	88.33	–	–	–	88.33	88.79	88.81*
Mean Score	35.77	58.27	82.21	88.33	89.42	85.58	92.61	92.02	93.62	93.84*

*Highest performance for task (row).

units, using Byte-Pair Encoding (BPE)[48] or variants such as WordPiece.[49] Essentially, the BPE algorithm tries to greedily identify a small set of subwords that can compactly form all words in a given corpus. It does this by initializing the vocabulary with all characters and delimiters found in the corpus. It then iteratively augments the vocabulary with a new subword that is most frequent in the corpus and can be formed by concatenating two existing subwords, until the vocabulary reaches the pre-specified size—e.g., 30,000 in standard BERT models or 50,000 in RoBERTa.[6] In this paper, we use the WordPiece algorithm, which is a BPE variant that augments the vocabulary using likelihood in an unigram language model rather than frequency in choosing which subwords to concatenate.

The text corpus and vocabulary may preserve the original case (cased) or convert all characters to lower case (uncased). Prior work, such as Gu et al.,[12] finds that case does not have significant impact on downstream tasks, so we simply use uncased in our work.

BERT[5] is a state-of-the-art neural language model based on a transformer architecture.[50] The transformer model introduces a multi-layer, multi-head self-attention mechanism, which has demonstrated superiority in leveraging GPU computation and modeling long-range text dependencies. Standard BERT pretraining inputs two text spans (e.g., sentences) and assigns a distinct segment ID to each. The input token sequence is first processed by a lexical encoder, which combines a token embedding, a position embedding, and a segment embedding by element-wise summation. This embedding layer is then passed to multiple layers of transformer modules. In each transformer layer, a contextual representation is generated for each token by summing a non-linear transformation of the representations of all tokens in the prior layer, weighted by attention computed using a given token's representation in the prior layer as query. The final layer outputs contextual representations for all tokens, which combines information from the whole text span.

BERT models come with two standard configurations: BASE uses 12 layers of transformer modules and 110 million parameters, LARGE uses 24 layers of transformer modules and 340 million parameters. Prior work applying BERT to biomedical NLP focuses on BASE models. By contrast, in this work, we conduct a systematic study on LARGE models as well, which reveals additional challenges for fine-tuning neural language models in biomedical NLP.

Pretraining objectives

Similar to other language models, the key idea of BERT pretraining is to predict held-out words in unlabeled text. Unlike most prior language models, BERT does not adhere to a generative model. Instead, Devlin et al.[5] introduces two self-supervised objectives: *Masked Language Model* and *Next Sentence Prediction*. MLM randomly replaces a subset of tokens by a special token (e.g., [MASK]), and asks the language model to predict them. The training objective is the cross-entropy loss between the original tokens and the predicted ones. Typically, 15% of the input tokens are chosen, among which a random 80% are replaced by [MASK], 10% are left unchanged, and 10% are randomly replaced by a token from the vocabulary. NSP is a binary classification task that determines for a given sentence pair whether one sentence follows the other in the original text. While MLM is undoubtedly essential for BERT pretraining, the utility of NSP has been called into question in prior work.[6] As such, we conduct ablation studies to probe how NSP and the use of segment IDs in pretraining might impact downstream fine-tuning performance.

Aside from standard BERT pretraining objectives, we also consider ELECTRA,[14] which has shown good performance in general-domain datasets such as GLUE[15] and SQuAD.[51,52] ELECTRA introduces an MLM-based generator to help pretrain a discriminator for use in end tasks. Specifically, given sample masked positions, first the generator predicts the most likely original tokens as in MLM, then the discriminator classifies, for all tokens, whether each is the original one. While ELECTRA shares some superficial similarity with generative adversarial network GAN,[53] the roles of generator and discriminator are very different. After pretraining, the generator in ELECTRA is discarded and the discriminator is used for downstream fine-tuning, whereas GAN typically discards the discriminator and uses the

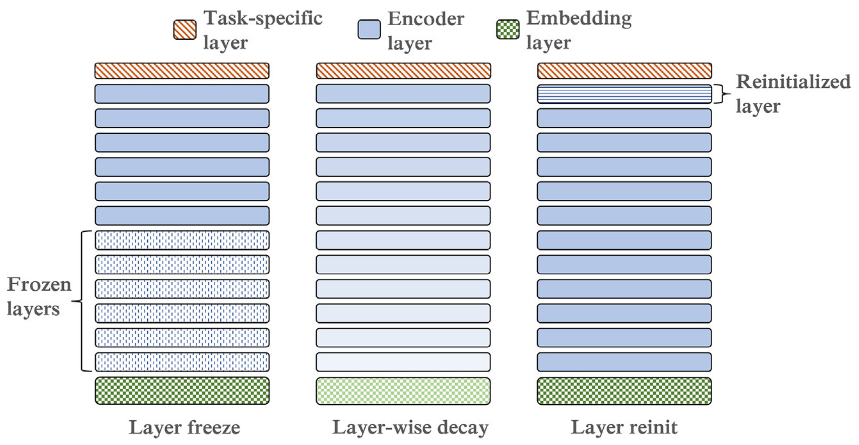

Figure 1. Illustration of major layer-specific adaptation methods for fine-tuning stabilization: freezing lower layers, adopting layer-wise decay of learning rate, and reinitializing the top layer

generator. The training objective is not adversarial, but a weighted combination of MLM for the generator and classification accuracy for the discriminator. By classifying on all tokens rather than just the masked ones, ELECTRA can potentially learn more from each example while adding little overhead as the majority of compute lies in transformer layers before classification. The generator, on the other hand, does incur additional compute. Also, if the generator becomes very accurate early on, there will be little learning signal for the discriminator. Therefore, ELECTRA typically uses lower capacity in the generator compared with the discriminator (e.g., one-third in BASE and one-fourth in LARGE for contextual representation dimension and attention head number).

Domain-specific pretraining

The study of neural language model pretraining originates in the general domain, including newswire and web. For example, the original BERT model was pretrained on Wikipedia and BooksCorpus.[5] RoBERTa,[6] another representative BERT model, was pretrained on a larger web corpus. Biomedical text is quite different from general-domain text and domain-specific pretraining has been shown to substantially improve performance in biomedical NLP applications.[12,18,19] In particular, Gu et al.[12] conducted a thorough analysis on domain-specific pretraining, which highlights the utility of using a domain-specific vocabulary and pretraining on domain-specific text from scratch. We build on their work and study how domain-specific pretraining might impact fine-tuning stability, especially for larger models and/or with alternative pretraining settings. To facilitate our investigation, we pretrained PubMedBERT-LARGE and PubMedELECTRA (BASE and LARGE) following the same setting of PubMedBERT (BASE) in Gu et al.[12] All BASE models use 12 layers, 768-dimension latent vectors, and 12 attention heads, with 110 million parameters. All LARGE models use 24 layers, 1,024-dimension latent vectors, and 16 attention heads, with 336 million parameters.

Fine-tuning stability

Prior work studying fine-tuning stability and mitigation methods tends to focus on the general domain—e.g., using BERT models pretrained on general-domain corpora and evaluating on GLUE[15] or SuperGLUE.[16] Table 1 summarizes representative recent work and common stabilization techniques. Small adjustments to the conventional optimization process may have surprisingly significant effect. For example, Mosbach et al.[8] and Zhang et al.[9] show that simply training for a longer time helps reduce fine-tuning instability with small training datasets. They also show that bias correction, which was proposed in the original ADAM algorithm[54] but was not used in fine-tuning from the original BERT paper,[5] can enhance fine-tuning stability by effectively reducing learning rates in the first few iterations.

Such minor adaptations are already highly effective for general-domain applications.[8] However, biomedical datasets are often much smaller than their general-domain counterparts. For example, as aforementioned for text similarity, the biomedical dataset BIOSSES[10] is much smaller than the general-domain dataset STS.[11] Similarly, the question-answering datasets in BLURB have only a few hundred instances, compared with more than 100,000 in SQuAD.[51]

Therefore, we systematically study advanced layer-specific adaptation techniques previously studied in the general domains: freezing pretrained parameters in the lower layers,[7] adopting layerwise learning-rate decay,[14] and reinitializing parameters in the top layers.[9] See Figure 1. Essentially, these techniques represent various ways to alleviate the vanishing gradient problem in training deep neural networks,[55] where optimization suffers from severe ill conditioning and requires adapting learning rates for individual layers. Interestingly, we find that their efficacy may interact with the pretraining setting and the end task.

Below is a list of all the methods we have explored with more details.

- longer training: use up to 100 epochs in fine-tuning (vs. up to five epochs in standard setting)
- ADAM debiasing: adopt bias correction in ADAM during fine-tuning
- layer freeze: fix pretrained parameters in the lower half layers of BERT models during fine-tuning (six layers for BASE models and 12 for LARGE models)
- layerwise decay: adopt layerwise learning-rate decay during fine-tuning (we follow ELECTRA implementation and use 0.8 and 0.9 as possible hyperparameters for learning-rate decay factors)
- layer reinit: randomly reinitialize parameters in the top layers before fine-tuning (up to three layers for BASE models and up to six for LARGE models)

ACKNOWLEDGMENTS

We thank the Patterns reviewers and editors for the detailed feedback and helpful suggestions.

AUTHOR CONTRIBUTIONS

H.P. and J.G. conceived the project and research design; R.T., H.C., Y.G., N.U., X.L., and T.N. conducted the research; T.N., J.G., and H.P. provided oversight and leadership for the research. All authors contributed to the preparation, review, and editing of the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

INCLUSION AND DIVERSITY

We support inclusive, diverse, and equitable conduct of research.

Received: October 10, 2022
Revised: December 12, 2022
Accepted: March 17, 2023
Published: April 14, 2023

REFERENCES

1. (2004). PubMed [Internet]. https://pubmed.ncbi.nlm.nih.gov/.
2. (2004). NCI cancer statistics [internet]. https://www.cancer.gov/about-cancer/understanding/statistics.
3. Wei, C.-H., Harris, B.R., Donghui Li and, T.Z.B., Huala, E., Kao, H.-Y., and Lu, Z. (2012). Accelerating Literature Curation with Text-Mining Tools: A Case Study of Using PubTator to Curate Genes in PubMed Abstracts. Database.
4. Wong, C., Rao, R., Yin, T., Statz, C., Mockus, S., Patterson, S., and Poon, H. (2021). Breaching the curation bottleneck with human-machine reading symbiosis. Preprint at MedRxiv. https://doi.org/10.1101/2021.07.14.21260440.
5. Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K.B. (2019). Pre-training of deep bidirectional transformers for language understanding. In Proc. Of NAACL-HLT, Volume 1 (Long and Short Papers), pp. 4171–4186.
6. 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. Preprint at arXiv. https://doi.org/10.48550/arXiv.1907.11692.
7. Grießhaber, D., Maucher, J., and Vu, N.T. (2020). Fine-tuning BERT for low-resource natural language understanding via active learning. In Proc. of 28th International Conference on Computational Linguistics (Barcelona, Spain (Online): International Committee on Computational Linguistics), pp. 1158–1171.
8. Mosbach, M., Andriushchenko, M., and Klakow, D. (2021). On the stability of fine-tuning {bert}: misconceptions, explanations, and strong baselines. In Proc. of ICLR https://openreview.net/forum?id=nzpLWnVAyah.
9. Zhang, T., Wu, F., Katiyar, A., Weinberger, K.Q., and Artzi, Y. (2021). Revisiting few-sample {bert} fine-tuning. In Proc. of ICLR https://openreview.net/forum?id=cO1IH43yUF.
10. Soğancıoğlu, G., Öztürk, H., and Özgür, A. (2017). Biosses: a semantic sentence similarity estimation system for the biomedical domain. Bioinformatics *33*, i49–i58.
11. Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. (2017). SemEval-2017 task 1: semantic textual similarity multilingual and crosslingual focused evaluation. In Proc. of SemEval-2017 (Vancouver, Canada:

Association for Computational Linguistics), pp. 1–14. https://doi.org/10.18653/v1/S17-2001.

12. Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. (2021). Domain-specific language model pretraining for biomedical natural language processing. ACM Transactions on Computing for Healthcare *1*, e3458754. https://doi.org/10.1145/3458754.
13. Aroca-Ouellette, S., and Rudzicz, F. (2020). On losses for modern language models. In *Proc. of EMNLP*. Online: Association for Computational Linguistics, pp. 4970–4981. https://doi.org/10.18653/v1/2020.emnlp-main.403.
14. Clark, K., Luong, M.-T., Le, Q.V., and Manning, C.D.E. (2020). Pre-training text encoders as discriminators rather than generators. In Proc. of ICLR https://openreview.net/forum?id=r1xMH1BtvB.
15. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S.R. (2019). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In Proc. of ICLR (OpenReview.net).
16. Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S.R.S. (2019). A stickier benchmark for general-purpose language understanding systems. In Proc. of NeurIPS, pp. 3261–3275.
17. Kanakarajan, K.r., Kundumani, B., and Sankarasubbu, M. (2021). BioELECTRA:pretrained biomedical text encoder using discriminators. In Proc. of BioNLP Workshop (Online: Association for Computational Linguistics), pp. 143–154. https://doi.org/10.18653/v1/2021.bionlp-1.16.
18. 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*, 1234–1240.
19. 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 Proc. of BioNLP Workshop and Shared Task (Florence, Italy: Association for Computational Linguistics), pp. 58–65. https://doi.org/10.18653/v1/W19-5006.
20. Sajjad, H., Dalvi, F., Durrani, N., and Nakov, P. (2020). Poor man's bert: smaller and faster transformer models. Preprint at arXiv. https://doi.org/10.48550/arXiv.2004.03844.
21. Neumann, M., King, D., Beltagy, I., and Ammar, W. (2019). ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing, pp. 319–327.
22. Zhang, Y., Zhang, Y., Qi, P., Manning, C.D., and Langlotz, C.P. (2021). Biomedical and clinical English model packages for the stanza python nlp library. J. Am. Med. Inf. Assoc. *28*, 1892–1899.
23. Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jin, D., Naumann, T., and McDermott, M. (2019). Publicly available clinical BERT embeddings. In Proc. of ClincalNLP Workshop (Minneapolis, Minnesota, USA: Association for Computational Linguistics), pp. 72–78. https://doi.org/10.18653/v1/W19-1909. https://www.aclweb.org/anthology/W19-1909.
24. Bressem, K.K., Adams, L.C., Gaudin, R.A., Tröltzsch, D., Hamm, B., Makowski, M.R., Schüle, C.Y., Vahldiek, J.L., and Niehues, S.M. (2020). Highly accurate classification of chest radiographic reports using a deep learning natural language model pre-trained on 3.8 million text reports. Bioinformatics *36*, 5255–5261.
25. Trieu, H.-L., Tran, T.T., Duong, K.N.A., Nguyen, A., Miwa, M., and Ananiadou, S. (2020). DeepEventMine: end-to-end neural nested event extraction from biomedical texts. Bioinformatics *36*, 4910–4917.
26. Zuo, M., and Zhang, Y. (2020). Dataset-aware multi-task learning approaches for biomedical named entity recognition. Bioinformatics *36*, 4331–4338.
27. Lee, C., Cho, K., and Kang, W.M. (2020). Effective regularization to fine-tune large-scale pretrained language models. In *Proc. of ICLR*. OpenReview.net https://openreview.net/forum?id=HkgaETNtDB.
28. Pruksachatkun, Y., Phang, J., Liu, H., Htut, P.M., Zhang, X., Pang, R.Y., Vania, C., Kann, K., and Bowman, S.R. (2020). Intermediate-task transfer learning with pretrained language models: when and why does it work? In *Proc. of ACL*. Online: Association for Computational Linguistics, pp. 5231–5247. https://doi.org/10.18653/v1/2020.acl-main.467. https://aclanthology.org/2020.acl-main.467.
29. Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Zhao, T. (2020). SMART: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. In *Proc. of ACL*. Online: Association for Computational Linguistics, pp. 2177–2190. https://doi.org/10.18653/v1/2020.acl-main.197. https://aclanthology.org/2020.acl-main.197.
30. Zhu, C., Cheng, Y., Gan, Z., Sun, S., Goldstein, T., and Liu, J.J.F. (2020). Enhanced adversarial training for natural language understanding. In Proc. of ICLR https://www.microsoft.com/en-us/research/publication/freelb-enhanced-adversarial-training-for-natural-language-understanding/.
31. Cheng, H., Liu, X., Pereira, L., Yu, Y., and Gao, J. (2021). Posterior differential regularization with f-divergence for improving model robustness. In *Proc. of NAACL-HLT*. Online: Association for Computational Linguistics, pp. 1078–1089. https://doi.org/10.18653/v1/2021.naacl-main.85. https://aclanthology.org/2021.naacl-main.85.
32. Schick, T., and Schütze, H. (2021). It's not just size that matters: small language models are also few-shot learners. In *Proc. of NAACL-HLT*. Online: Association for Computational Linguistics, pp. 2339–2352. https://doi.org/10.18653/v1/2021.naacl-main.185.
33. Gao, T., Fisch, A., and Chen, D. (2021). Making pre-trained language models better few-shot learners. In Proc. of ACL-IJCNLP (Volume 1: Long Papers) (Online: Association for Computational Linguistics), pp. 3816–3830. https://doi.org/10.18653/v1/2021.acl-long.295.
34. Tinn, R., Cheng, H., Gu, Y., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. (2023). microsoft/BiomedNLP-PubMedBERT-large-uncased- abstract: v0.1. Preprint at Zendo. https://doi.org/10.5281/zenodo.7627342.
35. Tinn, R., Cheng, H., Gu, Y., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. (2023). microsoft/BiomedNLP-PubMedELECTRA-base-uncased- abstract: v0.1. Preprint at Zendo. https://doi.org/10.5281/zenodo.7739298.
36. Tinn, R., Cheng, H., Gu, Y., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. (2023). microsoft/BiomedNLP-PubMedELECTRA-large-uncased- abstract: v0.1. Preprint at Zendo. https://doi.org/10.5281/zenodo.7739305.
37. Li, J., Sun, Y., Johnson, R.J., Sciaky, D., Wei, C.-H., Leaman, R., Davis, A.P., Mattingly, C.J., Wiegers, T.C., and Lu, Z. (2016). Biocreative v cdr task corpus: a resource for chemical disease relation extraction. Database. 2016.
38. Doğan, R.I., Leaman, R., and Lu, Z. (2014). Ncbi disease corpus: a resource for disease name recognition and concept normalization. J. Biomed. Inf. *47*, 1–10.
39. Smith, L., Tanabe, L.K., Ando, R.J.n., Kuo, C.-J., Chung, I.-F., Hsu, C.-N., Lin, Y.-S., Klinger, R., Friedrich, C.M., Ganchev, K., et al. (2008). Overview of biocreative ii gene mention recognition. Genome Biol. *9*, S2.
40. Kim, J.-D., Ohta, T., Tsuruoka, Y., Tateisi, Y., and Collier, N. (2004). Introduction to the bio-entity recognition task at JNLPBA. In Proc. of NLPBA/BioNLP Workshop (Geneva, Switzerland: COLING), pp. 73–78. https://www.aclweb.org/anthology/W04-1213.
41. Nye, B., Li, J.J., Patel, R., Yang, Y., Marshall, I.J., Nenkova, A., and Wallace, B.C. (2018). A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In Proc. of ACL (NIH Public Access), p. 197. 2018.
42. Krallinger, M., Rabal, O., Akhondi, S.A., Pérez, M.P., Santamaría, J., Rodríguez, G., Tsatsaronis, G., Intxaurrondo, A., López, J.A., and Nandal, U. (2017). Overview of the biocreative vi chemical-protein interaction track. In Proc. of BioCreative challenge evaluation workshop, *1*, pp. 141–146.
43. 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. J. Biomed. Inf. *46*, 914–920.

44. Bravo, À., Piñero, J., Queralt-Rosinach, N., Rautschka, M., and Furlong, L.I. (2015). Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. BMC Bioinf. *16*, 55.

45. Hanahan, D., and Weinberg, R.A. (2000). The hallmarks of cancer. Cell *100*, 57–70.

46. Jin, Q., Dhingra, B., Liu, Z., Cohen, W., and Lu, X. (2019). PubMedQA: A dataset for biomedical research question answering. In Proc. of EMNLP-IJCNLP (Hong Kong, China: Association for Computational Linguistics), pp. 2567–2577. https://doi.org/10.18653/v1/D19-1259.

47. Nentidis, A., Bougiatiotis, K., Krithara, A., and Paliouras, G. (2019). Results of the seventh edition of the bioasq challenge. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (Springer), pp. 553–568.

48. Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In Proc. Of ACL (Volume 1: Long Papers) (Berlin, Germany: Association for Computational Linguistics), pp. 1715–1725. https://doi.org/10.18653/v1/P16-1162.

49. Kudo, T., and Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Proc. of EMNLP: System Demonstrations (Brussels, Belgium: Association for Computational Linguistics), pp. 66–71. https://doi.org/10.18653/v1/D18-2012.

50. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In Proc. of NeurIPS, pp. 5998–6008.

51. Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proc. of EMNLP*. Austin (Texas: Association for Computational Linguistics), pp. 2383–2392. https://doi.org/10.18653/v1/D16-1264.

52. Rajpurkar, P., Jia, R., and Liang, P. (2018). Know what you don't know: unanswerable questions for SQuAD. In Proc. Of ACL (Volume 2: Short Papers) (Melbourne, Australia: Association for Computational Linguistics), pp. 784–789. https://doi.org/10.18653/v1/P18-2124.

53. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Proc. of NeurIPS, pp. 2672–2680.

54. Kingma, D.P., and Ba, J. (2015). Adam: A method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, Y. Bengio and Y. LeCun, eds.

55. Singh, B., De, S., Zhang, Y., Goldstein, T., and Taylor, G. (2015). Layer-specific adaptive learning rates for deep networks. In Proc. of IEEE ICMLA (IEEE), pp. 364–368.