Learning from Unlabelled Data for Clinical Semantic Textual Similarity

Yuxia Wang **Karin Verspoor** **Timothy Baldwin**
School of Computing and Information Systems
The University of Melbourne
Victoria, Australia
yuxiaw@student.unimelb.edu.au
karin.verspoor@unimelb.edu.au tb@ldwin.net

Abstract

Domain pretraining followed by task fine-tuning has become the standard paradigm for NLP tasks, but requires in-domain labelled data for task fine-tuning. To overcome this, we propose to utilise unlabelled domain data by assigning pseudo-labels from a general model. We evaluate the approach on two clinical STS datasets, and achieve $r = 0.80$ on N2C2-STS. Further investigation reveals that if the data distribution of unlabelled sentence pairs is closer to the test data, we can obtain better performance. By leveraging a large general-purpose STS dataset and small-scale in-domain training data, we obtain further improvements to $r = 0.90$, a new SOTA.

1 Introduction

Semantic textual similarity (STS) measures the degree of semantic equivalence between two text snippets, based on a graded numerical value, with applications including question answering (Yadav et al., 2020), duplicate detection (Poerner and Schütze, 2019), and entity linking (Zhou et al., 2020).

Modern pretrained language models have achieved impressive results for general STS (Devlin et al., 2019). However in low-resource domains without in-domain labelled data, results are generally lower (Wang et al., 2020b). In the clinical domain in particular, annotation requires medical experts (Wang et al., 2018; Romanov and Shivade, 2018), meaning that labelled datasets are generally small, hampering clinical STS.

We address the question of how to apply pretrained language models to such domain-specific tasks where there is little or no labelled data, focusing specifically on the task of clinical STS.

Employing a general STS model generally yields poor results over technical domains due to covariate shift. To bridge this gap, a standard approach is to pretrain the LM on in-domain text, such as ClinicalBERT (Alsentzer et al., 2019) using MIMIC-III (Johnson et al., 2016). However, existing research has tended to estimate effectiveness under the fine-tuning setting, rather than via inference tasks (Peng et al., 2019; Wang et al., 2020b).

In this paper, we first evaluate domain pretraining approaches for clinical STS, with no labelled data. Based on the assumption that general STS models trained on large-scale STS datasets will perform reasonably well on clinical sentence pairs (Section 4), we then experiment with learning from the pseudo-labelled data (Section 5).

Experimental results show both domain pretraining and pseudo-labelled data fine-tuning improve clinical STS, and the combination of the two achieves the best performance of $r = 0.80$ on N2C2-STS (Section 6.3). Further analysis shows that the score distribution and volume of pseudo-labelled pairs influence the performance of fine-tuning. We also find that training for more iterations leads to minor improvements.

The paper makes three major contributions: (1) we propose a simple pseudo-training method, and show it to perform well on clinical STS; (2) we evaluate several existing approaches to clinical STS in a zero-shot setting, and benchmark against our method; and (3) we achieve state-of-the-art results of $r = 0.90$ for N2C2-STS.

2 Related Work

The general approach to domain-specific task modelling is: (1) pretrain a language model (LM) on a large volume of open-domain text (Devlin et al., 2019; Liu et al., 2019); and (2) fine-tune on domain-specific text and task-specific labelled data (Gururangan et al., 2020; Peng et al., 2019). For this approach, however, domain-specific labelled data is required, an assumption that we seek to relax.

Proceedings of the 3rd Clinical Natural Language Processing Workshop, pages 227–233
November 19, 2020.

For STS, in the absence of labelled data, the simplest approach is to calculate the cosine similarity between the CLS-vectors of two sentences or averaged last-layer embeddings, but this tends to perform poorly, even worse than averaged GloVe (Pennington et al., 2014) embeddings. SBERT (Reimers and Gurevych, 2019) proposed to use a Siamese structure based on BERT to learn sentence representations, where they fine-tuned the model over general NLI data, and continued to fine-tune on general STS data (STS-B) (Cer et al., 2017). In this work, we experiment with this approach specifically in the clinical context.

3 Datasets and Tasks

We select two available clinical STS benchmark datasets for evaluation: MedSTS (Wang et al., 2018) and N2C2-STS (Wang et al., 2020a). The latter annotated 412 instances as new test bed, and updated train partition by labelling extra 574 instances and merging the former train and test cases (see Table 1). Our aim is to predict a score, given a sentence pair $(S1, S2)$, closing to the gold label — a numerical value ranging from 0 to 5, where 0 refers to completely dissimilar semantics while 5 is completely equivalent in the meaning.

For example,

S1: Discussed goals, risks, alternatives, advanced directives, and the necessity of other members of the surgical team participating in the procedure with the **patient**.

S2: Discussed risks, goals, alternatives, advance directives, and the necessity of other members of the healthcare team participating in the procedure with the **patient and his mother**.

Label: 4, as the two sentences are mostly equivalent and differ only in unimportant details (in bold).

Pearson's correlation (r) and Spearman's correlation (ρ) between the predicted and gold standard scores are used as evaluation metrics.

4 Observations

In modern NLP, large amounts of high-quality training data are a key element in building successful systems (Aharoni and Goldberg, 2020). This is also the case with STS, where additional training data has been shown to improve accuracy (Wang et al., 2020b). However, domain shifts inevitably lead to performance drops (Gururangan et al., 2020). Therefore, we ask: **RQ1** Can large-scale general-domain labelled STS data be transferred to train clinical STS models? **RQ2** How does low-quality training data impact clinical STS performance, vs. high-quality labelled data or no labelled data?

Dataset	Len	Train Size	Test Size
MedSTS	25.4	750	318
N2C2-STS	19.3	1642	412

Table 1: Clinical STS datasets. Train and Test Size = number of text pairs. Len = mean sentence length in tokens.

Eval set / Model	Data	r	ρ
STS-B dev:			
CLS-BERT	STS-B train	.900	.896
CLS-BERT	STS-G	.928	.927
N2C2-STS test:			
HConvBERT	STS-B train + N2C2-STS train	.894	.830
HConvBERT	STS-G + N2C2-STS train	.902	.836

Table 2: Pearson's r and Spearman's ρ evaluation on STS-B dev (upper half) and N2C2-STS test (bottom half), based on fine-tuning over STS-B train (5,749) and STS-G (28,518), for CLS-BERT and HConvBERT.

Effect of Larger General STS Corpus. We source general-domain labelled data from: (1) SemEval-STS shared tasks 2012–2017 (Agirre et al., 2012, 2013, 2014, 2015, 2016; Cer et al., 2017); and SICK-R (Marelli et al., 2014). This results in a total of 28,518 labelled sentence pairs, which we refer to as "STS-G".

We adapt a BERT encoder connected to a linear regression layer to fine-tune a general-domain STS model using STS-G, where the CLS-vector is used to represent the sentence pair (CLS-BERT). We compare this with a model trained only on STS-B. We evaluate both models on STS-B dev (same setup as Section 6.1).

For clinical STS, we employ a hierarchical convolution (HConv) model based on BERT (updating parameters of the last four layers), where the model is first fine-tuned with STS-B, then N2C2-STS is augmented by back-translation (Wang et al., 2020b). The model architecture and hyperparameter settings are the same as the original paper, such that we merely replace STS-B with STS-G, and observe that more training data improves clincial STS.

As shown in Table 2, the extra training data in STS-G results in an increase in r of up to .028, in the case of HConvBERT (Wang et al., 2020b), resulting in a new SOTA of $r = .902$.

Discussion. Though general-domain data lacks clinical information, the model clearly benefits from the extra out-of-domain training data (answering RQ1). This inspires us to rethink the clinical STS task as a combination of domain-specific text understanding and domain-invariant task learning, leading to the question: can the two aspects be learned separately? That is, can task learning take place via large volumes of general-domain labelled data, and domain-specific characteristics be learned from silver-standard labelled domain data, such as low-quality clinical sentence pairs labelled by a general STS model?

5 Method

Next, we investigate the use of pseudo-labelled clinical data based on the general STS model.

5.1 Pseudo-Labelled Sentence Pairs

Gururangan et al. (2020) illustrate that if the data distribution of the text used for pretraining is more similar to the task data, the performance will be better. Based on this, we propose a distribution-centric strategy for generating and selecting sentence pairs.

Generation. Two data sources — MIMIC-III clinical notes and N2C2-STS training data (ignoring labels) — are used to generate unlabelled sentence pairs. We sample 10,000 discharge summaries from MIMIC-III, which we segment into 27 parts based on section subtitles. Of these, we select five sections we consider to be most related to the N2C2-STS task: *diagnosis*, *medications*, *history of present illness*, *follow-up instructions* and *physical exam*. After sentence segmentation using SpaCy (Honnibal and Montani, 2017), we randomly sample sentence pairs from each section partition.

Labelling and Sampling. We take the CLS-BERT model trained on STS-G, and generate a score for all sentence pairs. To balance the data, we group into 5 equal-width bands based on score: $[0.0, 1.0]$, $(1.0, 2.0]$, $(2.0, 3.0]$, $(3.0, 4.0]$ and $(4.0, 5.0]$. We use all pairs whose assigned score is above 3.0, and sample N pairs from the other three intervals.

5.2 Iterative Training

We fine-tune the model over the resulting pseudo-labelled data, repeat the process of labelling and sampling, and further fine-tune the model on the second set of pseudo-labelled data.

Score	$[0.0, 1.0]$	$(1.0, 2.0]$	$(2.0, 3.0]$	$(3.0, 4.0]$	$(4.0, 5.0]$
500k	229622	211405	54517	4015	441
STS-PL	4015	4015	4015	4015	441
100k	45602	42479	10996	839	84
STS-PS	1500	1500	1500	839	84
500k	399975	81282	16841	1468	434
STS-DP	1468	1468	1468	1468	434

Table 3: Score distribution of 500k sentence pairs used for STS-PL and 100k pairs used for STS-PS. STS-DP is based on a domain-pretrained model (see Section 6.3).

6 Experiments

We first evaluate existing approaches for clinical STS in the zero-shot setting, and compare with our method. Then we analyse the impact of the volume of sampled instances and data distribution on the fine-tuning quality. We experiment with the number of iterations in Section 6.5.

6.1 Experimental Setup

We evaluate over MedSTS and N2C2-STS. As gathering naturally occurring pairs of sentences with different degrees of semantic similarity is very challenging (Wang et al., 2018), only 84 instances in $(4.0, 5.0]$ are sampled from a group of 100k unlabelled sentence pairs (see Table 3). To increase the number of instances with high similarity, another group of 500k unlabelled sentence pairs is generated from discharge summaries. Limiting to cases above 3.0, (1) "STS-PS" (Pseudo-labelled Small) = 5,423 pairs, is sampled from 100k based on $N = 1500$; and (2) "STS-PL" (Pseudo-labelled Large) = 16,501 pairs, is sampled from 500k based on $N = 4015$.

Unless otherwise indicated, pseudo labelling is based on CLS-BERT$_{\text{base}}$-STS-G (see Section 4). All models are trained with a batch size of 16, learning rate of 2e-5, and 3 epochs with linear scheduler setting warmup proportion of 0.1 of fine-tuning. For all CLS-BERT models, we update all 12 layers, and for HConvBERT we update the last 4 layers.

6.2 Results

We perform experiments over three models (SBERT, CLS-BERT, and HConvBERT), two pretraining configurations (general and clinical), and four training datasets (general gold-labelled STS-B and STS-G, clinical pseudo-labelled STS-PL and STS-PS).

Results are presented in Tables 4 and 5 for N2C2-STS and MedSTS, resp. Here, the subscripts for

Model	Data	r	ρ
Standard labels:			
IBM-N2C2	N2C2-STS train	.901	—
HConvBERT$_{\text{base}}$	N2C2-STS train + STS-G	.902	.836
Zero-shot setting:			
SBERT$_{\text{base}}$	NLI	.378	.392
SBERT$_{\text{base}}$	NLI + STS-B	.603	.604
CLS-BERT$_{\text{base}}$	STS-B	.682	.689
CLS-BERT$_{\text{clinical}}$	STS-B	.694	.697
CLS-BERT$_{\text{base}}$	STS-B + STS-PL	.780	.755
CLS-BERT$_{\text{base}}$	STS-B + STS-PS	.777	.749
CLS-BERT$_{\text{base}}$	STS-G	.721	.720
CLS-BERT$_{\text{clinical}}$	STS-G	**.788**	**.768**
CLS-BERT$_{\text{base}}$	STS-G + STS-PL	*.781*	*.767*
CLS-BERT$_{\text{base}}$	STS-G + STS-PS	.763	.750
HConvBERT$_{\text{base}}$	STS-B	.728	.719
HConvBERT$_{\text{clinical}}$	STS-B	.522	.526
HConvBERT$_{\text{base}}$	STS-B + STS-PL	.760	.740
HConvBERT$_{\text{base}}$	STS-B + STS-PS	.758	.733
HConvBERT$_{\text{base}}$	STS-G	.731	.716
HConvBERT$_{\text{clinical}}$	STS-G	.653	.653
HConvBERT$_{\text{base}}$	STS-G + STS-PL	.768	.749
HConvBERT$_{\text{base}}$	STS-G + STS-PS	.752	.734

Table 4: Results on N2C2-STS, based on fine-tuning on STS-B, STS-G, STS-PS and STS-PL.

Model	Data	r	ρ
Standard labels:			
CLS-BERT$_{\text{P+M}}$	MedSTS train	.848	—
Zero-shot setting:			
Baseline	—	.618	—
SBERT$_{\text{base}}$	NLI	.608	.594
SBERT$_{\text{base}}$	NLI + STS-B	.731	.679
CLS-BERT$_{\text{base}}$	STS-B	.786	.716
CLS-BERT$_{\text{clinical}}$	STS-B	.788	.693
CLS-BERT$_{\text{base}}$	STS-B + STS-PL	*.808*	.726
CLS-BERT$_{\text{base}}$	STS-B + STS-PS	**.815**	**.739**
CLS-BERT$_{\text{base}}$	STS-G	.792	.694
CLS-BERT$_{\text{clinical}}$	STS-G	.800	.694
CLS-BERT$_{\text{base}}$	STS-G + STS-PL	.801	.709
CLS-BERT$_{\text{base}}$	STS-G + STS-PS	.800	.702
HConvBERT$_{\text{base}}$	STS-B	.776	.698
HConvBERT$_{\text{clinical}}$	STS-B	.719	.655
HConvBERT$_{\text{base}}$	STS-B + STS-PL	.798	.713
HConvBERT$_{\text{base}}$	STS-B + STS-PS	.798	.716
HConvBERT$_{\text{base}}$	STS-G	.799	*.727*
HConvBERT$_{\text{clinical}}$	STS-G	.764	.690
HConvBERT$_{\text{base}}$	STS-G + STS-PL	.803	.712
HConvBERT$_{\text{base}}$	STS-G + STS-PS	.806	.723

Table 5: Results on MedSTS, based on fine-tuning on STS-B, STS-G, STS-PS and STS-PL.

model descriptors – "base" and "clinical" – correspond to the two pretraining configurations, general and clinical. The "Data" column indicates the corpus used for fine-tuning, and A+B means that the model is first fine-tuned on A then fine-tuned on B. The model using general ("base") pretraining and fine-tuning only on STS-B or STS-G is referred to as the "general STS model".

Both pretraining using in-domain text ("clinical") and fine-tuning on pseudo-labelled data (+STS-PS/STS-PL) improve performance over the general STS model, with fine-tuning on pseudo-labelled data generally performing better than domain pretraining, in addition to being computationally cheaper.

It may be argued that the performance improvement is gained simply as a result of using an enlarged data set for fine-tuning, instead of learning domain characteristics from clinical pseudo-labelled data. However, for both datasets, and under CLS-BERT$_{\text{base}}$ and HConvBERT$_{\text{base}}$, comparing results using: (1) STS-B with size of 5,749; (2) STS-B + STS-PS with size of 11,172 (5,749 + 5.423); and (3) STS-G with size of 28,518, we find that both (2) and (3) have higher r and ρ than (1), suggesting that enlarging the data size for fine-tuning is beneficial to improving performance. Simutaneously, (2) always performs much better than (3) though (3) is larger and has more gold labels; this indicates the gains are mainly attributable to learned domain characteristics rather than merely increased data. Moreover, based on the results for CLS-BERT$_{\text{base}}$ and HConvBERT$_{\text{base}}$ using STS-PL and STS-PS, it would appear that the amount and score distribution of the pseudo-labelled data influences fine-tuning performance, which we return to investigate further in Section 6.4.

6.3 Combination of Domain Pretraining (DP) and Fine-tuning

We adapt CLS-BERT$_{\text{clinical}}$-STS-G to predict scores for 500,000 pairs, generating STS-DP (6,306) after sampling as shown in Table 3. We continue to fine-tune CLS-BERT$_{\text{clinical}}$-STS-G using STS-DP, boosting the performance to $r = .803$ and $\rho = .788$, from $r = .788$ and $\rho = .768$.

6.4 Impact of Data Distribution and Amount

In this section, we investigate how data source, score distribution — percentage of instances distributed in five score interval, and the volume of sampled instances influence fine-tuning performance. Based on CLS-BERT$_{\text{base}}$ with STS-G, we continue to fine-tune over five different groups of data: (1) N2C2-STS training data without gold-standard labels, where the score distribution of pseudo labels is $0.04, 0.15, 0.25, 0.35, 0.21$; (2) data sampled from STS-PL in the same volume

Exp.	Source	Amount	Score distribution	r	ρ
0	—	0	—	.721	.720
1	N2C2-STS	1642	$0.04, 0.15, 0.25, 0.35, 0.21$	**.788**	**.788**
2	STS-PL	1642	$0.04, 0.15, 0.25, 0.35, 0.21$	.766	.738
3	STS-PL	1650	$0.20, 0.20, 0.20, 0.20, 0.20$	.761	.731
4	STS-PL	1648	$0.24, 0.24, 0.24, 0.24, 0.03$	.767	.748
5	STS-PL	16501	$0.24, 0.24, 0.24, 0.24, 0.03$	*.781*	*.767*

Table 6: Results for CLS-BERT$_{\text{base}}$-STS-G on N2C2-STS based on fine-tuning on different datasets. Exp.1 is N2C2-STS train data removing gold-standard labels, Exp.2 is sampled from STS-PL with same score distribution as Exp.1, Exp.3 is uniformly sampled from STS-PL, Exp.4 is proportionally sampled from STS-PL and Exp.5 is full STS-PL.

and with the score distribution as (1); (3) uniformly sampled from STS-PL with 330 pairs in each score interval; (4) proportionally sampled from STS-PL at a ratio of $1/10$ for each score interval; and (5) full STS-PL.

Comparing Experiments 2, 3 and 4 in Table 6, which have same data source and size (1.6k), and differ only in score distribution, we observe only minor performance differences. Experiments 1 and 2 rely on different sources, where Experiment 1 has the same source as the test data, and performs much better than Experiment 2. An aligned data source therefore is the optimal scenario. Looking at Experiments 4 and 5, where the difference is in the amount of sampled data, it is clear that more instances brings further improvements. But *Could performance be improved consistently with increased pseudo-labelled data?*

To answer this question, we proportionally sampled from STS-PL by ratio of 0.1, 0.2, 0.3, 0.4, 0.6, 0.8, 1.0, and also sampled from 500k unlabelled sentence pairs setting $N = 5000$, 6000, 7000, 7500, 8000, resulting in 12 subsets in sizes ranging from 1,648 to 28,456, for fine-tuning based on CLS-BERT$_{\text{base}}$-STS-G. As shown in Figure 1,[1] from 0 to 16,501, both r and ρ gradually increase, and then fluctuate around 0.77 and 0.76 resp. This reveals the trade-off between increasing the number of pseudo-labelled fine-tuning instances and error propagation due to cumulative noise.

6.5 Impact of Number of Iterations

Based on CLS-BERT$_{\text{base}}$ with STS-G, we investigate the impact of multiple iterations of fine-tuning in Table 7, as introduced in Section 5.2. The performance boost from additional iterations is modest. Increasing iterations from 2 to 3, the accuracy does not improve, which is consistent with the findings in Figure 1.

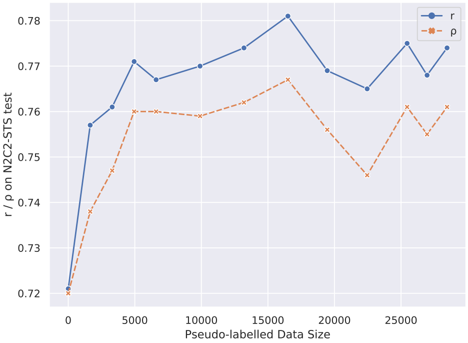

Figure 1: Impact of pseudo-labelled data size on N2C2-STS test.

Iteration	Amount	Score distribution	r	ρ
1	16501	$0.243, 0.243, 0.243, 0.243, 0.027$	.781	.767
2	22205	$0.245, 0.245, 0.245, 0.245, 0.020$	.788	.765
3	27320	$0.245, 0.245, 0.245, 0.245, 0.020$	.788	.759

Table 7: Results on N2C2-STS through differing number of iterations of iterative fine-tuning. Amount = number of fine-tuning instances.

7 Conclusion

In this paper, we have proposed a simple method of pseudo-labelling in-domain data and iterative training, to improve clinical STS. Evaluation over two clinical STS datasets demonstrates the effectiveness of the approach, and domain pretraining is shown to achieve further improvements. Further investigation indicated that keeping the distribution of pseudo-labelled instances close to that of the in-domain data improves performance. We also observed modest improvements through more iterations of iterative training. Our work provides an alternative approach to employing domain-specific unlabelled data to support clinical STS. As future work, we plan to explore the application of our method to other model structures such as SBERT.

Acknowledgments

This work was supported in part by China Scholarship Council (CSC). We are grateful to the anonymous reviewers for their insightful comments.

[1] Random sampling affects the model performance, particularly when the data size is less than 5000, so we sampled five times for 1648, 3300 and 4948, so these results are averages over multiple samples of the given size.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, et al. 2015. Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 252–263.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. Semeval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 81–91.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 497–511.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. * sem 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43.

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 385–393. Association for Computational Linguistics.

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online. Association for Computational Linguistics.

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, Minneapolis, Minnesota, USA.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada.

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

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-Wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Marco Marelli, Luisa Bentivogli, Marco Baroni, Raffaella Bernardi, Stefano Menini, and Roberto Zamparelli. 2014. Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 1–8.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Nina Poerner and Hinrich Schütze. 2019. Multi-view domain adapted sentence embeddings for low-resource unsupervised duplicate question detection. 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 1630–1641, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Alexey Romanov and Chaitanya Shivade. 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. Association for Computational Linguistics.

Yanshan Wang, Naveed Afzal, Sunyang Fu, Liwei Wang, Feichen Shen, Majid Rastegar-Mojarad, and Hongfang Liu. 2018. MedSTS: a resource for clinical semantic textual similarity. *Language Resources and Evaluation*, pages 1–16.

Yanshan Wang, Sunyang Fu, and Hongfang Liu. 2020a. Overview of the 2019 n2c2/ohnlp track on clinical semantic textual similarity. Preprint.

Yuxia Wang, Fei Liu, Karin Verspoor, and Timothy Baldwin. 2020b. Evaluating the utility of model configurations and data augmentation on clinical semantic textual similarity. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 105–111, Online. Association for Computational Linguistics.

Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2020. Unsupervised alignment-based iterative evidence retrieval for multi-hop question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4514–4525, Online. Association for Computational Linguistics.

Shuyan Zhou, Shruti Rijhwani, John Wieting, Jaime Carbonell, and Graham Neubig. 2020. Improving candidate generation for low-resource cross-lingual entity linking. *Transactions of the Association for Computational Linguistics*, 8:109–124.