

Probing Biomedical Embeddings from Language Models

Qiao Jin

University of Pittsburgh
qiao.jin@pitt.edu

Bhuwan Dhingra

Carnegie Mellon University
bdhingra@cs.cmu.edu

William W. Cohen

Google, Inc.
wcohen@google.com

Xinghua Lu

University of Pittsburgh
xinghua@pitt.edu

Abstract

Contextualized word embeddings derived from pre-trained language models (LMs) show significant improvements on downstream NLP tasks. Pre-training on domain-specific corpora, such as biomedical articles, further improves their performance. In this paper, we conduct probing experiments to determine what additional information is carried *intrinsically* by the in-domain trained contextualized embeddings. For this we use the pre-trained LMs as fixed feature extractors and restrict the downstream task models to not have additional sequence modeling layers. We compare BERT (Devlin et al., 2018), ELMo (Peters et al., 2018a), BioBERT (Lee et al., 2019) and BioELMo, a biomedical version of ELMo trained on 10M PubMed abstracts. Surprisingly, while fine-tuned BioBERT is better than BioELMo in biomedical NER and NLI tasks, as a fixed feature extractor BioELMo outperforms BioBERT in our probing tasks. We use visualization and nearest neighbor analysis to show that better encoding of entity-type and relational information leads to this superiority.

1 Introduction

NLP has seen an upheaval in the last year, with contextual word embeddings, such as ELMo (Peters et al., 2018a) and BERT (Devlin et al., 2018), setting state-of-the-art performance on many tasks. These empirical successes suggest that unsupervised pre-training from large corpora could be a vital part of NLP models. In specific domains like biomedicine, NLP datasets are much smaller than their general-domain counterparts¹, which leads to a lot of ad-hoc models: some infer through knowledge bases (Chandu

et al., 2017), while others leverage large-scale general domain datasets for domain adaptation (Wiese et al., 2017). However, unlabeled biomedical texts are abundant, and their full potential has perhaps not yet been fully realized.

We train a domain-specific version of ELMo on 10M PubMed abstracts, called BioELMo². Experiments on biomedical named entity recognition (NER) dataset BC2GM (Smith et al., 2008) and biomedical natural language inference (NLI) dataset MedNLI (Romanov and Shivade, 2018) clearly show the utility in training in-domain contextual word representations, but we would also like to know exactly what extra information is carried *intrinsically* in these embeddings.

To answer this question, we design two *probing tasks*, one for NER and one for NLI, where contextualized embeddings are used solely as fixed feature extractors and no sequence modeling layers are allowed above the embeddings. This setting prohibits the model from capturing task-specific contextual patterns, and instead only utilizes the information already present in the representations. In parallel to our work of BioELMo, Lee et al. (2019) introduce BioBERT, which is a biomedical version of in-domain trained BERT. We also probe BioBERT in our experiments.

Expectedly, BioELMo and BioBERT perform significantly better than their general-domain counterparts. When fine-tuned, BioBERT outperforms BioELMo, however, when used as fixed feature extractors, BioELMo is better than BioBERT in our probing tasks. Visualizations and nearest neighbor analyses suggest that it’s because BioELMo more effectively encodes entity-types and information about biomedical relations, such as disease and symptom interactions, than BioBERT.

¹For example, MedNLI (Romanov and Shivade, 2018) only has about 11k training instances while the general domain NLI dataset SNLI (Bowman et al., 2015) has 550k.

²Available at <https://github.com/Andy-jqa/bioelmo>.

2 Related Work

Embeddings from Language Models: ELMo (Peters et al., 2018a) is a pre-trained deep bidirectional LSTM (biLSTM) language model. ELMo word embeddings are computed by taking a weighted sum of the hidden states from each layer of the LSTM. The weights are learned along with the parameters of a task-specific downstream model, but the LSTM layers are kept fixed. Recently, Devlin et al. (2018) introduced BERT, and they showed that pre-training transformer networks on a masked language modeling objective leads to even better performance by fine-tuning the transformer weights on a broad range of NLP tasks. We study biomedical in-domain versions of these contextualized word embeddings in comparison to the general ones.

Biomedical Word Embeddings: Context-independent word embeddings, such as word2vec (w2v) (Mikolov et al., 2013) trained on biomedical corpora, are widely used in biomedical NLP models. Some recent works reported better NER performance with in-domain trained ELMo than general ELMo (Zhu et al., 2018; Sheikhshab et al., 2018). Lee et al. (2019) introduce BioBERT, which is BERT pre-trained on biomedical texts and set new state-of-the-art performance on several biomedical NLP tasks. We reaffirm these results on biomedical NER and NLI datasets with in-domain trained contextualized embeddings, and further explore *why* they are superior.

Probing Tasks: Designing tasks to probe sentence or token representations for linguistic properties has been a widespread practice in NLP. InferSent (Conneau et al., 2017) uses transfer tasks to probe for sentence embeddings pre-trained on supervised data. Many studies (Dasgupta et al., 2018; Poliak et al., 2018) design new test sets to probe for specific linguistic signals in sentence representations. Tasks to probe for token-level properties are explored by Blevins et al. (2018); Peters et al. (2018b), where they test whether token embeddings from different pre-training schemes encode part-of-speech and constituent structure.

Tenney et al. (2018) extend token-level probing to span-level probing and consider a broader range of tasks. Our work is different from them in the following ways – (1) We probe for biomedical domain-specific contextualized embeddings and

compare them to the general-domain embeddings; (2) For NER, instead of classifying the tag for a given span, we adopt an end-to-end setting where the spans must also be identified. This allows us to compare the probing results to state-of-the-art numbers; (3) We also probe for relational information using the NLI task in an end-to-end style.

3 Methods

3.1 Biomedical Contextual Embeddings

BioELMo: We train BioELMo on the **PubMed** corpus. PubMed provides access to MEDLINE, a database containing more than 24M biomedical citations³. We used 10M recent abstracts (2.46B tokens) from PubMed to train BioELMo. The statistics of this corpus are very different from more general domains. For example, the token **patients** ranks 22 by frequency in the PubMed corpus while it ranks 824 in the 1B Word Benchmark dataset (Chelba et al., 2013). We use the Tensorflow implementation⁴ of ELMo to train BioELMo. We keep the default hyperparameters and it takes about 1.7K GPU hours to train 8 epochs. BioELMo achieves an averaged forward and backward perplexity of 31.37 on test set.

BioBERT: In parallel to our work, Lee et al. (2019) developed BioBERT, which is pre-trained on English Wikipedia, BooksCorpus and fine-tuned on PubMed (7.8B tokens in total). BioBERT was initialized with BERT and further trained on PubMed for 200K steps.⁵

To get fixed features of tokens, we use the learnt downstream task-specific layer weights to calculate the average of 3 layers (1 token embedding layer and 2 biLSTM layers) for BioELMo and 13 layers (1 token embedding layer and 12 transformer layers) for BioBERT. As fixed feature extractors, BioELMo and BioBERT are not fine-tuned by downstream tasks.

3.2 Downstream Tasks

We first use BioELMo with state-of-the-art models and fine-tune BioBERT on the downstream tasks, to test their full capacity. In §3.3 we introduce our probing setup which tests BioBERT and

³<https://www.ncbi.nlm.nih.gov/pubmed/>

⁴<https://github.com/allenai/bilm-tf>

⁵We note there is a difference in the size of training corpora for BioBERT and BioELMo, but since we trained BioELMo before BioBERT was available, we could not control for this difference.

<p>Premise: He returned to the clinic three weeks later and was prescribed with antibiotics.</p> <p>Hypothesis: The patient has an infection.</p> <p>Label: Entailment</p>

Figure 1: Relation information in a MedNLI instance.

BioELMo as fixed feature extractors.

NER: For BioELMo, following Lample et al. (2016), we use the contextualized embeddings and a character-based CNN for word representations, which are fed to a biLSTM, followed by a conditional random field (CRF) (Lafferty et al., 2001) layer for tagging. For BioBERT, we use the single sentence tagging setting described in Devlin et al. (2018), where the final hidden states of each token are trained to classify its NER label.

NLI: For BioELMo, We use the ESIM model (Chen et al., 2016), which encodes the premise and hypothesis using biLSTM. The encodings are fed to a local inference layer with attention, another biLSTM layer and a pooling layer followed by softmax for classification. For BioBERT, we use the sentence pair classification setting described in Devlin et al. (2018), where the final hidden states of the first token (special ‘[CLS]’) are trained to classify the NLI label for the sentence pair.

3.3 Probing Tasks

We design two probing tasks where the contextualized embeddings are only used as fixed feature extractors and restrict the down-stream models to be non-contextual, to investigate the information intrinsically carried by them. One task is on NER to probe for entity-type information, and the other is on NLI to probe for relational information.

NER Probing Task: As shown in Figure 2 (left), we embed the input tokens to $\mathbf{R} = [\mathbf{E}_1; \mathbf{E}_2; \dots; \mathbf{E}_L] \in \mathbb{R}^{L \times D_e}$, where L is the sequence length and D_e is embedding size. The embeddings are fed to several feed-forward layers:

$$\widetilde{\mathbf{E}}_i = \text{FFN}(\mathbf{E}_i) \in \mathbb{R}^T$$

where T is the number of tags. $[\widetilde{\mathbf{E}}_1; \widetilde{\mathbf{E}}_2; \dots; \widetilde{\mathbf{E}}_L]$ is then fed to a CRF output layer. CRF doesn’t model the context but ensures the global consistency across the assigned labels, so it’s compatible with our probing task setting.

NLI Probing Task: Relational information between tokens of premises and hypotheses is vital to solve MedNLI task: as shown in Figure 1, the hypothesis is an entailment because **antibiotics** are used to treat an **infection**, which is a drug-disease

relation. We design the task shown in Figure 2 (right) to probe such relational information: We embed the premise and hypothesis separately to $\mathbf{P} \in \mathbb{R}^{L_1 \times D_e}$ and $\mathbf{H} \in \mathbb{R}^{L_2 \times D_e}$, where L_1, L_2 are sequence lengths. Then we use bilinear layers⁶ to get $\mathbf{S} = [\mathbf{S}_1; \mathbf{S}_2; \dots; \mathbf{S}_R] \in \mathbb{R}^{R \times L_1 \times L_2}$ where

$$\mathbf{S}_r = \mathbf{P} \mathbf{W}_r \mathbf{H}^T \in \mathbb{R}^{L_1 \times L_2},$$

and $\mathbf{W}_r \in \mathbb{R}^{D_e \times D_e}$ is the weight matrix of a bilinear layer. Note that each element of \mathbf{S}_r encodes the interaction between a token from the premise and a token from the hypothesis. We denote

$$\mathbf{h}_{ij} = [\mathbf{S}_1[i, j] \quad \dots \quad \mathbf{S}_R[i, j]]^T \in \mathbb{R}^R, \quad (1)$$

as the **distributed relation representation** between token i in premise and token j in hypothesis, and R is the tunable dimension of it. We then apply an element-wise maximum pooling layer:

$$\widetilde{\mathbf{h}} = \max_{i,j} \mathbf{h}_{ij} \in \mathbb{R}^R.$$

We use a linear layer to compute the softmax logits of the NLI labels, e.g. $p(\text{entailment}) \propto \exp(\widetilde{\mathbf{h}}^T \mathbf{w}_{\text{ent}})$, where \mathbf{w}_{ent} is the learnt weight vector corresponding to the entailment label.

For BERT, we probe two variants. The first, denoted as BERT / BioBERT, feeds the premise and hypothesis to the model separately. The second, denoted as BERT-tog / BioBERT-tog, concatenates the two sentences by the ‘[SEP]’ token and feeds to the model *together* to get the embeddings. This is how BERT is supposed to be used for sentence pair classification tasks, but it’s not comparable to ELMo in our setting since ELMo doesn’t take two sentences together as input.

4 Experiments

4.1 Experimental Setup

Data: For the NER task, we use the BC2GM dataset. BC2GM stands for BioCreative II gene mention dataset (Smith et al., 2008). The task is to detect gene names in sentences. It contains 15k training and 5k test sentences. We also test on the general-domain CoNLL 2003 NER dataset (Tjong Kim Sang and De Meulder, 2003), where the task is to detect entities such as person and location.

For the NLI task, we use the MedNLI dataset (Romanov and Shivade, 2018), where the task is, given a pair of sentences (premise and hypothesis), to predict whether the relation of entailment, contradiction, or neutral (no relation) holds between

⁶We also tried models without bilinear layers, which turn out to be suboptimal.

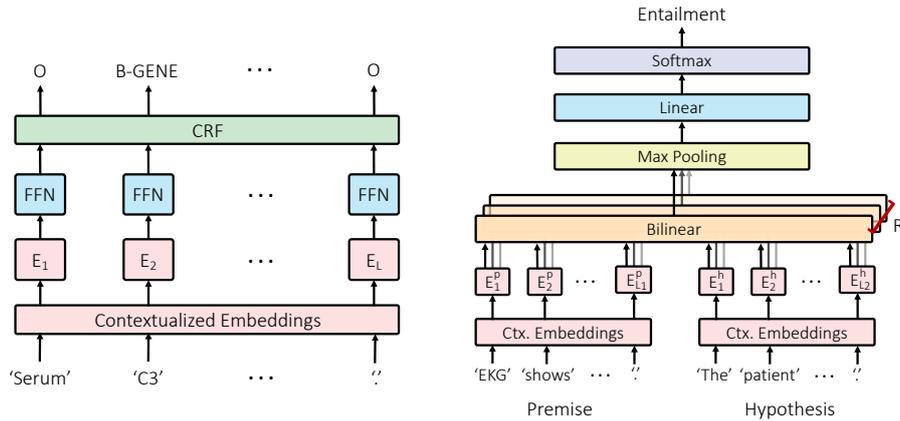


Figure 2: **Left:** NER probing task. The contextual word representations are directly used to predict the NER labels, followed by a CRF layer to ensure label consistency. **Right:** NLI probing task. Bilinear operators map pairs of word representations to relation representations which are used to predict the entailment label.

them. The premises are sampled from doctors’ notes in the clinical dataset MIMIC-III (Johnson et al., 2016). The hypotheses and annotations are generated by clinicians. It contains 11,232 training, 1,395 development and 1,422 test instances. We also test on the general-domain SNLI dataset (Bowman et al., 2015), where the premises and hypotheses are drawn from image captions.

Compared Settings: For each dataset, the **Whole** setting refers to the state-of-the-art model we used (described in §3.2), including contextual modeling layers or fine-tuning of the embedding encoder. **Probing** and **Control** settings describe the probing task model introduced in §3.3. The control setting tests the representations on a general-domain dataset/task, to check whether we lose any information in domain-specific embeddings. Probing and control results are averaged over three seeds. **Compared Embeddings:** We compare: (1) non-contextual biomedical w2v trained on a biomedical corpus of 5.5B tokens (Moen and Ananiadou, 2013), (2) ELMo trained on a general-domain corpus of 5.5B tokens⁷, (3) BioELMo⁸, (4) Cased base version of BERT trained on a general-domain corpus of 3.3B tokens⁹ and (5) BioBERT¹⁰.

4.2 Main Results

4.2.1 NER Results

In Domain v.s. General Domain: Results in Table 1 show that BioBERT and BioELMo in

⁷<https://allennlp.org/elmo>

⁸Though BioELMo uses the smallest corpus to train, it performs better than BioBERT in probing setting, and general ELMo in whole and probing setting.

⁹<https://github.com/google-research/bert>

¹⁰<https://github.com/dmis-lab/biobert>

Method	F1 (%)		
	Whole	Probe	Ctrl.
Ando (2007)	87.2	–	–
Rei et al. (2016)	88.0	–	–
Sheikhshab et al. (2018)	89.7	–	–
Biomed w2v	84.9	78.5	67.5
General ELMo	87.0	82.9	84.0
General BERT	89.2	84.9	83.6
BioELMo	90.3	88.4	80.9
BioBERT	90.6	88.2	83.4

Table 1: NER test results. **Whole:** whole model performance on BC2GM; **Probe:** Probing task performance on BC2GM; **Ctrl.:** Probing task performance on CoNLL 2003 NER. We use the official evaluation codes to calculate the F1 scores where there are multiple ground-truth tags, so the F1 scores are much higher than what were reported in Lee et al. (2019).

the Whole setting perform better than the general BERT and ELMo and biomed w2v, setting new state-of-the-art performance for this dataset.

BioBERT and BioELMo remains competitive in the Probing setting, doing much better than their general domain counterparts and even general ELMo in the Whole setting. This shows that with the right pre-training, the downstream model can be considerably simplified.

Unsurprisingly, in the Control setting BioBERT and BioELMo do worse than their general counterparts, indicating that the gains come at the cost of losing some general-domain information. However, the performance gaps (absolute differences) between ELMo and BioELMo are larger in the biomedical domain than it is in the general domain, which is also true for BERT and BioBERT. For ELMo and BioELMo, we believe it is because the PubMed corpus contains many mentions of

general-domain entities whereas the reverse is not true. Because BioBERT is initialized with BERT and also uses general-domain corpora like English Wikipedia for pre-training, it’s not surprising that BioBERT is just 0.2 worse than BERT on CoNLL 2003 NER in control setting.

BioELMo v.s. BioBERT: Fine-tuned BioBERT outperforms BioELMo with biLSTM and CRF on BC2GM. As a feature extractor, BioBERT is slightly worse than BioELMo in probing task of BC2GM, but outperforms BioELMo in probing task of CoNLL 2003, which can be explained by the fact that BioBERT is also pre-trained on general-domain corpora.

Method	Accuracy (%)		
	Whole	Probe	Ctrl.
Romanov and Shivade (2018)	76.6	–	–
Biomed w2v	74.2	71.1	59.2
General ELMo	75.8	69.6	60.8
General BERT	–	67.6	62.1
General BERT-tog	77.8	71.0	74.1
BioELMo	78.2	75.5	58.3
BioBERT	–	70.1	58.8
BioBERT-tog	81.7	73.8	69.9

Table 2: NLI test results. **Whole**: whole model performance on MedNLI; **Probe**: Probing task performance on MedNLI; **Ctrl.**: Probing task performance on SNLI. To make the results comparable, we only use the same number of SNLI training instances as that of MedNLI.

4.2.2 NLI Results

In Domain v.s. General Domain: Table 2 shows that BioBERT and BioELMo in the Whole setting perform better than their general domain counterparts and biomedical w2v for NLI, setting state-of-the-art performance for this dataset as well.

Once again, we observe that BioBERT and BioELMo outperform their general domain counterparts in the Probing settings, which comes at the cost of losing general domain information as indicated in the Control setting results.

Note that the Probing task *only* models relationships between tokens, but we still see competitive accuracy in that setting (75.5% vs 76.6% previous best). This suggests that, (i) many instances in MedNLI can be solved by identifying token-level relationships between the premise and the hypothesis, and (ii) BioELMo already captures this kind of information in its embeddings.

BioELMo v.s. BioBERT: Fine-tuned BioBERT does much better than BioELMo with ESIM model. However, BioELMo performs better than

BioBERT by a large margin in the probing task of MedNLI. We explore this in more detail in the next section. Again, BioBERT is better than BioELMo in probing task of SNLI because it’s also pre-trained on general corpora.

We notice that the *-tog* setting improves the BERT performance. Encoding two sentences separately, BioELMo still outperforms BioBERT-tog. It suggests that BioELMo is a better feature extractor than BioBERT, even though the latter has superior performance when fine-tuned on MedNLI.

4.3 Analysis

4.3.1 Entity-type Information

In biomedical literature, the acronym **ER** has multiple meanings: out of the 124 mentions we found in 20K recent PubMed abstracts, 47 refer to the gene “estrogen receptor”, 70 refer to the organelle “endoplasmic reticulum” and 4 refer to the “emergency room” in hospital. We use t-SNE (Maaten and Hinton, 2008) to visualize different contextualized embeddings of these mentions in Figure 3.

In Domain v.s. General Domain: For general ELMo, by far the strongest signal separating the mentions is whether they appear inside or outside parentheses. This is not surprising given the recurrent nature of LSTM and language modeling training objective for learning these embeddings. BioELMo does a better job of grouping mentions of the same entity (ER as estrogen receptor) together, which is clearly helpful for the NER task.

ER mentions of the same entities cluster better by BioBERT than general BERT: there are two major clusters corresponding to estrogen receptor and endoplasmic reticulum by BioBERT as indicated by the dashed circles, while entities of different types are scattered almost evenly by BERT.

BioELMo v.s. BioBERT: Clearly BioELMo better clusters entities from the same types together. Unlike ELMo/BioELMo, Whether the ER mention is inside parentheses doesn’t affect BERT/BioBERT representations. It can be explained by encoder difference between ELMo and BERT: For ELMo, to predict ‘)’ in forward LM, representations of token ‘ER’ inside the parentheses need to encode parentheses information due to the recurrent nature of LSTM. For BERT, to predict ‘)’ in masked LM, the masked token can attend to ‘(’ without interacting with ‘ER’ representations, so BERT ‘ER’ embedding does’t need to encode parentheses information.

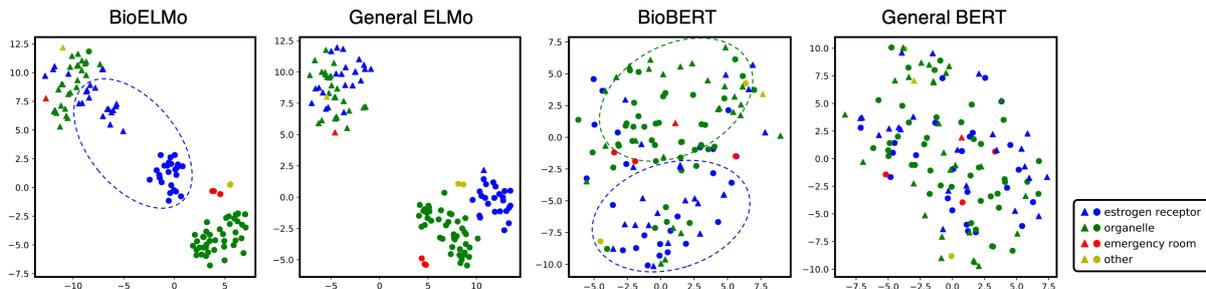


Figure 3: t-SNE visualizations of the token **ER** embeddings in different contexts by BioELMo, general ELMo, BioBERT and general BERT. ● and ▲ represent **ER** mentions within and outside of parentheses, respectively. Colors refer to different actual meanings of the **ER** mention.

Relation Type	NN w/ Representation of Same Type (%)						
	BioELMo	ELMo	BioBERT-tog	BioBERT	BERT-tog	BERT	Biomed w2v
disease-symptom	54.2	52.1	44.5	38.8	34.2	37.0	40.9
disease-drug	32.8	34.4	26.1	17.9	27.7	22.6	23.6
number-indication	70.5	63.9	47.0	45.3	48.1	49.5	74.4
synonyms	63.6	56.4	60.8	55.8	56.4	52.8	51.7
All	57.5	53.3	47.1	42.1	43.3	42.5	49.5
Subset Accuracy (%)	73.9	62.8	71.4	65.0	65.8	64.5	69.7

Table 3: Average proportion of nearest neighbor (NN) representations that belong to the same type for different embeddings, averaged over three random seeds. Biomed w2v performs best for number-indication relations, probably because it uses a vocabulary of over 5M tokens, in which about 100k are numbers. Subset accuracy denotes the probing task performance in the subset of MedNLI test set used for this analysis.

4.3.2 Relational Information

We manually examine all test instances with the “entailment” label in MedNLI, and found 78 token pairs across the premises and hypotheses which strongly suggest entailment. Among them, 22 are disease-symptom pairs, 13 are disease-drug pairs, 19 are numbers and their indications (e.g.: 150/93 and hypertension) and 24 are synonyms or closely related concepts (e.g.: Lasix[®] and diuretic). Figure 1 shows an example of disease-drug relationship. We hypothesize that a model is required to encode relation information to perform well in MedNLI task. We evaluate relation representations from different embeddings by nearest neighbor (NN) analysis: For each distributed relation representation (Eq. 1) of these token pairs, we calculated the proportions of its five nearest neighbors that belong to the same relation type. We report the average proportions in table 3 and use it as a metric to measure the effectiveness of representing relations by different embedding schemes. We also show model performance for this subset (78 instances for relation analysis) in table 3. The trends of subset accuracy moderately correlate with the NN proportions (Pearson correlation coefficient $r = 0.52$).

In Domain v.s. General Domain: For all relations, BioELMo is significantly¹¹ better than ELMo in representing same relations closer to each other, while there is no significant difference between BioBERT and BERT. This indicates that even pre-trained by in-domain corpus, as fixed feature extractor, BioBERT still cannot effectively encode biomedical relations compared to BERT. **BioELMo v.s. BioBERT:** BioELMo significantly outperforms BioBERT and even BioBERT-tog for all relations. This explains why BioELMo does better than BioBERT in the probing task: BioELMo better represents vital biomedical relations between tokens in premises and hypotheses.

5 Conclusion

We have shown that BioELMo and BioBERT representations are highly effective on biomedical NER and NLI, and BioELMo works even without complicated downstream models and outperforms untuned BioBERT in our probing tasks. This effectiveness comes from its ability as a fixed feature extractor to encode entity types and especially their relations, and hence we believe they should

¹¹Significance is defined as $p < 0.05$ in two-proportion z test.

benefit any task which requires such information.

A long-term goal of NLP is to learn universal text representations. Our probing tasks can be used to test whether learnt representations effectively encode entity-type or relational information. Moreover, comprehensive characterizations of BioELMo and BioBERT as fixed feature extractors would also be an interesting further direction to explore.

6 Acknowledgement

We are grateful for the anonymous reviewers who gave us very insightful suggestions. Bhuwan Dhingra is supported by a grant from Google.

References

- Rie Kubota Ando. 2007. Biocreative ii gene mention tagging system at ibm watson. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, volume 23, pages 101–103. Centro Nacional de Investigaciones Oncológicas (CNIO) Madrid, Spain.
- Terra Blevins, Omer Levy, and Luke Zettlemoyer. 2018. Deep rnns encode soft hierarchical syntax. *arXiv preprint arXiv:1805.04218*.
- Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.
- Khyathi Chandu, Aakanksha Naik, Aditya Chandrasekar, Zi Yang, Niloy Gupta, and Eric Nyberg. 2017. Tackling biomedical text summarization: Oaqa at bioasq 5b. *BioNLP 2017*, pages 58–66.
- Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2013. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.
- Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2016. Enhanced lstm for natural language inference. *arXiv preprint arXiv:1609.06038*.
- Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*.
- Ishita Dasgupta, Demi Guo, Andreas Stuhlmüller, Samuel J Gershman, and Noah D Goodman. 2018. Evaluating compositionality in sentence embeddings. *arXiv preprint arXiv:1802.04302*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- 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*, 3:160035.
- John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.
- Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: pre-trained biomedical language representation model for biomedical text mining. *arXiv preprint arXiv:1901.08746*.
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- SPFGH Moen and Tapio Salakoski2 Sophia Ananiadou. 2013. Distributional semantics resources for biomedical text processing. In *Proceedings of the 5th International Symposium on Languages in Biology and Medicine, Tokyo, Japan*, pages 39–43.
- Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.
- Matthew E Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. *arXiv preprint arXiv:1808.08949*.
- Adam Poliak, Aparajita Haldar, Rachel Rudinger, J Edward Hu, Ellie Pavlick, Aaron Steven White, and Benjamin Van Durme. 2018. Collecting diverse natural language inference problems for sentence representation evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 67–81.
- Marek Rei, Gamal KO Crichton, and Sampo Pyysalo. 2016. Attending to characters in neural sequence labeling models. *arXiv preprint arXiv:1611.04361*.

- Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain. *arXiv preprint arXiv:1808.06752*.
- Golnar Sheikhsab, Inanc Birol, and Anoop Sarkar. 2018. In-domain context-aware token embeddings improve biomedical named entity recognition. In *Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*, pages 160–164.
- Larry Smith, Lorraine K Tanabe, Rie Johnson nee Ando, Cheng-Ju Kuo, I-Fang Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph M Friedrich, Kuzman Ganchev, et al. 2008. Overview of biocreative ii gene mention recognition. *Genome biology*, 9(2):S2.
- Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2018. What do you learn from context? probing for sentence structure in contextualized word representations.
- Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.
- Georg Wiese, Dirk Weissenborn, and Mariana Neves. 2017. Neural domain adaptation for biomedical question answering. *arXiv preprint arXiv:1706.03610*.
- Henghui Zhu, Ioannis Ch Paschalidis, and Amir Tahmasebi. 2018. Clinical concept extraction with contextual word embedding. *arXiv preprint arXiv:1810.10566*.