Relation Extraction using Multiple Pre-Training Models in Biomedical Domain

Satoshi Hiai[1], **Kazutaka Shimada**[1],
Taiki Watanabe[2], **Akiva Miura**[2]*, **Tomoya Iwakura**[2]
[1]Kyushu Institute of Technology, Fukuoka, Japan
[2]Fujitsu Ltd., Tokyo, Japan
[3]Atid Ltd., Tokyo, Japan
s_hiai@pluto.ai.kyutech.ac.jp, shimada@ai.kyutech.ac.jp
{watanabe-taiki, iwakura.tomoya}@fujitsu.com
akiva.miura@studio-atid.com

Abstract

The number of biomedical documents is increasing rapidly. Accordingly, a demand for extracting knowledge from large-scale biomedical texts is also increasing. BERT-based models are known for their high performance in various tasks. However, it is often computationally expensive. A high-end GPU environment is not available in many situations. To attain both high accuracy and fast extraction speed, we propose combinations of simpler pre-trained models. Our method outperforms the latest state-of-the-art model and BERT-based models on the GAD corpus. In addition, our method shows approximately three times faster extraction speed than the BERT-based models on the ChemProt corpus and reduces the memory size to one sixth of the BERT ones.

1 Introduction

The amount of biomedical documents is increasing rapidly. The documents contain valuable knowledge, such as chemical compound names and their relations. However, the current knowledge extraction is considerably manual. Therefore, a demand for extracting knowledge automatically from large-scale biomedical text data is increasing.

Biomedical relation extraction (RE) models based on BERT (Devlin et al., 2019) have shown great performance (Lee et al., 2019; Beltagy et al., 2019). The methods using BERT models pre-trained on biomedical corpora achieved state-of-the-art (SOTA) performance on several biomedical RE datasets. However, BERT models require a huge amount of computational resources and generally need a long time for extraction processes. By processing data in parallel with multiple computational resources, we can process larger text as compared with a single resource. However, a high-end GPU or a distributed environment for efficient computation is not available in many situations. Even if we can utilize such computational resources, substantial energy consumption becomes a problem (Strubell et al., 2019). Therefore, more lightweight and accurate RE models are expected.

In this paper, we construct a biomedical RE model by combining word embeddings obtained from multiple lightweight models. The RE model can be executed in a wide range of environments, such as a CPU and a middle-class GPU, by reducing memory consumption during the learning process or the inference process. During the inference process of our proposed model, the amount of calculation can be suppressed and the model can process documents at high speed. Since the calculation in each lightweight model is small scale, memory consumption can be suppressed. Furthermore, by selecting whether or not to utilize each word embeddings, we can customize the model to be suitable for the computer environment. Hence, our model can process faster than the BERT-based models in the inference process.

We adopt more lightweight pre-trained models: ELMo (Peters et al., 2018) and Contextual String Embeddings (CSE) (Akbik et al., 2018). ELMo is contextualized *word-level* embeddings from the language model (LM) based on multiple layers of bidirectional long-short term memories (Bi-LSTMs) (Hochreiter and Schmidhuber, 1997). On the other hand, CSE is *character-level* embeddings of each input word from LM based on single-layer Bi-LSTM. Subword information of words plays an important role in the estimation of kinds and features of chemicals since chemical names tend to contain characteristic sub-word patterns such as prefixes and suffixes. Therefore, we propose RE models that combine ELMo and CSE to utilize both word-level and character-level features effectively.

We investigate the effectiveness of our RE mod-

[*] Work done while the author was at Fujitsu Laboratories Ltd.

Proceedings of Recent Advances in Natural Language Processing, pages 530–537
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_060

els in terms of not only the accuracy but also the processing speed and the memory size. The contributions of this paper are as follows:

- We apply the strategy that feeds the complementary features from pre-training models to RE tasks in the biomedical domain: GloVe, ELmo, and CSE.
- The proposed model outperforms the latest SOTA F1 score on the GAD corpus. As a case study, we show that BERT-based models do not always produce the best performance.
- Our model performs approximately three times faster extraction speed than BERT-based models on the ChemProt corpus and reduces the memory size to one sixth of the BERT ones.

2 Related Work

Beltagy et al. (Beltagy et al., 2019) have proposed a method with pre-trained the BERT model called SciBERT. They pre-trained the BERT model on the large scale computer science and biomedical corpora. They constructed a new vocabulary of the BERT model for the tasks of science and biomedical domains. The SciBERT model with their vocabulary achieved SOTA performance on several benchmark tasks on the domains. Lee et al. (Lee et al., 2019) have proposed a model called BioBERT. They pre-trained the BERT model on a large scale biomedical corpus containing 4.5 billion words. They applied the model to biomedical NER, RE, and question answering tasks and achieved high performance on the benchmark tasks. For the RE tasks, they utilized the sentence classifier of the original version of BERT, which uses a [CLS] token for the classification of relations. They used pre-defined strings such as @GENE$ and @DISEASE$ to express a pair of target entities. For instance, a sentence with two target entities (gene and disease in this case) is represented as **Example 1**.

Example 1 *Serine at position 986 of @GENE$ may be an independent genetic predictor of angiographic @DISEASE$.*

Zhou et al. (Zhou et al., 2016) have proposed a relation classification model with an attention-based Bi-LSTM model. They used pre-defined indicator tags to express a pair of target entities. For instance, a sentence with a pair of target entities is represented as **Example 2**.

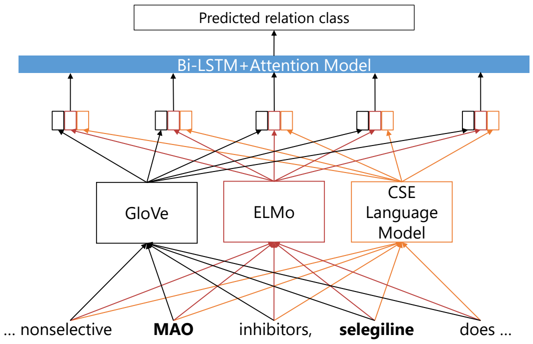

Figure 1: Overview of the proposed method

Example 2 *<e1> Flowers </e1> are carried into the <e2> chapel </e2>.*

Entity pairs were anonymized using the predefined strings in the method of Lee et al. (Lee et al., 2019). In contrast, this model can predict a relation class using surface information of the target entity pair. Sub-word information such as prefixes and suffixes plays an important role in estimations of kinds and features of chemicals. Therefore, we express the entity pair using tags in our method.

BERT is constructed by multiple layers of multi-head self-attention layers and requires large-scale computational resources. More lightweight pre-training LM models have also been proposed. Jin et al. (Jin et al., 2019) have proposed models for biomedical NLI tasks using ELMo pre-trained on large-scale in-domain text data. ELMo is an LM based on multi-layer Bi-LSTMs for aiming at obtaining contextualized word-level embeddings. CSE is generated by a character-level LM. The LM is lightweight since it is constructed with a single layer of Bi-LSTM. Sharma et al. (Sharma and Daniel Jr, 2019) have proposed a biomedical NER method with CSE generated from the LM pre-trained on a biomedical corpus. Watanabe et al. (Watanabe et al., 2019) have proposed a method with a multi-task learning model using CSE. Their method achieved SOTA performance on the biomedical NER task. Sharma et al. and Watanabe et al. evaluated the effectiveness of CSE on the biomedical NER tasks. However, they did not evaluate the effectiveness of CSE on the biomedical RE task. We evaluate the effectiveness of ELMo and CSE on the biomedical RE tasks.

3 Proposed Method

Figure 1 shows an overview of our method. We incorporate three types of word embeddings into the RE model: GloVe (Pennington et al., 2014), CSE, and ELMo. First, we train a character-level LM for CSE and a word-level LM for ELMo using large-scale biomedical corpora. Then we obtain GloVe, CSE, and ELMo vectors corresponding to each word in a sentence as shown in the middle of Figure 1. Next, we construct an RE model as shown in the top of Figure 1. We explain the pre-training procedure for the GloVe embeddings and the language models for ELMo and CSE vectors in Section 3.1. Then, we explain the RE model based on the combinations of multiple word embeddings in Section 3.2.

3.1 Pre-Training

We obtain word embeddings by concatenating GloVe, CSE, and ELMo vectors for training and extraction. We use the GloVe embeddings[1] trained on general domain corpora (the Wikipedia and the Gigaword corpus). For pre-training the CSE language model, we use the PubMed[2], the PMC[3], and the ChemRxiv[4]. The data from PubMed, PMC, and ChemRxiv contain 190k, 270k, and 300k biomedical papers, respectively. We use the ELMo embeddings[5] trained on the PubMed corpus.

3.2 Relation Extraction Model

For the RE task, we apply a Bi-LSTM with an attention model (Zhou et al., 2016). The relation extraction model outputs a predicted class label. Here, we express the stacked embeddings as $X = x_1, x_2, ..., x_n$. The predicted class label is computed as follows:

$$\begin{aligned} \overrightarrow{\mathbf{h_i}} &= \overrightarrow{LSTM}(\mathbf{x_i}, \overrightarrow{\mathbf{h}}_{\mathbf{i-1}}) \quad (1) \\ \overleftarrow{\mathbf{h_i}} &= \overleftarrow{LSTM}(\mathbf{x_i}, \overleftarrow{\mathbf{h}}_{\mathbf{i+1}}) \quad (2) \\ \mathbf{h_i} &= [\overrightarrow{\mathbf{h_i}}; \overleftarrow{\mathbf{h_i}}] \quad (3) \end{aligned}$$

where $\mathbf{x_i}$ is the i-th input vector. $\overrightarrow{\mathbf{h_i}}$ and $\overleftarrow{\mathbf{h_i}}$ are hidden states of the forward LSTM and backward LSTM, respectively. $[\cdot;\cdot]$ indicates concatenation of two vectors. We calculate a weight a_i for each hidden state $\mathbf{h_i}$ as follows:

$$\begin{aligned} m_i &= \boldsymbol{\omega}^T tanh(\mathbf{h_i}) \quad (4) \\ a_i &= \frac{\exp(m_i)}{\sum_{j=1}^{n} \exp(m_j)} \quad (5) \end{aligned}$$

$\boldsymbol{\omega}$ is a vector of trainable parameters. We obtain the final hidden state $\mathbf{h}^*$ as follows:

$$\begin{aligned} \mathbf{r} &= \textstyle\sum_{i=1}^{n} a_i \mathbf{h_i} \quad (6) \\ \mathbf{h}^* &= tanh(\mathbf{r}) \quad (7) \end{aligned}$$

Then, the model calculates a predicted label $\hat{\mathbf{y}}$ as follows:

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}) &= softmax(\mathbf{W_s}\mathbf{h}^* + \mathbf{b_s}) \quad (8) \\ \hat{\mathbf{y}} &= \arg\max_{\mathbf{y}} P(\mathbf{y}|\mathbf{X}) \quad (9) \end{aligned}$$

During training, we use the loss function:

$$Loss_{RE} = -\frac{1}{N}\sum_{i=1}^{N} log(p(\mathbf{y_i}|\mathbf{X})) \quad (10)$$

where N is the number of class labels.

We use an SGD optimizer (Bottou, 1991). We set parameters as follows: a learning rate is 0.1, a batch size is 32, and the number of hidden units is 256.

Dataset	Class		# samples
GAD	Positive		2,801
	Negative		2,529
ChemProt	Positive	CPR:3	1,973
		CPR:4	5,002
		CPR:5	471
		CPR:6	726
		CPR:9	1,814
		Total	9,986
	Negative		31,298

Table 1: Dataset statistics. The number of samples for ChemProt is the sum of the number of samples in training, validation, and test sets.

4 Experimental Settings

4.1 Dataset

We use the Genetic Association Database (GAD) (Bravo et al., 2015) and the Bio-Creative VI Chemical-Protein RE dataset (ChemProt) (Islamaj Doğan et al., 2017) as RE datasets to evaluate our model. Table 1 shows the statistics for each dataset.

The GAD dataset contains gene-disease relations. Relations between a gene and a disease within the same sentence were annotated. It is a binary classification task. The ChemProt dataset consists of 2,432 PubMed abstracts with chemical-protein relations annotated by domain experts. This

[1] https://nlp.stanford.edu/projects/glove/

[2] http://www.nlm.nih.gov/databases/download/pubmed_medline.html

[3] https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

[4] https://chemrxiv.org/

[5] https://allennlp.org/elmo

Dataset		E-SVM	SPINN	SciBERT	BioBERT	*Baseline* (GloVe)	*Proposed1* (GloVe +CSE)	*Proposed2* (GloVe +ELMo)	*Proposed3* (GloVe +CSE +ELMo)
GAD	P	79.21	-	**85.54**	76.43	77.31	78.74	83.10	82.64
	R	**89.25**	-	80.61	87.65	80.01	87.43	85.43	86.36
	F	83.93	-	82.83	81.61	78.59	82.83	84.16	**84.38**
ChemProt	P	-	74.85	**79.73**	77.02	58.13	67.41	76.49	76.05
	R	-	56.06	70.85	**75.90**	57.43	58.79	65.82	66.17
	F	-	64.11	75.03	**76.46**	57.78	62.81	70.76	70.77

Table 2: Experimental results. **Bold** and underline indicate the best score and the second best score, respectively. The values of E-SVM (Bhasuran and Natarajan, 2018), SPINN (Lim and Kang, 2018), and BioBERT (Lee et al., 2019) are referred from the original papers. For SciBERT (Beltagy et al., 2019), because the experimental setting in the paper differs from the setting of BioBERT, we re-evaluate it with the same setting as BioBERT.

dataset was used in the shared task of the BioCreative VI text mining chemical-protein interaction track. This dataset contains pairs of a chemical and a protein within the same sentence annotated with five kinds of relation labels: CPR: 3, CPR: 4, CPR: 5, CPR: 6, CPR: 9. The task is to classify an instance pair into one of the five classes and non-relation.

For the GAD dataset, there is no separate test set. Therefore, we follow Lee et al. (2019) and report the performance of a 10-fold cross-validation on the dataset. We used the same 10-folds as Lee et al. (2019) with the provided datasets on their webpage[6].

4.2 Target Entity Pair Indicators

To indicate the location of target entities, Zhou et al. (2016) used indicator tags to express a pair of target entities in an RE task. In the same way, we inserted tags before and after each target entity. For the GAD dataset, we use the <gene> and <dise> tags for genes and diseases, respectively. For the ChemProt dataset, we use the <gene> and <chem> tags for genes and chemicals, respectively. For instance, ***1-aminoadamantane*** and ***Fos*** are the target entities in **Example 3**.

Example 3 *Amantadine (<chem>* ***1-aminoadamantane*** *</chem>) induced <gene>* ***Fos*** *</gene> expression in the central, dorsal-medial and ventral-medial part of the striatum.*

4.3 Methods in this Experiment

We trained one baseline and three proposed models. Each method was based on a Bi-LSTM attention model with a combination of three different inputs. The trained models are followed:

- A Bi-LSTM attention model with GloVe (*Baseline*: GloVe)
- A Bi-LSTM attention model with GloVe and CSE (*Proposed1*: GloVe+CSE)
- A Bi-LSTM attention model with GloVe and ELMo (*Proposed2*: GloVe+ELMo)
- A Bi-LSTM attention model with GloVe, CSE, and ELMo (*Proposed3*: GloVe+CSE+ELMo)

We evaluate the effectiveness of each embedding through the comparison between the baselines and our method.

In addition, we compared our methods with the SOTA methods, namely BioBERT and SciBERT on ChemProt, and E-SVM on GAD dataset. BioBERT used the BERT model pre-trained on large-scale biomedical corpora. SciBERT used the BERT model pre-trained on large-scale biomedical and computer science corpora. The current SOTA method for GAD dataset was based on an ensemble of SVMs (Bhasuran and Natarajan, 2018) (E-SVM). SPINN was the previous SOTA method before the BERT-era that was not based on pre-trained LMs (Lim and Kang, 2018).

5 Experimental Results

5.1 Evaluation on accuracy

Table 2 shows the experimental results. For all the datasets, we reported precision, recall, and F-measure (F1) scores for the positive classes.

The F1 score of the *Proposed1* (GloVe+CSE) outperformed the F1 score of *Baseline* (GloVe) for GAD and ChemProt datasets. The F1 score of the *Proposed3* (GloVe+CSE+ELMo) outperformed *Baseline* and other all the proposed methods. The

[6] https://github.com/dmis-lab/biobert

Method	Time (s)	/ Sample (ms)
Proposed1 (GloVe+CSE)	86.53	5.99
Proposed2 (GloVe+ELMo)	212.51	14.71
Proposed3 (GloVe+CSE+ELMo)	239.06	16.55
SciBERT	537.32	37.20
BioBERT	680.39	47.11

Table 3: Extraction time required for the test-set of ChemProt.

result shows the effectiveness of LMs pre-trained on the biomedical corpora.

For the GAD dataset, the *Proposed3* showed the best performance. It outperformed SciBERT, BioBERT, and E-SVM. For the ChemProt dataset, although this method also outperformed SPINN, the previous SOTA method, it did not reach the scores of SciBERT and BioBERT.

Another approach to improve the accuracy is to incorporate a BERT model with our combination models. We also attempted to incorporate the BERT embeddings generated by the BioBERT model into our model although the result is not documented in Table 2 . However, all the methods combining BERT embeddings (GloVe+BERT, GloVe+ELMo+BERT, GloVe+CSE+BERT, and GloVe+ELMo+CSE+BERT) resulted slightly lower performance than BioBERT. In addition, the models with BERT embeddings lead to vanishment of the effectiveness and motivation, namely construction of a lightweight model.

5.2 Evaluation on Processing Speed

Table 3 shows the extraction time of the proposed methods, SciBERT, and BioBERT for processing the test set of ChemProt[7]. The F1 score of the *Proposed3* was approximately 6 points lower than BioBERT. However, the extraction speed was 2.85 times faster than that of BioBERT. Although the F1 score of the *Proposed1* was approximately 8 points lower than the *Proposed3*, the extraction speed was 2.76 times faster than that of the *Proposed3*. We can see a trade-off between the F1 scores and the extraction speed. If users need the extraction speed for the application, our method is useful although the accuracy is comparatively sacrificed.

5.3 Evaluation on Memory Size

In the experiment, we used NVIDIA Tesla V100 GPU with 32GB memory. This is the high-end GPU for data center use at present. We observed the maximum memory consumption during the learning execution of BioBERT and the proposed method. BioBERT consumed approximately 12GB of memory. It indicates that BioBERT needs high-end GPUs to execute the learning. On the other hand, our method consumed approximately 2GB of memory. The memory consumption of our method was lower than that of BioBERT (1/6). In addition, considering the memory consumption, we believe that our model can be executed even on a middle-class GPU.

6 Discussion

We showed the effectiveness of our method in the previous section. In this section, we discuss our method from various perspectives, including some negative results. First, we discuss a comparison between the BERT-based models and the proposed method in Section 6.1. Then, we discuss the experimental settings of the proposed method in Section 6.2 and 6.3.

6.1 Datasets and Model Performance

For the GAD dataset, our method outperformed the BERT-based models. However, for the ChemProt dataset, the precision, recall, and F1 scores were lower than the scores of the BERT-based models. In the GAD dataset, the number of the negative samples is almost the same as the number of positive samples as shown in Table 2. On the other hand, in the ChemProt dataset, the number of negative samples is three times more than the number of positive samples. We analyzed the classification errors of our method for the positive samples of the ChemProt dataset. About 92% of the misclassifications were positive samples classified into the negative class, not into the other positive classes. It seems that the data imbalance affects our model

[7] We did not evaluate the extraction time for the GAD dataset because the dataset size is small. However, it seems that there is no large difference between the average time per sample for the GAD and the time for the ChemProt.

Corpus	F1 score
PubMed	69.57
PubMed+ChemRxiv	69.31
PubMed+PMC	70.19
PubMed+PMC+ChemRxiv	70.77

Table 4: Effects of the pre-training corpus. The score of PubMed+PMC+ChemRxiv is the same as the *Proposed3* in Table2.

performance negatively. On the other hand, the decrease in the performance of BioBERT was not as large as the decrease in that of our model. We need to analyze the cause of the difference between our methods and the BERT-based models.

The difference between the ChemProt dataset and the GAD dataset is not only the data imbalance but also the scale. The number of samples in the ChemProt dataset is almost eight times larger than the samples in the GAD dataset. We can use more data in the ChemProt dataset to train the models, as compared with the GAD dataset . In practice, large-scale learning data are, however, not always available. Therefore, it is effective to use different models according to the size of data. On the other hand, the relation between the size of the dataset and the suitable model is unobvious. Therefore, we need to analyze the relationship deeply for real and useful applications in biomedical domains.

6.2 Effects of Pre-Training Corpora

We used the CSE pre-trained on the PubMed, PMC, and ChemRxiv corpora in *Proposed3*. We evaluated the effectiveness of each corpus for the pre-training. Table 4 shows the results for the ChemProt. The F1 score of the model pre-trained on the PubMed + ChmeRxiv was lower than the model pre-trained on the PubMed. On the other hand, the F1 score of the model pre-trained on the PubMed + PMC was higher than the model pre-trained on the PubMed. The F1 score of the model pre-trained on the PubMed + PMC + ChemRxiv was the best score. The combination of PubMed, PMC, and ChemRxiv was effective although introducing ChemRxiv alone provided no benefit to the pre-training.

6.3 Effects of Target Entity Pair Indicators

We used the indicator tags to express a pair of two target entities. On the other hand, Lee et al. (Lee et al., 2019) replaced the target entity pair with pre-defined strings. In this section, we compare the methods using the indicator tags with the methods using the entity pair replacement. For the GAD dataset, we replaced diseases and genes with pre-defined strings @DISEASE$ and @GENE$, respectively. For the ChemProt dataset, Lim and Kang (2018), replaced chemicals and proteins with pre-defined strings "bc6entc" and "bc6entg", respectively. We used the same pre-defined strings that were used in Lim and Kang.

		Proposed 3 (GloVe+CSE+ELMo)	
Dataset		Tag	Replacement
GAD	P	**82.64**	81.88
	R	**86.36**	85.26
	F	**84.38**	83.46
ChemProt	P	**76.05**	75.38
	R	**66.17**	63.96
	F	**70.77**	69.20

Table 5: Effects of the target entity pair indicators.

For instance, ***1-aminoadamantane*** and ***Fos*** in **Example 4** are replaced with "***bc6entc***" and "***bc6entg***" respectively as shown in **Example 5**.

Example 4 *Amantadine (**1-aminoadamantane**) induced **Fos** expression in the central, dorsal-medial and ventral-medial part of the striatum.*

Example 5 *Amantadine (**bc6entc**) induced **bc6entg** expression in the central, dorsal-medial and ventral-medial part of the striatum.*

We evaluated the *proposed3* using each of the indicator tags and the entity pair replacement. Table 5 shows the experimental results for the GAD and ChemProt datasets. As a result, the use of the indicator tags was effective as compared with that of the replacement approach. We can use the surface information of the target entity pair by using indicator tags. Therefore, the result shows the effectiveness of the surface information of the entity pairs.

7 Conclusions

In this paper, we reported the effectiveness of lightweight and high-performance RE models for the biomedical domain. Our method used the combination of word embeddings generated by the pre-trained LMs (the ELMo model and the CSE model). The ELMo model is a *word-level* LM and the CSE model is a *character-level* LM. We proposed RE models that combined ELMo and CSE to utilize both word-level and character-level features effectively.

We evaluated the proposed methods on the biomedical RE datasets. We used the ChemProt dataset and the GAD dataset. We compared our methods with BERT-based methods (BioBERT and SciBERT) and the SOTA methods. We also evaluated the model performance and the inference time. Experimental results showed the effectiveness of the combinations of the LMs. For the GAD dataset, we obtained the SOTA score. For the ChemProt dataset, our model showed approximately three times faster extraction speed than BioBERT. In addition, our model reduced the memory size to one sixth of the BERT-based models. However, the F1 score of our method was lower than that of BioBERT for the ChemProt dataset. In future work, we analyze the causes of the high extraction speed and the low performance of our model for the ChemProt dataset in terms of the parameter size and architectures.

Acknowledgment

A part of this work was carried out in the RIKEN Center for Advanced Intelligence Project.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620.

Balu Bhasuran and Jeyakumar Natarajan. 2018. Automatic extraction of gene-disease associations from literature using joint ensemble learning. *PLOS ONE*, 13(7):1–22.

Léon Bottou. 1991. Stochastic gradient learning in neural networks. In *Proceedings of Neuro-Nımes*, page 91.

Álex Bravo, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I. Furlong. 2015. Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. *BMC Bioinformatics*, 16:55.

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.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Rezarta Islamaj Doğan, Sun Kim, Andrew Chatraryamontri, Chih-Hsuan Wei, Donald C Comeau, Rui Antunes, Sérgio Matos, Qingyu Chen, Aparna Elangovan, Nagesh C Panyam, Karin Verspoor, Hongfang Liu, Yanshan Wang, Zhuang Liu, Berna Altınel, Zehra Melce Hüsünbeyi, Arzucan Özgür, Aris Fergadis, Chen-Kai Wang, Hong-Jie Dai, Tung Tran, Ramakanth Kavuluru, Ling Luo, Albert Steppi, Jinfeng Zhang, Jinchan Qu, and Zhiyong Lu. 2017. Overview of the biocreative vi precision medicine track: mining protein interactions and mutations for precision medicine. In *the sixth BioCreative challenge evaluation workshop*, volume 1, pages 141–146.

Qiao Jin, Bhuwan Dhingra, William Cohen, and Xinghua Lu. 2019. Probing biomedical embeddings from language models. In *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pages 82–89.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Sangrak Lim and Jaewoo Kang. 2018. Chemical-gene relation extraction using recursive neural network. *Database*, 2018:1–11.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

Shreyas Sharma and Ron Daniel Jr. 2019. BioFLAIR: Pretrained Pooled Contextualized Embeddings for Biomedical Sequence Labeling Tasks. In *arXiv preprint arXiv:1908.05760.*

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650.

Taiki Watanabe, Akihiro Tamura, Takashi Ninomiya, Takuya Makino, and Tomoya Iwakura. 2019. Multitask learning for chemical named entity recognition with chemical compound paraphrasing. 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 6243–6248.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212.