General Embeddings Outperform Domain-Specific Embeddings for Biology Textbook Relation Extraction

Eli Pugh
Dept. of Mathematics
Stanford University
epugh@stanford.edu

Ethan Shen
Dept. of Computer Science
Stanford University
ezshen@stanford.edu

Cole Winstanley
Dept. of Symbolic Systems
Dept. of Electrical Engineering
Stanford University
colew@stanford.edu

Abstract

Here, we present a comparative evaluation of SciBERT (Beltagy et al., 2019) and BERT (Devlin et al., 2018) embeddings for relation extraction on the Life Biology textbook (Sadava et al., 2017) dataset, which was developed using distant supervision. This can used for building a knowledge base for biology students to query and augment their educational experience. For this task, we first built relation extraction models using pretrained embeddings which are fed into a stacked LSTM followed by a dense classifier on the benchmark TACRED (Zhang et al., 2017) dataset, and show that BERT outperforms SciBERT. We then compare BERT and SciBERT performance on the constructed LIFE dataset, and see that surprisingly, BERT outperforms SciBERT.

1 Introduction

Textbooks contain a vast wealth of information, but it is often difficult to find the answer to a specific question in the dense text. We adapt the natural language relation extraction task to solving this textbook query problem. In particular, we are building a system that takes in preprocessed sentences from a textbook and builds a knowledge base that can be queried with specific questions. In addition, this knowledge base will be able to point out the specific part of the text that corresponds to the answer to a question.

1.1 Task Definition

The task for our model is to build a knowledge base of relational data based on a large corpus of natural text. Each entry in the knowledge base will consist of a key-value pair, where the key is a pair of entities, and the value is the relation between them. For example, one entry might be:

```
(cell wall, cell membrane) => "encloses"
```

2 Background and Relevant Literature

2.1 Unsupervised Pre-Training: BERT

The introduction of new language representation models has led to a paradigm shift in NLP. Here, we provide a brief overview of these models and their potential application to our projects. Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) furthers this work by training a bidirectional transformer network. The key insight lies in its training procedure, which can be outlined in the following.

They first pre-training by random selection of 15% of the training corpus. They then replace 80% of these selected words with a masked token, 10% with a random word, and leave the other 10% as is. They then train the model to correct these, therefore learning about words from their surrounding context on each side. The second pre-training step is next sentence prediction. Given input sentences, BERT is trained to predict whether input 2 immediately follows input 1 in context. This allows BERT to learn about sentence relations, which is useful for extending to other NLP tasks.

BERT is a task-agnostic model that can be fine-tuned to surpass a wide range of benchmarks for NLU while requiring minimal changes to the model architecture.

2.2 Improving Relation Extraction by Pre-trained Language Representations

In recent years, state-of-the-art performance has been achieved by careful tuning of neural models which incorporate lexical and syntactic features, such as part-of-speech tags and dependency trees. Given the effectiveness of pre-training BERT encodings for various other NLP tasks, Alt et. al. proposes a natural question: can we leverage these pretrained models to further push the state of the art in relation extraction and semantic role labeling, without relying on lexical or syntactic features? Using this strategy, they introduce TRE, a Transformer for Relation Extraction, and achieve state-of-the-art results using a simple neural network layer on top of OpenAI GPT encodings.

In this paper, the authors do the following: (1) describe TRE, a Transformer based relation extraction model that, unlike previous methods, relies on deep language representations instead of explicit linguistic features, (2) show experiments outperforming state-of-the-art methods on the TACRED dataset, (3) report ablation studies showing that pre-trained language representations prevent overfitting and achieve better generalization in the presence of complex entity mentions. They also make the code open-source, which we will adapt for our project.

The TRE architecture is split into two modules – unsupervised pre-training of language representations and supervised fine-tuning on relation extraction. The pretraining step is described in (Radford et al., 2018), and reviewed above. Importantly, given a corpus $C = \{c_1, ..., c_n\}$ of tokens c_i, the language modeling objective maximizes the likelihood

$$L_1(C) = \sum_i \log P(c_i | c_{i-1}, ..., c_{i-k}; \theta)$$

and uses a weighted softmax function to predict the next word given a sequence of words. In the supervised module, the pre-trained embeddings from GPT is fed through another softmax weighted linear model to get the relation prediction to optimize the following objective:

$$L_2(D) = \sum_i^{|D|} \log P(r_i | x_i^1, ..., x_i^m)$$

where $D = ([x_i^1, ...x_i^m], a_i^1, a_i^2, r_i)$ is the labeled relation extraction dataset. The paper mentions that introducing language modeling as an auxiliary objective during fine-tuning improves generalization and leads to faster convergence, resulting in the following overall objective:

$$L(D) = \lambda * L_1(D) + L_2(D)$$

where λ is a scalar language model weight.

Beyond showing SOTA performance on TACRED and other datasets, the authors also show analysis and ablation experiments to demonstrate the contributions of different parts of their models to success. Comparing a model with pre-trained representations to a model with randomly initialized representations, they show that the TACRED validation F1 score decreases from 63.3 to 20. Furthermore, they show that entity masking has an important regularizing affect on model generalization. Briefly, entity masking limits information about entities in order to prevent overfitting. These strategies include replacing entity mentions with UNK tokens, named entity types (NE strategy), and/or grammatical roles (GR). Experimentally, they see that no entity masking gives the highest precision, though combining NE + GR masking strategies gives the highest Recall and F1 scores.

Overall, Alt. et. al. was the first work to apply language model pre-training to relation extraction, and demonstrate the effectiveness of this strategy.

2.3 SCIBERT: Pretrained Contextualized Embeddings for Scientific Text

While the above papers show success in leveraging unsupervised pretrained embeddings for NLP tasks, both BERT and GPT are still trained on general domain corpora such as news articles and Wikipedia. These general embeddings are therefore less effective on domain-specific tasks with specialized vocabularies, such as in scientific or medical domains. In this paper, Beltagy et. al. (Beltagy et al., 2019) release a BERT model pretrained on a large corpus of scientific text with code available at https://github.com/allenai/scibert. They build a new WordPiece vocabulary (Kudo and Richardson, 2018; Sennrich et al., 2015; Kudo, 2018), SCIVOCAB, with a token overlap with the baseline BERT vocabulary of 42%. SCIBERT outperforms BERT on biomedical tasks across different datasets, computer science tasks, and other scientific domains. These results address a major drawback of BERT embeddings, which will be useful given our domain-specific task.

3 Data

We will use two datasets when training and testing our models. The first is Text Analysis Conference Relation Extraction Dataset (TACRED) (Zhang et al., 2017), and the second is the Intelligent Life biology textbook (Sadava et al., 2017).

3.1 TACRED

TACRED is a widely-used relation extraction dataset compiled by Stanford researchers. Its built from the TAC Knowledge Base Population (TAC KBP) corpus which includes newswire and web text. TACRED has 106,264 examples spread over 41 relation types as well as the special no_relation type (Zhang et al., 2017).

The TACRED examples include the context sentence, subject and object spans, Stanford named entity recognition (NER) types, and the truth relation. (Zhang et al., 2017) The examples are well-annotated and the NER types give extra information not found in raw text that is often useful for predicting relation type.

TACRED was built to more closely mirror real-world situations than existing datasets. The average sentence length is 36.4, which is makes predicting relations more challenging when compared to datasets with shorter sentence length. In addition, 79.5% of TACRED examples are no_relation, which is higher than most similar datasets. Though this is still lower than real-world occurrence of no-relation, it more effectively penalizes models that have high false positive rates (Zhang et al., 2017).

Another advantage of the TACRED dataset is that Stanford researchers have also included benchmark performances of many standard models, such as logistic regression, patterns, CNNs, LSTMs, LSTMs with position aware attention, and other neural systems.

3.2 Intelligent Life

Intelligent Life (Sadava et al., 2017) is a biology textbook that is currently building a knowledge base (KB) for question answering as well as other tools like visualizing concept graphs. Currently only the first 10 chapters of the knowledge base have been human-verified. This includes 150,000 relation instances from around 20 relation types. While only the first 10 chapters have human-verified relations, the entire text is included, giving around 30,000 sentences of raw text. (Sadava et al., 2017)

Since Intelligent Life is a biology textbook, it has a very domain-specific vocabulary. This poses a challenge, since many models use word embedding methods that will have a large number of lookup misses. This means that any model that wishes to perform well on this dataset will need to use character-level embedding or be trained on a scientific corpus.

Another challenge that Intelligent Life presents is that only 10 chapters are human-labelled. This means that distant supervision will almost certainly improve accuracy, since there is very limited training data in only the first 10 chapters. This makes learning relations for the remaining chapters tricky when compared to a more simple supervised training approach used with TACRED.

3.2.1 Data Cleaning

In hopes of generating as much comparability as possible between the TACRED and Intelligent Life datasets, we performed some data cleaning on the Intelligent Life data. (The TACRED dataset is already very well-prepared for training.) We did the following before loading the data into the model:

- Removed sentences that have more than 80 tokens. This occurred in less than 1% of sentences, most of which were not usual natural language, such as long lists, or parts apparently extracted from tables in the text.

- Removed sentences for which the subject and object were labeled as overlapping. For instance, the sentence `The cell wall is stiff` does not contain an actual relation between `cell` and `cell wall`. This removed 35% of the examples.

- Removed examples where an entity was related to itself. While these could form real relations, they have a very high false positive rate because many sentences simply repeat their subject in multiple clauses.

3.3 Data Analysis

Here are some basic statistical differences between the TACRED and LIFE dataset:

	TACRED	LIFE
# unique relations	42	43
# sentences	106,264	49,137
avg sentence length	36.4	27.86
% no relation	79.5%	74.19%

Among where a relation was found, we plotted the distributions of relations in Figures 1,2. We also plotted the sentence length distributions in Figures 3,4, which show that the datasets are quite comparable.

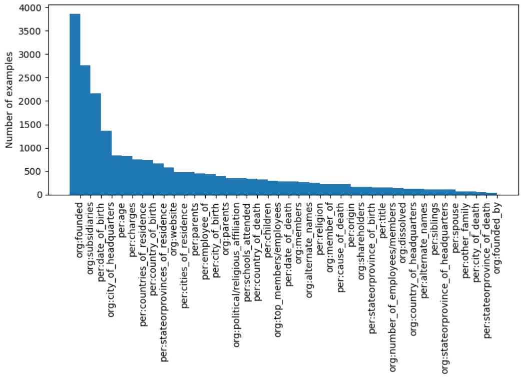


Figure 1: Relation label distributions for TACRED

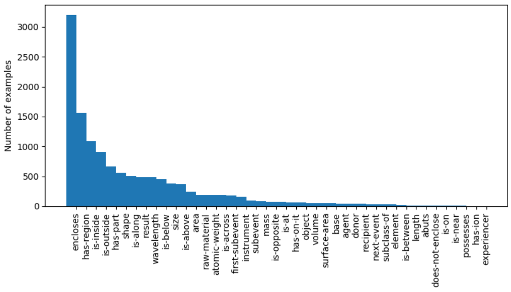


Figure 2: Relation label distributions for LIFE

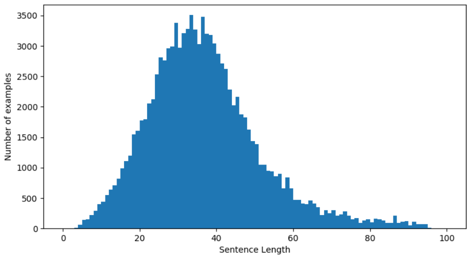


Figure 3: Sentence length distributions for TACRED

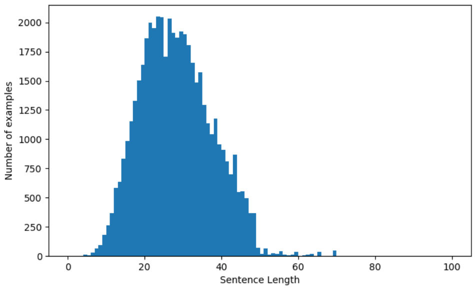


Figure 4: Sentence length distributions for LIFE

4 Methods

4.1 Tokenization for (Sci)BERT

A primary goal of this project was to compare BERT and SciBERT on domain-specific relation-extraction tasks, such as in the Life textbook. In order to utilize these embeddings, we needed to tokenize the sentences in both datasets so that the four word indices pointing to the start and end of both the subject and object are updated. This meant we needed to use the (Sci)BERT vocabulary files and a WordPiece tokenizer to first preprocess the data before feeding to our BERT model (Wu et al., 2016). By using WordPiece, many words were decomposed into multiple parts, so we changed the indices of the subject and object during this process as well. After this step, we added the subject and object at the end of the sentence, as well as `[CLS]` at the start and `[SEP]` between subject and object.

Example input sentence: ['Proteins', 'and', 'other', 'molecules', 'are', 'embedded', 'in', 'the', 'lipids.']

Tokenized sentence: ['[CLS]', 'Pro', '##tein', '##s', 'and', 'other', 'molecules', 'are', 'embedded', 'in', 'the', 'lip', '##ids', '.', '[SEP]', 'Pro', '##tein', '##s', '[SEP]', 'lip', '##ids', '[SEP]']

4.2 Baseline Models

We experimented with a few different models described in the TACRED dataset release, and then made adjustments to their highest performing model. TACRED compares results from many models including patterns, logistic regression, CNN, LSTM, and LSTM with position-aware attention (Zhang et al., 2017). We first tested their LSTM and LSTM with position-aware attention models that are publicly released.

The most performant model from the TACRED dataset tests is described in their paper, "Position-aware Attention and Supervised Data Improve Slot Filling" (Zhang et al., 2017). This model uses GloVe word embeddings (Pennington et al., 2014) as inputs $\mathbf{x}_1, \cdots, \mathbf{x}_n$. It then passes these through an LSTM model where the hidden states are $\mathbf{h}_1, \cdots, \mathbf{h}_n$, and the final state $\mathbf{q}$ is $\mathbf{h}_n$. The positional encodings $\mathbf{p_i^s}$ are

$$\mathbf{p}_i^s = \begin{cases} i - s_1, & i < s_1 \\ 0, & s_1 \leq i \leq s_2 \\ i - s_2, & i > s_2 \end{cases} \quad (1)$$

The attention weights a_i are calculated as

$$\begin{aligned} u_i &= \mathbf{v}^T \tanh(\mathbf{W}_h \mathbf{h}_i + \mathbf{W}_q \mathbf{q} + \mathbf{W}_s \mathbf{p}_i^s + \mathbf{W}_o \mathbf{p}_i^o) \\ a_i &= \frac{\exp(u_i)}{\sum_{j=1}^{n} \exp(u_j)} \end{aligned} \quad (2)$$

where the $\mathbf{W}$s and $\mathbf{v}$ are trainable parameters. The final sentence representation $\mathbf{z}$ is calculated as

$$\mathbf{z} = \sum_{i=1}^{n} a_i \mathbf{h}_i, \tag{3}$$

which is then fed into a fully-connected softmax classifier to predict which of the 41 relations (or no_relation) the example exhibits.

4.3 Our Model

In order to make an extension to this already impressive model, we utilized BERT (Devlin et al., 2018) and SciBERT (Beltagy et al., 2019) contextual embeddings as inputs $\mathbf{x}_1, \cdots, \mathbf{x}_n$ rather than GloVe embeddings. This idea stems from the work of Shi and Lin (Shi and Lin, 2019a), where they use BERT embeddings and a bidirectional LSTM to feed into a multi-layer perceptron classifier. We hoped these additions to the Zhang et al. positional attention model would greatly improve accuracy, since BERT embeddings achieve SOTA results on almost all NLP tasks and bidirectionality could improve understandings of relations where the subject comes after the object. In addition, one potential advantage of using BERT embeddings is their built-in positional encoding, so models using BERT embeddings seem to preserve more structure than embeddings without position or context.

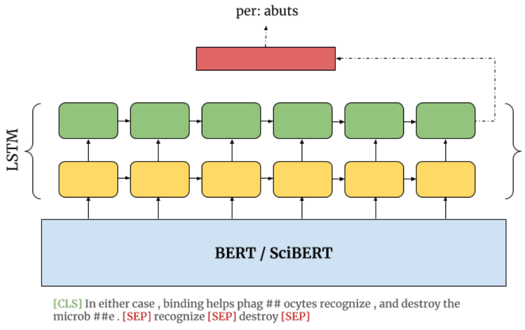

per: abuts
LSTM
BERT / SciBERT
[CLS] In either case , binding helps phag ## ocytes recognize , and destroy the microb ##e . [SEP] recognize [SEP] destroy [SEP]

Figure 5: Contextual-embedding LSTM model

During development, we were surprised to find that our model didn't see any notable performance gains from the position-aware attention layer or replacing the LSTM with a biLSTM. Instead, we allocated more time to data exploration, cleaning, and featurization, and removed the positional-attention layer and biLSTM in favor of a stacked LSTM. Our final model architecture was simple. We began with BERT and SciBERT featurization, and then fed these embeddings to a stacked LSTM. The final state was fed into a dense softmax classifier, which output the relation prediction. Our model is shown above in Figure 5.

4.4 Evaluation Metrics

We used mostly standard relation extraction metrics to evaluate how well our models perform the task. One challenge with relation extraction as a task is that recall is often difficult to define due to the distant supervision aspect; relationships that are not present in a knowledgebase or evaluation dataset are not necessarily actual negatives. So that we can address these issues, we will begin our evaluation of our model on our target dataset (the LIFE dataset) based on a confusion matrix, where no relation is a category in addition to the approximately 20 relations in the dataset. Our primary target will be the $F_{0.5}$ macro average over the 20 defined relations. This gives precision more weight than recall.

The focus on precision recognizes specific features of the task of building a question and answer system for a textbook. For a student trying to learn from the textbook, it is much more harmful to give a wrong answer to a question than giving no answer at all. Giving a wrong answer could deceive the student and therefore be deleterious on the effectiveness of the textbook. Using the macro average instead of the micro average recognizes that in general, it is more important for students to learn the relations themselves, instead of the instances of them. (e.g. it is more important to understand what symbiosis is than the fact that ants and acacia are an example of it.) Therefore, the macro average, which prioritizes performance on the relation level independent of the number of instances, correlates best to benefit for student users.

5 Hypothesis

- A simple LSTM model (Shi and Lin, 2019b) with BERT word embeddings will perform well on the LIFE dataset as it has on the TACRED dataset.

- Using SciBERT embeddings, potentially along with regularization tweaks, will beat BERT embeddings on LIFE dataset because BERT lacks scientific vocabulary. This will allow us to perform useful relation extraction for knowledge inference on LIFE dataset.

6 Experiments

6.1 Results

We performed our experiments using a single NVIDIA Telsa P100 GPU, implementing our models in PyTorch. We used `bert-as-service` (Xiao, 2018) to retrieve the BERT and SciBERT pretrained embeddings.

We first ran benchmark tests on the TACRED dataset to verify the quality of our data tokenization techniques and model architectures. Training our BERT and SciBERT models on the TACRED dataset resulted in a dev set $F_{0.5}$ score of 0.507 and 0.452

6d. We observed mild overfitting after 10 epochs for both models, which suggests that our results could be improved given more time for hyperparameter tuning and experimentation with regularization methods. Though the scores are lower than the published state-of-the-art BERT relation extraction score of 67.8 on the TACRED dataset (Shi and Lin, 2019b), our results gave us confidence in our model architecture to proceed.

We then sought to compare the performance of the BERT and SciBERT models on our LIFE relation extraction dataset. Our final test set results are below:

	TACRED	LIFE
BERT	0.4330	0.3350
SciBERT	0.5239	0.3204

6.2 Error Analysis

We experienced a few different sources of error, and some were much more pervasive than others. One particularly difficult source of error to remedy was the noise introduced by using distant supervision. There were many training examples that were labelled as relations, but the sentence didn't actually display the relation. For the examples that showed the relation, our model seemed to do well, as seen in the following example. The model correctly inferred that an allele is an object of a mutation.

Correct: [CLS] this mutation was in a cell that underwent me ##iosis to form gamet ##es , some of the resulting gamet ##es would carry the t allele , and some offspring of this pe ##a plant would carry the t allele . [SEP] mutation [SEP] allele [SEP] (relation: object)

In the following example, we see that the sentence doesn't actually show the relation that plasma is outside of cells, and our network has trouble with many of these examples.

Inorrect: [CLS] When a TH cell binds to the displayed antigen & # 82 ##11 ; MHC II complex , it releases cytokines that cause the B cell to produce a clone of plasma cells and memory cells (Figure 41 . 13 ##A) . [SEP] plasma [SEP] cell [SEP] (relation: is-outside)

The example is mis-labeled, and our model gets it wrong, so this detracts from accuracy but does not indicate a lack of performance in the model.

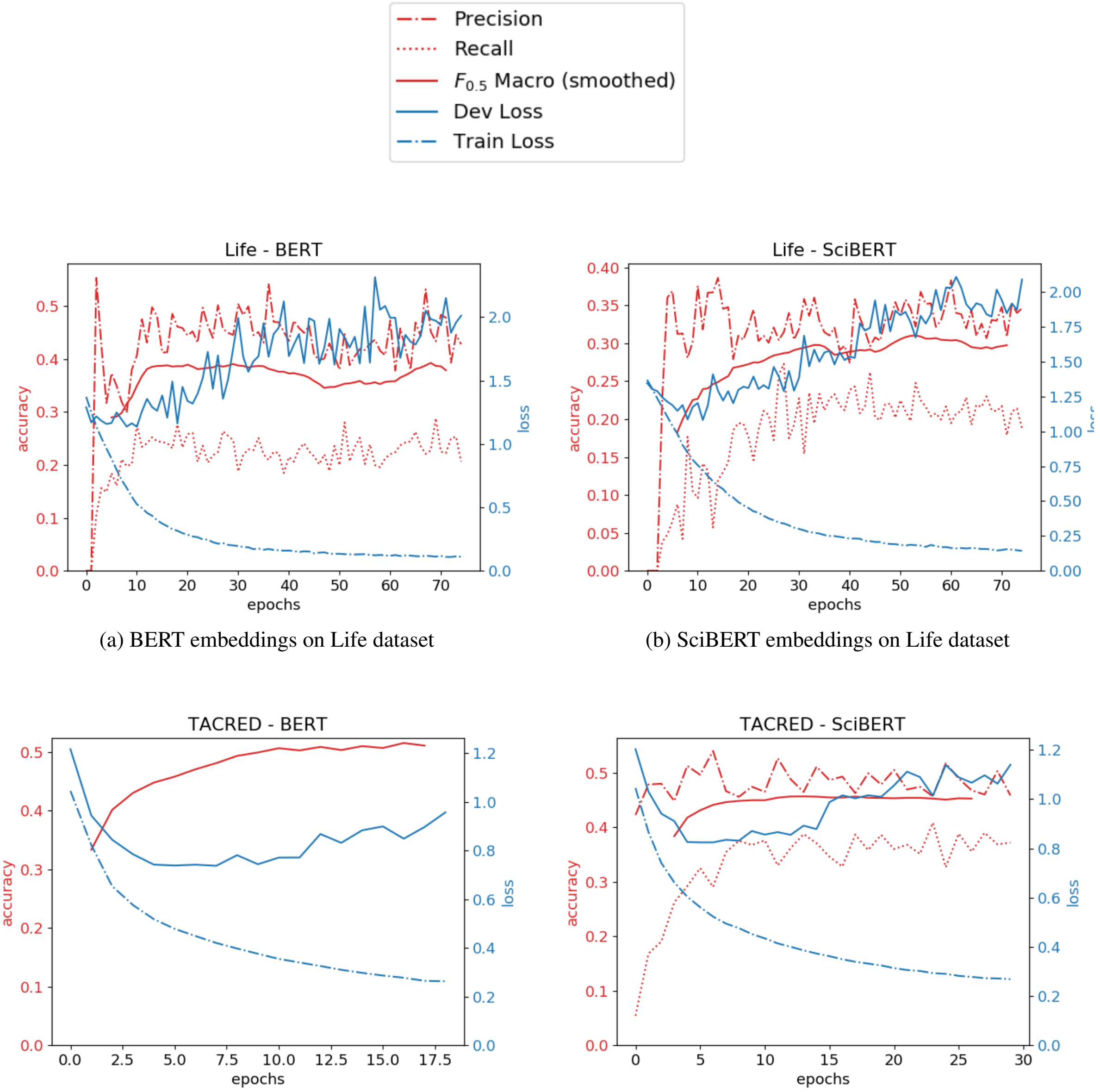
Another issue that we encountered was with BERT, and especially SciBERT, splitting up words that were not in the vocab, but probably should be. This is mostly due to the altered vocabulary of SciBERT, and can be seen in the splitting of the word `pea` in the correct example above.

In the following example, the source of error is most likely due to the LSTM not fully understanding the structure of the sentence. This may be due to the interpretation of `copy` as a verb instead of a noun.

Incorrect: [CLS] A) In one method of transpos ##ition , the DNA sequence is replicated and the copy inserts elsewhere in the genome . [SEP] copy [SEP] DNA [SEP]' (relation: object)

We attempted to improve our distant supervision methods to alleviate these errant predictions. As noted in the methods section, we attempted a few experiments for data cleaning: (1) removed sentences that have more than 80 tokens, (2) removed sentences for which the subject and object were labeled as overlapping, and (3) removed examples where an entity was related to itself. We then retrained our models and compared the previous results with our cleaned data. As seen in Figure 7, there is no significant difference.

Our hypothesis for why the BERT embeddings outperformed SciBERT mainly lies in the tokenizer, which we didn't adequately consider in the hypothesis-forming stage. We observed that a large proportion of the sentences with problems in the Life-SciBERT experiment had split tokenizations (i.e. out-of-vocabulary words that get split into subword embeddings), which we believe interfere with the model's interpretation of the structure of the sentences. This appears to be the case in the previous example with the word "copy." One useful insight that we can draw from this is that domain-specific vocabulary is not as critical for re-



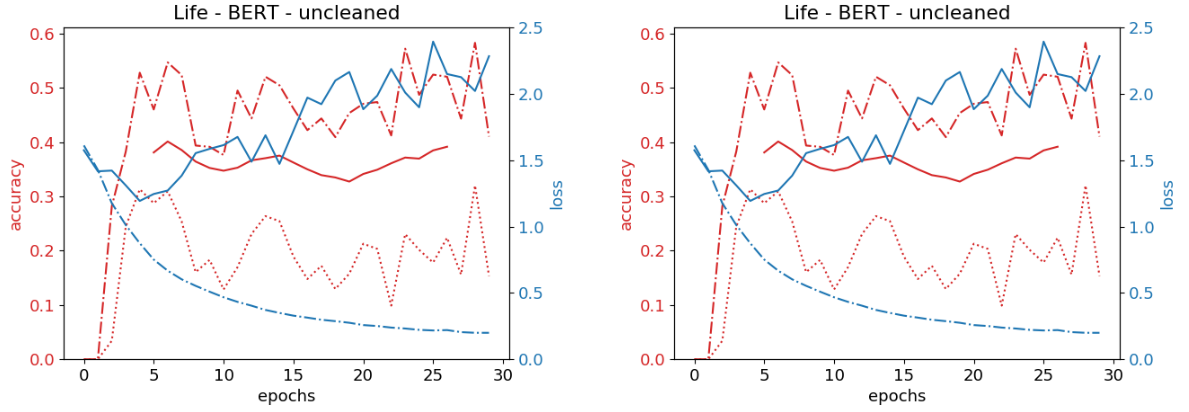
(a) BERT embeddings on Life dataset

(b) SciBERT embeddings on Life dataset

(c) BERT embeddings on TACRED dataset

(d) SciBERT embeddings on TACRED dataset

Figure 6: Relative performance of our final model using each combination of embeddings and dataset.



(a) BERT embeddings on TACRED dataset - no data cleaning

(b) SciBERT embeddings on TACRED dataset - no data cleaning

Figure 7: Performance of our model on the uncleaned datasets - notice similarity with the cleaned datasets.

lation extraction as a model that richly understands the grammatical structure of a variety of sentences.

6.3 Conclusion and Future Work

Though our work shows the limitations of domain-specific embeddings on our constructed LIFE dataset, further investigation is necessary to eliminate other possible sources of error. Though we confirmed our data preprocessing techniques and model architectures by training BERT and SciBERT on the TACRED dataset, we still observed significant overfitting, and scores below state-of-the-art though we were using a very similar model. Furthermore, to verify that our SciBERT data preprocessing method is correctly utilizing the domain-specific context embeddings, we would need to replicate the statistically significant improvements that SciBERT has over BERT on relation extraction tasks with the established ChemProt or SciERC datasets, which was published by Beltagy et. al. (Beltagy et al., 2019).

A deeper look at the quality of the examples generated by our distant supervision methods is also necessary. During our data cleaning process, we noticed that many of the examples were derived from a small number of relations, compared to the total number of constructed relations. Taking into account the success of SciBERT on the ChemProt and SciERC datasets, it might be elucidating to examine the example quality, vocabulary distribution, and relation statistics of the datasets where SciBERT shows significant improvement over BERT.

Finally, a significant potential area for improvement is to fine-tune the BERT and SciBERT embeddings on the new LIFE task. Overall, the validation of quality at each step would allow us to eliminate possible sources of error, and increase our scores on the LIFE dataset.

7 Authorship Statement

E.P., E.S and C.W. contributed to the literature review, experimental design, and paper drafting. E.P. and C.W. coded the model. E.S. performed data preprocessing experiments. We would like to acknowledge Ignacio Cases for guidance and moral encouragement, and Vinay Chaudhri for providing the dataset.

References

Iz Beltagy, Arman Cohan, and Kyle Lo. 2019. Scibert: Pretrained contextualized embeddings for scientific text. *CoRR*, abs/1903.10676.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. *CoRR*, abs/1804.10959.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226.

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.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. *Technical report, OpenAI*.

David E. Sadava, David M. Hillis, H Craig Heller, and Sally D. Hacker. 2017. *UnIntelligent Life: The Science of Biology*, volume 11.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *CoRR*, abs/1508.07909.

Peng Shi and Jimmy Lin. 2019a. Simple BERT models for relation extraction and semantic role labeling. *CoRR*, abs/1904.05255.

Peng Shi and Jimmy Lin. 2019b. Simple BERT models for relation extraction and semantic role labeling. *CoRR*, abs/1904.05255.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Han Xiao. 2018. bert-as-service. `https://github.com/hanxiao/bert-as-service`.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 35–45.