# NLP Relation Extraction in Biomedical Literature

Jacob Krucinski
Department of Computer Science, LTH
Lund, Sweden
ja6750kr-s@student.lu.se

***Abstract*—Every year, a large volume of research articles is being published to biomedical journals which are collected in the PubMed database with over 35 million articles. At this scale, thorough literature reviews and information extraction far exceeds human processing abilities. In order for researchers to fully utilize these grand collections of information, automated text mining is needed to extract relevant entities such as cells, diseases, chemicals, and genes/proteins, known as Named Entity Recognition (NER). Once entities have been identified, relation extraction (RE) can be performed to assemble knowledge triplets that can be transformed into knowledge graphs that support researchers in their aim to develop novel drugs and medical interventions, while also predicting and minimizing harmful side-effects. Prior research by my group, the Cell Death, Lysosomes and Artificial Intelligence group at Lund University, has recently developed a highly customizable Natural Language Processing (NLP) pipeline for NER. In this paper, I explore key models such as BioGPT and SciBERT for relation extraction using annotated corpora including BC5CDR, ChemProt, and DrugProt. The end goal was to train high-performing models and explore integration with the NER pipeline. The main tasks were BioGPT hyperparameter tuning, BioGPT free-form text RE, and SciBERT finetuning with expanded corpora. During hyperparameter tuning, the metrics were within $\pm$ 2% of the published metrics. The free-form text RE with BioGPT showed potential with suggestions for mechanisms and pathways for relations. Lastly, using a combined corpus for SciBERT finetuning increased the precision, recall, and F1 scores by 5%.**

CONTENTS

**I Introduction** 1

**II Background** 2
- II-A Corpora . . . 2
- II-B BioGPT . . . 2
- II-C SciBERT . . . 3

**III Methods** 3
- III-A Hyperparameter Tuning . . . 3
- III-B Free-Form Relation Extraction with BioGPT . . . 3
- III-C ChemProt and DrugProt Combined Corpus and SciBERT Fine-tuning . . . 4

**IV Results and Discussion** 4
- IV-A Hyperparameter Tuning . . . 4
- IV-B Free-Form Relation Extraction with BioGPT . . . 5
- IV-C ChemProt and DrugProt Combined Corpus and SciBERT Fine-tuning . . . 5

**V Conclusion** 9

**References** 9

**Biographies** 9
- Jacob Krucinski . . . 9
- V-A Bad NER Prompting for Free-form Text RE . . . 10
- V-B Full Confusion Matrices . . . 10
- V-C Oversampling Results . . . 16

978-1-7281-7436-5/21/$31.00 ©2023 European Union

## I. INTRODUCTION

Natural Language Processing (NLP) is an interdisciplinary field between linguistics and computer science in which human language is processed for various tasks including information extraction, sentiment analysis, text summarization, and machine translation [1]. The data is either in text or audio form, and it is common to use neural network techniques. As such, the data must be pre-processed into numeric form, which is done in two steps: tokenization and encoding/embedding. Tokenization takes text and splits it into chunks, which are either words, subwords or sentences. Then, encoding is used to transform tokens into vectors, which a neural network understands as input, and then can train on, and finally use for inference.

Relation Extraction is a sub-domain of NLP with the objective of predicting relations/attributes in a sentence, typically defined as a classification task [2]. It is used in the creation of relation knowledge graphs, where entities are nodes, and the relation/attributes are the edges. In the medical domain, such graphs can be used e.g. in drug design to formulate chemical composition and predict potential adverse side effects or in the analysis of cell signaling pathways [3].

The literature provides a variety of NLP model architectures which are backed up with performance metrics on common NLP tasks and corpora. However, when applied for a new downstream task with new corpora, performance is often not as good as expected. Rather than resorting to new models, specific architecture parameters, called hyperparameters, can be adjusted or "tuned" to optimize a specific performance metric of the model. Such hyperparameters are training-specific, such as the learning rate, or architecture-specific such as hidden layer sizes, number of attention heads, and dropout percentages. In this project (see section IV-A), I performed hyperparameter tuning on validation loss while training BioGPT [4].

The second focus of my project was to enable RE on free-form text, similar to the free-form capability of the EasyNER pipeline developed by my host group [5]. I pursued two approaches: using the text generation capability of BioGPT, and modifying the inference script. The reasoning behind the first approach is that text generation is a unique capability of a biomedical LLM, and the BioGPT contributors provided starter code for it. As for the second approach, it would recreate the same output format which could easily be used for evaluation with the pre-existing scripts.

Lastly, I continued the work of a previous student, Nils Broman [6], to determine if performance could be improved in a fine-tuned SciBERT [7] model when trained on two RE corpora combined: ChemProt [8] and DrugProt [9]. The reasoning behind fine-tuning, also known as "transfer learning", is that training a Large Language Model (LLM) from scratch can take days, even on top-of-the-line GPUs to get desired performance. Instead, fine-tuning acts makes use of a pre-trained model as a "stepping stone", using its weights which are already partially suited to the task at hand, and only require small updates. Nils' SciBERT models were trained on ChemProt only, or a combination of ChemProt and an artificial corpus he created. I modified his pre-processing scripts to process DrugProt instead, and then combined it with the existing processed ChemProt. The goal was to investigate whether more training data would increase the performance of the SciBERT model.

## II. Background

### A. Corpora

One widely-used corpus in the area of RE is the BC5CDR, BioCreative V Chemical Disease Relation corpus. It was created for the BioCreative V challenge in 2017, to promote research in biomedical text mining [CITE]. It contains 500 training and 500 test PubMed abstracts with annotations for chemicals and diseases, and the relations between them. The corpus is well-balanced, with approximately 5,000 chemical and 4,000 disease mentions, for a total of 1,000 relations in each dataset. The entities are labeled with their start and end indices in the combined title and abstract text with a space in between, along with their associated MESH IDs. These IDs are defined by the National Institute of Health, National Library of Medicine. The relations include a label for Chemical-Induced Disease (CID) and two MESH IDs for the related entities. An example is provided below:

```
439781|t|Indomethacin induced hypotension
in sodium and volume depleted rats.
439781|a|After a single oral dose of
4 mg/kg indomethacin (IDM) to sodium
and volume depleted rats plasma renin
activity (PRA) and systolic blood pressure
fell significantly within four hours.
In sodium repleted animals indomethacin
did not change systolic blood pressure
(BP) although plasma renin activity
was decreased. Thus, indomethacin by
inhibition of prostaglandin synthesis may
diminish the blood pressure maintaining
effect of the stimulated renin-angiotensin
system in sodium and volume depletion.
439781 0 12 Indomethacin Chemical D007213
439781 21 32 hypotension Disease D007022
...
439781 CID D007213 D007022
```

The ChemProt corpus is a collection of 1,820 PubMed abstracts with a total of 16,075 gold standard annotated interactions between chemicals and proteins. It was initially released in 2017 for the BioCreative VI track for text mining of chemical/protein relations. For each data subset: train, development/validation, test, and sample set, there is a file containing the abstracts, extracted entities with numbering and start/end indices, and the extracted relations. There are 22 relations groups, simplified into 10 ChemProt relation (CPR) classes by semantic similarity and overlapping biological properties. An example of the entity annotations is given below:

```
10076535 T10 CHEMICAL 1404 1419
alpha-estradiol
10076535 T11 CHEMICAL 1438 1452
beta-estradiol
```

With the TX entity labels defined, the relations are defined as:

```
10076535 CPR:2 N DIRECT-REGULATOR Arg1:T23
Arg2:T55
10076535 CPR:2 N DIRECT-REGULATOR Arg1:T2
Arg2:T48
```

The DrugProt corpus is similar to the ChemProt corpus, but instead focuses on interactions between drugs and proteins. It contains 4,250 PubMed abstracts with 21,035 annotations. The main difference was in the relation annotation formatting. There was no CPR code or evaluation label given, and instead just the relation group. Aside from the NOT group, all relations groups were the same as ChemProt CPR classes.

After merging the ChemProt and DrugProt corpus, there were a total of 23,710 training relations and 7,317 validation relations. This is a 3.7x increase in the training set size and 2.1x increase in the validation set size. It is important to note that there was no test or sample subset for this merged corpus, as the DrugProt corpus only has a usable train and validation set. This is because the test set was given without the relations file, as it was part of the BioCreative VII Track 1 challenge to score teams.

### B. BioGPT

The BioGPT model is a domain-specific Large Language Model (LLM) for biomedical text mining and generation. It is based on the GPT-2 medium model architecture, utilizing the same Transformer decoder. It has 24 layers, 16 attention heads, 1024 hidden size, for a total of 355M parameters [4]. GPT stands for Generative Pre Trained Transformers. Transformers were first introduced by Viswani et. al in 2017 [10]. Since then, they have dominated the world of NLP with their self-attention mechanism in which the model learns the importance of the

different tokens in the sequence. The self-attention mechanism, also known as scaled dot product attention, uses the following formula:

$$Attention(Q, K, V) = softmax\Big(\frac{QK^T}{\sqrt{d_k}}\Big)V \qquad (1)$$

$Q$ is a matrix of queries representing current tokens, $K$ is a matrix of keys for all the tokens, $V$ is the matrix with the corresponding values for each token, and $d_k$ is the size of the query and key vectors. The dot product $QK^T$ calculates the similarity between tokens of $Q$ and $K$ by producing a similarity score for each pair. Next, the scaling by $\sqrt{d_k}$ prevents steep gradients during training, which could negatively affect performance. Lastly, the softmax activation is applied to normalize the scores to sum to 1 and then used to weight the values in $V$ to compute the final output.

### C. SciBERT

SciBERT is branched off the BERT Large Language Model (LLM) and was specifically pre-trained on scientific literature in biomedicine. With this targeted pre-training, SciBERT is better able to understand unique terminology and concepts found in biomedical literature. BERT, which stands for Bidirectional Encoder Representation from Tranformers, is a state-of-the-art (SOTA) LLM which can be fine-tuned for many downstream tasks including relation extraction. It differs from BioGPT in the fact that it uses both encoder and decoder blocks. The full Transformer architecture is shown below:

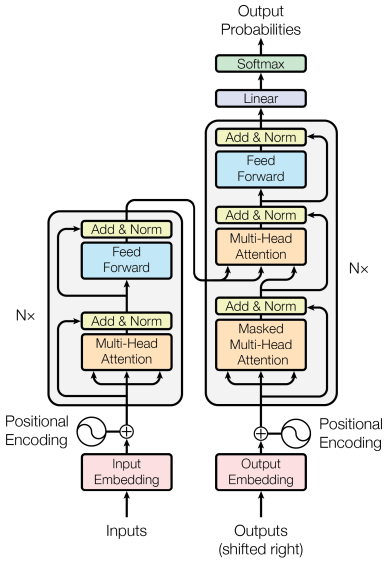


Fig. 1. Transformer Model Architecture. The model consists of two modules: an Encoder (left) and Decoder (right). At its core is the Multi-Head Attention block, which allows the model to attend to other parts of the sequence and learn context.

The main advantages of BERT are its bidirectionality, the capability of learning long-distance dependencies in text, and reduced computational overhead thanks to parallelization. The capability of bidirectionality comes from two unique approaches in its pre-training. Rather than using traditional next word prediction, BERT is pre-trained on Masked Language Modeling (MLM) and next sentence prediction (NSP). MLM entails the prediction of masked words, ensuring that the model learns context globally. As for NSP, BERT learns which is the next most likely sequence to come after a given sentence. The corpora for these pre-training tasks came from Wikipedia (2500M words) and unpublished books from BookCorpus (800M words) [6]. As for SciBERT, it is pretrained on a scientific domain: a subset of 1.14M papers from Semantic Scholars, with 82% of papers from the biomedical domain, and 18% from the computer science domain [7].

## III. Methods

### A. Hyperparameter Tuning

To perform hyperparameter tuning on the BioGPT model, I reviewed the documentation for the `fairseq-train` command in the fairseq library, which is a framework developed by the Facebook AI Research Group to simplify training various models in the NLP domain with common benchmarking corpora [11]. After reviewing the training details in other model publications including SciBERT and BioBERT, the hyperparameters I tuned were the learning rate and dropout. The learning rate controls how much the model weights are updated based on the gradients of the loss function. A larger step size allows the optimizer to converge on a local minimum faster, but if it is too large, then the optimizer will fail to find the minimum. Related to the learning rate, I also changed the Adam Beta values, which are decay rates for first and second moments of the gradient and control the smoothness and speed of convergence. Dropout is used to prevent overfitting by ignoring a percentage of outputs from a layer. This prevents the model from learning too heavily on certain features.

In the first experiment, I used dropout values of 0.1, Adam betas (0.9, 0.98), and a learning rate of 1e-5 with an inverse square root scheduler. In the second experiment, I increased the dropout to 0.2, changed the Adam betas to (0.9, 0.99), and a learning rate of 2e-5 (same scheduler). All other unchanged hyperparameters can be found in the `preprocess_train_vJacob.sh` script (see appendix section V).

### B. Free-Form Relation Extraction with BioGPT

One of the goals in my research was to explore whether SOTA RE models like BioGPT can be integrated into a NER pipeline developed by my host group. Aside from traditional dataset loading from pre-existing corpora, having the ability to perform RE on free-form text allows great flexibility and ease of use for the end-user. I built upon the existing BioGPT codebase [4] to perform this task.

I chose to explore two methods, one using end-to-end RE as used in the paper, and another using the unique capability of text generation. End-to-end RE means that model performs both NER and RE together, without any intermediate annotations. For the text-generation approach, I used two sub-approaches. The first one involved prompt engineering to first perform NER for chemicals and diseases, and then check for relations between each chemical and disease entity pair. The second sub-approach involved using a look-up to manually

extract entity pairs, and then setting up prompts in the "rel-is" format to get the relation existence and type. The "rel-is" type is defined in the BioGPT paper as "the relation between subject is rel.noun". For example, "the relation between dextropropoxyphene and mu-type opioid receptor is inhibitor". To benchmark both methods, I used the same 3 inputs: the first sentence from PubTator ID 439781 (in the test set of the BC5CDR corpus), a partial abstract from PubTator ID 23666265, and a custom sentence relating cyclosporine and depression (not found in any set of the corpus). Since text generation has no objective evaluation metric, example inputs and outputs are provided. As for the custom input, I chose a chemical and drug from the training entity set which did not have an abstract suggesting a relation between them (in either the train or test set). This way, the model would need to leverage its learned context, and not be mimicking input it had trained on.

### C. ChemProt and DrugProt Combined Corpus and SciBERT Fine-tuning

In order to merge the ChemProt and DrugProt corpus, I started by modifying scripts developed by a previous student of the group, Nils Broman. The `extract_relations.py` script to use the correct directories where I stored the downloaded DrugProt corpus and save different formatted data fields. The DrugProt relations file did not have the CPR code and GENE-Y/GENE-N columns as ChemProt did. I used this script on both the training and development/validation sets. The DrugProt corpus also contained a "test+background" subset, but it had no relation data and thus could not be used for training and evaluation. Next, I modified Nils' `add_custom_labels.py` script to use the new directories and preprocess the DrugProt corpus. This script maps the ChemProt/ DrugProt CPR codes to the following 5 classes: INTERACTOR, NOT, PART-OF, REGULATOR-NEGATIVE, and REGULATOR-POSTIVE. With pre-processing done, I merged the ChemProt and DrugProt training and development sets using the `cat` command in Linux with the following format: `cat CHEMPROT_FILE DRUGPROT_FILE > MERGED_FILE`.

For training and evaluation, I re-used Nils' pipeline with the finetune script, evaluation, and plotting Jupyter notebook. This way, the results between my finetuned model and his model can easily be compared. All hyperparameters were left unchanged, except for the number of epochs, which was set to 10. When calculating the precision, recall, and F1, 3 averaging types were used: macro, micro, and weighted. For macro averaging, the metrics for each class are averaged, whereas for micro averaging, all samples from all classes contribute equally. Weighted averaging is similar to macro averaging but adds a weight to each metric depending on the number of samples in each class. Due to the class imbalance for the combined corpus (see table I), an oversampling strategy of 10x, 4x, and 3x for the NOT, PART-OF, and REGULATOR-POSITIVE classes, respectively.

All model training was performed on an NVIDIA DGX A100 40 GB compute node on the Berzelius cluster at Linköping University [12].

TABLE I
CLASS DISTRIBUTION OF THE COMBINED CHEMPROT AND DRUGPROT TRAINING SET. A SIMILAR DISTRIBUTION IS SEEN FOR THE VALIDATION SET.

| Class | Count | Percentage |
|---|---|---|
| REGULATOR-NEGATIVE | 10207 | 43.04% |
| INTERACTOR | 8435 | 35.58% |
| REGULATOR-POSITIVE | 3634 | 15.33% |
| PART-OF | 1193 | 5.03% |
| NOT* | 241 | 1.02% |

## IV. RESULTS AND DISCUSSION

### A. Hyperparameter Tuning

For the hyperparameter tuning exploration, Figure 2 shows the training and validation loss curves for the first experiment with dropout values of 0.1, Adam betas (0.9, 0.98), and a learning rate of 1e-5 with an inverse square root scheduler training for 50 epochs. The training loss started at 4.734 and continued decreasing to 1.716. At epoch 22, the minimum validation loss of 2.811 was obtained and continued increasing for the rest of the epochs to 2.985. This suggests that the model is overfitting at later epochs, i.e. it is learning the training data too well, and thus lacking the capability to generalize on new data. This suggested that 30 epochs is a suitable training period.

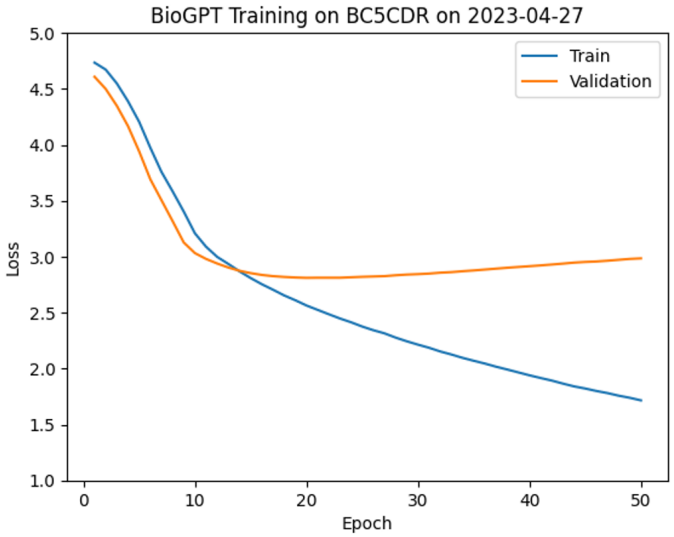


Fig. 2. Train and validation loss curves from hyperparameter tuning experiment 1. BioGPT was trained on the BC5CDR training set for 50 epochs. For the model, the dropout was set to 0.1. The Adam optimizer was used with beta values (0.9, 0.98), and an initial learning rate of 1e-5 with an inverse square root schedule.

Figure 3 below shows the training and validation loss curves for the second experiment with dropout values of 0.2, Adam betas (0.9, 0.99), and a learning rate of 2e-5 with the same scheduler training for 50 epochs. The minimum validation loss is reached in an earlier epoch than in the first experiment, with a minimum of 2.827 at epoch 14. Again, it continued increasing for the rest of the epochs to 3.130,

which is 0.145 higher than the first experiment. It seems that the dropout did not affect the overfitting, but the altered learning rate parameters allowed the model to converge faster in the beginning of training.

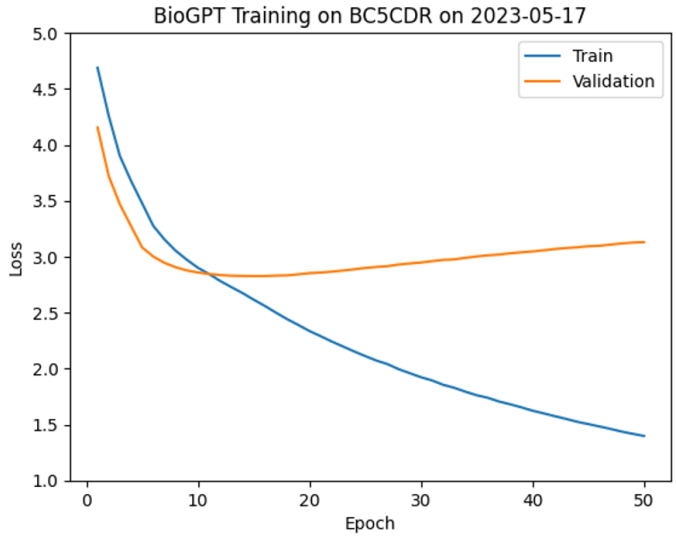


Fig. 3. Train and validation loss curves from hyperparameter tuning experiment 2. BioGPT was trained on the BC5CDR training set for 50 epochs. For the model, the dropout was set to 0.2. The Adam optimizer was used with beta values (0.9, 0.99), and an initial learning rate of 2e-5 with an inverse square root schedule.

For these graphs, I used an averaged model checkpoint calculated from the last 5 epochs in training, as per the original inference script. With brief experimentation, the performance metrics were better than using the last or best checkpoint. This method likely attenuates weights in an overfitted model.

For more standardized metrics, the precision, recall, and F1 scores, along with the performance decrease/increase were compared to the original publication.

TABLE II
PRECISION, RECALL, AND F1-SCORE PERFORMANCE COMPARISON OF THE TWO HYPERPARAMETERS EXPERIMENTS TO THE ORIGINAL BIOGPT PAPER. CHANGES COMPARED TO THE ORIGINAL PAPER ARE SHOWN IN THE PARENTHESES. THE MODEL SHOWN FOR EXPERIMENT 1 AND 2 IS THE AVERAGE MODEL CHECKPOINT FROM THE LAST 5 TRAINING EPOCHS. ALL 3 MODELS WERE TRAINED AND EVALUATED ON THE BC5CDR CORPUS TRAINING SET FOR RELATION EXTRACTION.

| Model | Paper | Experiment 1 | Experiment 2 |
|---|---|---|---|
| Precision | 0.4944 | 0.4872 (-0.0072) | 0.5176 (+0.0232) |
| Recall | 0.4128 | 0.4118 (-0.0010) | 0.4005 (-0.0123) |
| F1 | 0.4498 | 0.4464 (-0.0034) | 0.4516 (+0.0018) |

### B. Free-Form Relation Extraction with BioGPT

After hyperparameter tuning the BioGPT model, it was used for free-form text end-to-end RE, the same way the BioGPT authors performed inferencing [4]. For comparison, I passed the same free-form text input into the fine-tuned model from experiment 2 and into a model using the checkpoint provided by the BioGPT authors. Results can be found in the table in Figure 4. With a sentence from the PubTator abstract 439781 (part of the BC5CDR test set), the model I trained failed to find any entities. On the other hand, the model created from the published BioGPT checkpoint did find one chemical, sodium, but failed to relate it to a disease. With the PubTator 23666265 full abstract, both models were only able to extract the first relation between cyclophosphamide (CYP) and cystitis, but not pain and edema. With a custom input sentence, both models successfully found the relation between cyclosporine and depression.

The first sub-approach with prompt engineering to perform NER did not work well (see Appendix section V-A). A reason for this could be that the pre-trained BioGPT model was not trained on those kinds of prompts, but rather on describing entities. Therefore, it tried to explain the found entities, rather than listing the entities one after another.

The second sub-approach via "rel-is" prompting proved more promising. The entity recognition worked quite well, although the generated text was not very clear. Instead, as seen in Figure 5) below, the generated text said the relation was "not clear" or "not well understood" despite clear correlation in the input text. Interestingly, the remaining generated text actually provided biological mechanisms and pathways connected to the chemical-disease relation (see the red text).

### C. ChemProt and DrugProt Combined Corpus and SciBERT Fine-tuning

As inference with the BioGPT model proved difficult, I switched to a SciBERT model trained on the combined ChemProt and DrugProt corpus. The same model had been trained with only the ChemProt corpus by Nils Broman, which I defined to be the baseline model. To directly compare the performance metrics, I re-used the evaluation script to plot training loss & accuracy, and epoch- level precision, recall, and F1-scores. Also, multi-class confusion matrices were generated to show prediction accuracy for each of the 5 classes. As Nils' model checkpoint was not available for evaluation, and thus his model metrics could not be plotted together with mine. As a result, there is a difference in y-axis scale on the figures and the epoch difference on the x- axis. In addition, due to overfitting, I focus on the validation metrics to represent the true performance of the models.

In terms of training and validation loss, my model (Figure 6) consistently lower values than Nils' model (Figure 7). At the end of epoch 1, my validation loss was 0.28, and his was 0.48. With continued training, the training loss continually decreases, but the validation loss increases. This is a result of overfitting.

The peak macro, micro, and weighted validation precision scores for the combined corpus model were 0.911, 0.938, and 0.938, respectively (see Figure 8). For Nils' model, the peak macro, micro, and weighted precision were 0.885, 0.881, and 0.880, respectively (see Figure 9). Therefore, my model had a 4.7% average increase in precision across all sub-scores when compared to the baseline model.

| Source | Free-form Text and Output | True Entities | True Relations |
|---|---|---|---|
| PubTator ID 439781 | IN:<br>After a single oral dose of 4 mg/kg indomethacin (IDM) to sodium and volume depleted rats plasma renin activity (PRA) and systolic blood pressure fell significantly within four hours, suggesting hypotension.<br><br>OUT (with my modWhatel):<br>227508 CID -1 -1 1.0<br><br>OUT (with paper model):<br>227508 CID D012964 -1 1.0<br>(sodium) | C: Indomethacin<br>D: hypotension | Indomethacin/hypotension |
| Pubtator 23666265 (partial) | IN:<br>The purpose of the study is to explore the function of P2X3 and NK1 receptors antagonists on cyclophosphamide (CYP)-induced cystitis in rats. METHODS: Sixty female Sprague-Dawley (SD) rats were randomly divided into three groups. The rats in the control group were intraperitoneally (i.p.) injected with 0.9% saline (4 ml/kg); the rats in the model group were i.p. injected with CYP (150 mg/kg); and the rats in the intervention group were i.p. injected with CYP with subsequently perfusion of bladder with P2X3 and NK1 receptors' antagonists, Suramin and GR 82334. Spontaneous pain behaviors following the administration of CYP were observed. Urodynamic parameters, bladder pressure-volume curve, maximum voiding pressure (MVP), and maximum cystometric capacity (MCC), were recorded. Pathological changes in bladder tissue were observed. Immunofluorescence was used to detect the expression of P2X3 and NK1 receptors in bladder. MCC in the model group was lower than the control and intervention groups ($P < 0.01$). Histological changes evident in model and intervention groups rats' bladder included edema, vasodilation, and infiltration of inflammatory cells.<br><br>OUT (same for my model and paper model):<br>227508 CID D003520 D003556 1.0<br>(CYP <--> cystitis) | C: CYP<br>D: Cystitis, pain, edema | CYP/cystitis<br>CYP/pain<br>CYP/edema |
| Custom | IN:<br>A patient perscribed 100 mg/ml cyclosporine suffered from high blood pressure, negative thoughts, and depression.<br><br>OUT (same for my model and paper model):<br>227508 CID D016572 D006973 1.0<br>(cyclosporine <--> hypertension) | C: Cyclosporin<br>D: Depression | Cyclosporin/depression |

Fig. 4. Free-form text end-to-end RE results using both the averaged model checkpoint from experiment 2 and the model checkpoint from the BioGPT paper. Both models either failed to find the chemical and disease entities and their relation, or only found one of the multiple relations.

| Source | Free-form Text and Output | True Entities | True Relations |
|---|---|---|---|
| PubTator ID 439781 | IN:<br>After a single oral dose of 4 mg/kg indomethacin (IDM) to sodium and volume depleted rats plasma renin activity (PRA) and systolic blood pressure fell significantly within four hours, suggesting hypotension.<br><br>OUT:<br>The relation between indomethacin and hypotension is not clear, but it may be due to a direct effect of indomethacin on the vascular smooth muscle or to an effect of indomethacin on the renin-angiotensin-aldosterone system, which is known to be involved in the pathogenesis of indomethacin-induced hypotension in patients with chronic renal failure (CRF) and in patients with normal renal function (NRF). (ABSTRACT TRUNCATED AT 250 WORDS) The present study was designed to investigate the effect of indomethacin on the renin-angiotensin-aldosterone system in patients with CRF and in patients with NRF.<br><br>The relation between sodium and hypotension is not well understood, but it has been suggested that sodium may play a role in the pathogenesis of hypotension in patients with renal insufficiency and in those with congestive heart failure (CHF), both of which are associated with increased sympathetic nervous system (SNS) activity and increased plasma norepinephrine (NE) levels, and that sodium depletion may be a factor in the development of hypotension in these patients. (ABSTRACT TRUNCATED AT 250 WORDS) The purpose of this study was to investigate the relation between sodium and hypotension in patients with renal insufficiency and in those with CHF. | C: Indomethacin<br>D: hypotension | Indomethacin/hypotension |
| Pubtator 23666265 (partial) | IN:<br>The purpose of the study is to explore the function of P2X3 and NK1 receptors antagonists on cyclophosphamide (CYP)-induced cystitis in rats. METHODS: Sixty female Sprague-Dawley (SD) rats were randomly divided into three groups. The rats in the control group were intraperitoneally (i.p.) injected with 0.9% saline (4 ml/kg); the rats in the model group were i.p. injected with CYP (150 mg/kg); and the rats in the intervention group were i.p. injected with CYP with subsequently perfusion of bladder with P2X3 and NK1 receptors' antagonists, Suramin and GR 82334. Spontaneous pain behaviors following the administration of CYP were observed. Urodynamic parameters, bladder pressure-volume curve, maximum voiding pressure (MVP), and maximum cystometric capacity (MCC), were recorded. Pathological changes in bladder tissue were observed. Immunofluorescence was used to detect the expression of P2X3 and NK1 receptors in bladder. MCC in the model group was lower than the control and intervention groups ($P < 0.01$). Histological changes evident in model and intervention groups rats' bladder included edema, vasodilation, and infiltration of inflammatory cells.<br><br>OUT:<br>The relation between cyclophosphamide and edema is discussed in the light of the results obtained in the present study and those reported in the literature, as well as in the light of the results obtained in the present study and those reported in the literature concerning the effect of cyclophosphamide on the permeability of the blood-brain barrier and on the permeability of the blood-cerebrospinal fluid barrier...<br><br>The relation between cyclophosphamide and pain is complex and may be explained by a number of factors, including the effects of the drug on the central nervous system, the effects of the drug on the immune system, the effects of the drug on the central nervous system, and the effects of the drug on the peripheral nervous system...<br><br>The relation between cyclophosphamide and cystitis is not well understood, but it is thought to be due to a direct effect of the drug on the urothelium and / or to an indirect effect of the drug on the autonomic nervous system and / or the urothelium itself, or to a combination of these two factors... | C: CYP<br>D: Cystitis, pain, edema | CYP/cystitis<br>CYP/pain<br>CYP/edema |
| Custom | IN:<br>A patient perscribed 100 mg/ml cyclosporine suffered from high blood pressure, negative thoughts, and depression.<br><br>OUT:<br>The relation between cyclosporine and depression is complex and may be influenced by several factors, including the dose of cyclosporine, the duration of cyclosporine therapy, and the presence of concomitant medications, such as lithium, which may affect the pharmacokinetics of cyclosporine and the development of cyclosporine-induced depression. (ABSTRACT TRUNCATED AT 250 WORDS) Cyclosporine-induced depression is a serious and potentially life-threatening side effect of cyclosporine therapy, and it is important to identify patients at risk for the development of cyclosporine-induced depression so that appropriate interventions can be instituted. | C: Cyclosporin<br>D: Depression | Cyclosporin/depression |

Fig. 5. Free-form text RE via "rel-is" prompting results using the averaged model checkpoint from experiment 2. All entities were detected via the lookup-table NER method and the generated text suggested pathways triggered by the chemical, which in turn cause the disease.

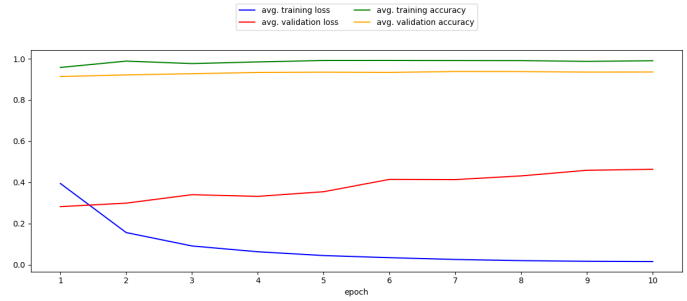


Fig. 6. Train and validation loss and accuracy curves from fine-tuning the SciBERT model on the combined ChemProt and DrugProt corpus for 10 epochs.

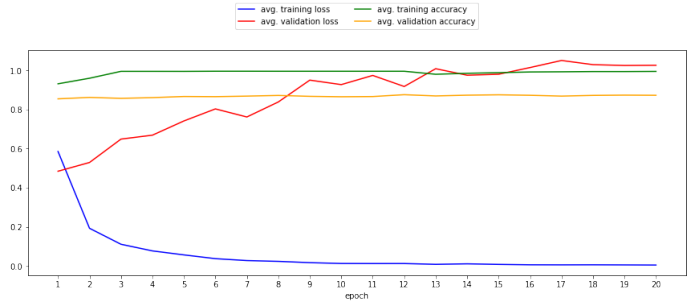


Fig. 7. Train and validation loss and accuracy curves from fine-tuning the SciBERT model on only the ChemProt corpus for 20 epochs. This figure was copied from a prior student project for comparison.

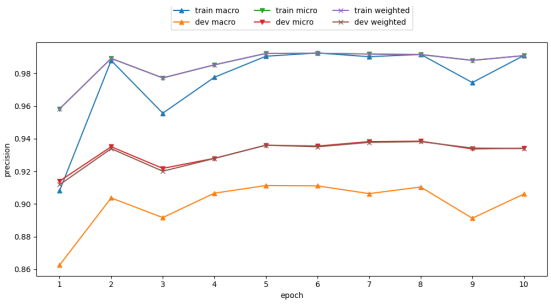


Fig. 8. Macro, micro, and weighted precision scores for each epoch when fine-tuning the SciBERT model on the combined ChemProt and DrugProt corpus for 10 epochs.

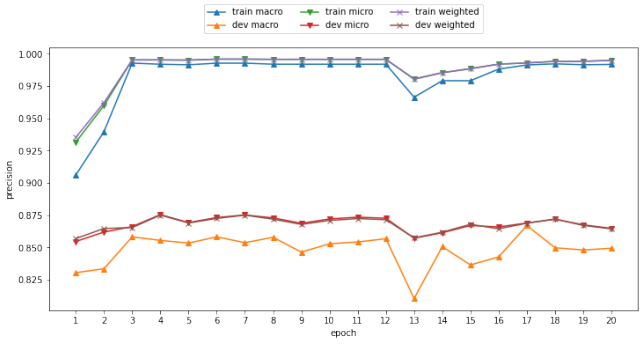


Fig. 9. Macro, micro, and weighted precision scores for each epoch when fine-tuning the SciBERT model on only the ChemProt corpus for 20 epochs. These results were copied from a prior student project report for comparison.

For recall, there was a similar improvement in performance. The peak macro, micro, and weighted validation recall scores of the combined corpus model were 0.882, 0.939, and 0.939 respectively (see Figure 10). For Nils' model, the peak macro, micro, and weighted recall scores were 0.857, 0.881, and 0.881 respectively (see Figure 11). Therefore, my model had a 4.7% average increase in precision across all sub-scores when compared to the baseline model.

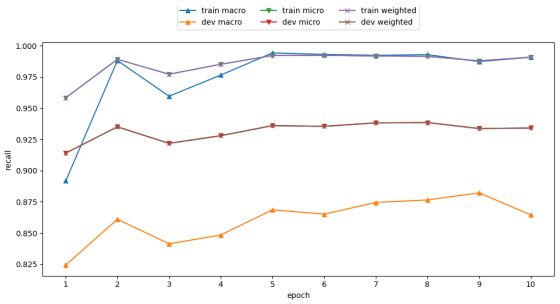


Fig. 10. Macro, micro, and weight recall scores for each epoch when fine-tuning the SciBERT model on the combined ChemProt and DrugProt corpus for 10 epochs.

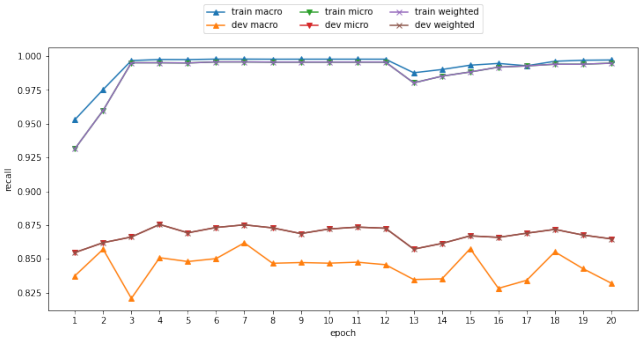


Fig. 11. Macro, micro, and weight recall scores for each epoch when fine-tuning the SciBERT model on only the ChemProt corpus for 20 epochs. These results were copied from a prior student project report for comparison.

The F1 score, which is derived from precision and recall, was consequently also improved in my model. The peak macro, micro, and weighted validation recall scores of the combined corpus model were 0.890, 0.938, and 0.938

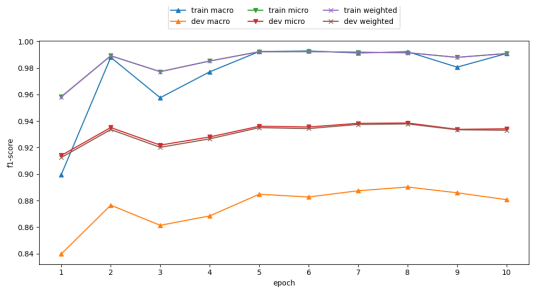


Fig. 12. Macro, micro, and weight F1 scores for each epoch when fine-tuning the SciBERT model on the combined ChemProt and DrugProt corpus for 10 epochs.

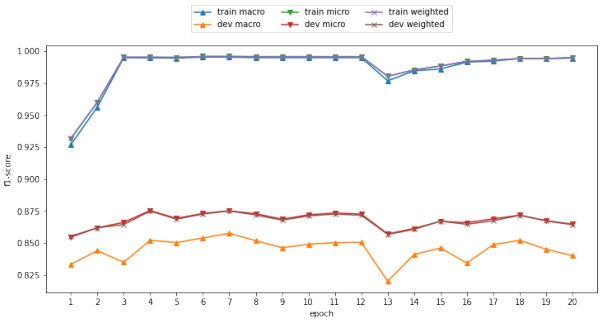


Fig. 13. Macro, micro, and weight F1 scores for each epoch when fine-tuning the SciBERT model on only the ChemProt corpus for 20 epochs. These results were copied from a prior student project report for comparison.

respectively (see Figure 12). For Nils' model, the peak macro, micro, and weighted recall scores are 0.861, 0.881, 0.880 respectively (see Figure 13). Therefore, my model had a 4.8% average increase in precision across all sub-scores when compared to the baseline model.

Lastly, I examined the multi-class confusion matrices for the majority best epoch only based on the macro, micro, and weighted metrics (Figures 14-17, all other epoch confusion matrices can be found in the Appendix section V-B). Since the best performance metrics may not have come from the same epoch, the most frequent epoch was the majority best epoch. The labels from top to bottom on the y-axis and left to right on the x-axis are: INTERACTOR, NOT, PART-OF, REGULATOR-NEGATIVE, and REGULATOR-POSTIVE.

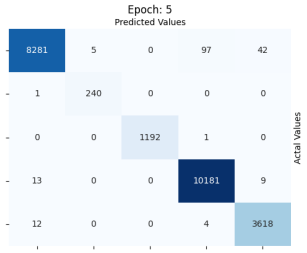


Fig. 14. Confusion matrix for the training set at epoch 5, the majority best epoch, of the SciBERT model trained on the combined ChemProt and DrugProt corpus

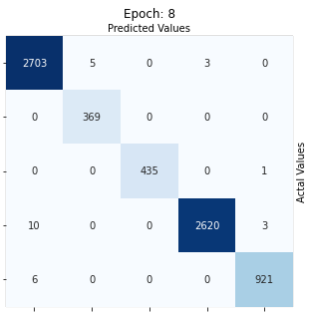


Fig. 15. Confusion matrix for the training set at epoch 8, the majority best epoch, of the SciBERT model trained on only the ChemProt corpus.

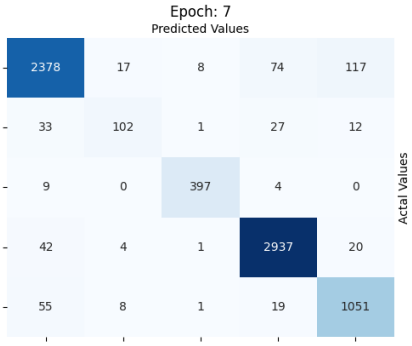


Fig. 16. Confusion matrix for the development set at epoch 7, the majority best epoch, of the SciBERT model trained on the combined ChemProt and DrugProt corpus.

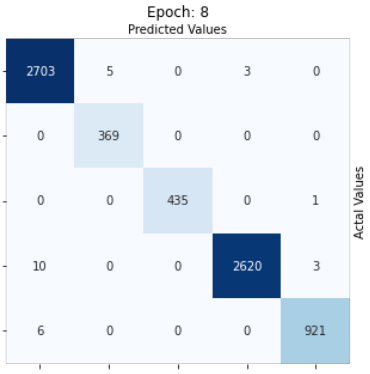


Fig. 17. Confusion matrix for the development set at epoch 31 (not shown in loss curves), the majority best epoch, of the SciBERT model trained on only the ChemProt corpus.

All of these performance metrics have been summarized in Table III

TABLE III
MACRO, MICRO, AND WEIGHTED PRECISION, RECALL AND F1 SCORES FOR THE SCIBERT MODEL TRAINED ON THE COMBINED CHEMPROT AND DRUGPROT CORPUS AND THE BASELINE MODEL TRAINED ON THE CHEMPROT ONLY CORPUS TRAINED FOR AN EXTENDED 40 EPOCHS (METRICS FROM 20 EPOCH TRAINING WERE NOT AVAILABLE).

| | **Combined Model** | | | **Baseline Model** | |
|---|---|---|---|---|---|
| **Metric** | Train | Dev | Dev change | Train | Dev |
| **Precision** | | | | | |
| macro | 0.9925 | 0.9113 | +0.0266 | 0.9955 | 0.8847 |
| micro | 0.9924 | 0.9385 | +0.0577 | 0.9958 | 0.8808 |
| weighted | 0.9924 | 0.9382 | +0.0584 | 0.9958 | 0.8798 |
| **Recall** | | | | | |
| macro | 0.9943 | 0.8822 | +0.0252 | 0.9973 | 0.8570 |
| micro | 0.9924 | 0.9385 | +0.0577 | 0.9958 | 0.8808 |
| weighted | 0.9924 | 0.9385 | +0.0577 | 0.9958 | 0.8808 |
| **F1-score** | | | | | |
| macro | 0.9928 | 0.8902 | +0.0292 | 0.9952 | 0.8610 |
| micro | 0.9924 | 0.9385 | +0.0577 | 0.9958 | 0.8808 |
| weighted | 0.9924 | 0.9382 | +0.0583 | 0.9958 | 0.8799 |

The same evaluation of loss curves, epoch precision/recall/F1, confusion matrices, and performance summary were repeated on the oversampled version of the combined corpus. Results can be found in Appendix Section V-C. As seen in Table IV, metrics were within +/- 1%.

## V. Conclusion

In this paper, I explored three biomedical relations corpora, BC5CDR, ChemProt, and DrugProt, and two scientific LLMs, BioGPT and SciBERT for relation extraction. After hyperparameter tuning on the BioGPT model I was able to closely match the performance given in the original paper, and improve precision by 2%. Free-form relation extraction via BioGPT text generation showed potential, as the generated text suggested effects on systems of the body and how that could contribute to the disease. SciBERT finetuning on a combined ChemProt and DrugProt corpus improved precision, recall, and F1 scores by 5% for all three metrics.

Future work for hyperparameter tuning should involve experimenting with learning rate schedulers, Adam Betas, and max token length and source/target positions. For the free-form text RE with text generation via NER, manual NER could be done using the combined train and test entities, or using the Aits Lab NER pipeline. Prompt engineering could also be explored, but due to the nature of the pre-trained text generation model, this will likely have little improvement. Lastly, with the merged ChemProt and DrugProt corpus, other SOTA models could be fine-tuned such as RoBERTa [13] and integrated into the EasyNER pipeline.

## Acknowledgments

I would like to give a special thanks to Sonja Aits for giving me the opportunity to explore NLP at an advanced level, and to Rafsan Ahmed and Salma Kazemi Rashed for their support and getting me up to speed on their existing research.

## References

[1] "Natural language processing." [Online]. Available: https://en.wikipedia.org/wiki/Natural_language_processing

[2] "Relation extraction." [Online]. Available: https://paperswithcode.com/task/relation-extraction

[3] M. L. MacDonald, J. Lamerdin, S. Owens, B. H. Keon, G. K. Bilter, Z. Shang, Z. Huang, H. Yu, J. Dias, T. Minami *et al.*, "Identifying off-target effects and hidden phenotypes of drugs in human cells," *Nature chemical biology*, vol. 2, no. 6, pp. 329–337, 2006.

[4] R. Luo, L. Sun, Y. Xia, T. Qin, S. Zhang, H. Poon, and T.-Y. Liu, "BioGPT: generative pre-trained transformer for biomedical text generation and mining," *Briefings in Bioinformatics*, vol. 23, no. 6, 09 2022, bbac409. [Online]. Available: https://doi.org/10.1093/bib/bbac409

[5] Aitslab, "Aitslab/easyner: A customzable pipeline for information extraction." [Online]. Available: https://github.com/Aitslab/EasyNER

[6] N. Broman, "Bionlp/nils at master-aitslab/bionlp." [Online]. Available: https://github.com/Aitslab/BioNLP/tree/master/nils

[7] I. Beltagy, K. Lo, and A. Cohan, "Scibert: A pretrained language model for scientific text," *arXiv preprint arXiv:1903.10676*, 2019.

[8] C. Arighi, M. Krallinger, and F. Leitner, "Biocreative - chemprot corpus: Biocreative vi." [Online]. Available: https://biocreative.bioinformatics.udel.edu/news/corpora/chemprot-corpus-biocreative-vi/

[9] M. Krallinger, O. Rabal, A. Miranda-Escalada, and A. Valencia, "DrugProt corpus: Biocreative VII Track 1 - Text mining drug and chemical-protein interactions," Jun. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5119892

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[11] "fairseq." [Online]. Available: https://fairseq.readthedocs.io/en/latest/index.html

[12] "Berzelius." [Online]. Available: https://www.nsc.liu.se/systems/berzelius/

[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[14] J. Krucinski, "Bionlp/jacob at master-aitslab/bionlp." [Online]. Available: https://github.com/Aitslab/BioNLP/tree/master/jacob

**Jacob Krucinski** is a third year undergraduate student at the University of Connecticut studying Computer Science and Engineering and Mathematics. For the Spring 2023 semester, he is studying abroad at Lunds Tekniska Högskola in Lund, Sweden. His interests like in computer vision and machine learning(ML), and most recently NLP. He has done 2 internships, one at Lockheed Martin working on computer vision ML model for a surveillance drone, and the other at Medtronic working on ML methods for surgical stapling quality assessment. This summer he will be joining MathWorks in Natick, Massachussetts, USA as an intern in the API engineering team.

## APPENDIX

*Documentation of Key Scripts*

This section contains the main scripts used in my project, particularly modified scripts. All code and scripts can be found at [14].

*Hyperparameter Tuning of BioGPT:*

1) `examples/BC5CDR/Infer_vJacob.sh`: End-to-end RE on BC5CDR test set using hyperparameter tuned model on Berzelius
2) `preprocess_train_vJacob.sh`: Bash script to set up BioGPT training jobs
3) `train_eval/biogpt_loss_curves.ipynb`: Plot train and validation loss curves from a training .out log file

*Free-Form Relation Extraction with BioGPT:* The first two files are in the `free_form_RE` subfolder.

1) `free_form_RE_textgen.py`: Text generation approach using prompt engineering for NER
2) `free_form_RE_textgen_NER_lookup.py`: Text generation approach with manual NER and "rel-is" prompting
3) `examples/RE-BC5CDR/infer_free_form.sh`: Bash script to perform end-to-end RE on any .txt file

*ChemProt and DrugProt Combined Corpus and SciBERT Fine-tuning:* The main scripts I edited can be found in the `jacob_edits` subfolder and are described below:

1) `bert_finetune_job.sh`: Bash script to set up SciBERT training jobs (after `config.json` has been set)
2) `plotting-notebook.ipynb`: Plots epoch-level loss curves, accuracy, precision/recall/F1 scores, and confusion matrices for the SciBERT fine-tuning with the combined ChemProt and DrugProt corpus
3) `plotting-notebook-os.ipynb`: Plots epoch-level loss curves, accuracy, precision/recall/F1 scores, and confusion matrices for the SciBERT fine-tuning with the *oversampled* combined ChemProt and DrugProt corpus

More modified scripts can be found in the `scripts` subfolder described below:

1) `add_custom_labels.py`: Pre-processing script to map CPR labels to the 5 custom classes (see section III-C for details)
2) `bert_finetune.py`: Script to perform SciBERT fine- tuning. I added a new oversampling strategy for the combined ChemProt/DrugProt corpus
3) `roberta_finetune.py`: Script to perform RoBERTA fine- tuning (work in progress)
4) `main.py`: Main script use to run various subtasks of the RE pipeline (pre-processing, training, validation, etc.). I added support for RoBERTa finetune (work in progress)

### A. Bad NER Prompting for Free-form Text RE

### B. Full Confusion Matrices

| Source | Free-form Text and Output | True Entities |
|---|---|---|
| PubTator ID 439781 | IN:<br>List all chemicals and diseases in the following text: After a single oral dose of 4 mg/kg indomethacin (IDM) to sodium and volume depleted rats plasma renin activity (PRA) and systolic blood pressure fell significantly within four hours, suggesting hypotension.<br><br>OUT:<br>PRA and blood pressure returned to normal within 24 hours after IDM was removed from the rats. (ABSTRACT TRUNCATED AT 250 WORDS) The results of this study suggest that the hypotensive effect of IDM in volume depleted rats may be due to inhibition of the renin-angiotensin system. | C: Indomethacin<br>D: hypotension |
| Pubtator 23666265 (partial) | IN:<br>List all chemicals and diseases in the following text: The purpose of the study is to explore the function of P2X3 and NK1 receptors antagonists on cyclophosphamide (CYP)-induced cystitis in rats. METHODS: Sixty female Sprague-Dawley (SD) rats were randomly divided into three groups. The rats in the control group were intraperitoneally (i.p.) injected with 0.9% saline (4 ml/kg); the rats in the model group were i.p. injected with CYP (150 mg/kg); and the rats in the intervention group were i.p. injected with CYP with subsequently perfusion of bladder with P2X3 and NK1 receptors' antagonists, Suramin and GR 82334. Spontaneous pain behaviors following the administration of CYP were observed. Urodynamic parameters, bladder pressure-volume curve, maximum voiding pressure (MVP), and maximum cystometric capacity (MCC), were recorded. Pathological changes in bladder tissue were observed. Immunofluorescence was used to detect the expression of P2X3 and NK1 receptors in bladder. MCC in the model group was lower than the control and intervention groups ($P < 0.01$). Histological changes evident in model and intervention groups rats' bladder included edema, vasodilation, and infiltration of inflammatory cells.<br><br>OUT:<br>($P < 0.01$) Immunofluorescence showed that the expression of P2X3 and NK1 receptors in bladder MCC in the model group was lower than the control and intervention groups. | C: CYP<br>D: Cystitis, pain, edema |
| Custom | IN:<br>List all chemicals and diseases in the following text: A patient perscribed 100 mg/ml cyclosporine suffered from high blood pressure, negative thoughts, and depression.<br><br>OUT:<br>A patient perscribed 100 mg / ml cyclosporine suffered from high blood pressure, negative thoughts, and depression. (ABSTRACT TRUNCATED AT 250 WORDS) The results of this study suggest that cyclosporine may be a risk factor for the development of hypertension and depression in patients with renal failure, and that cyclosporine should be used with caution in patients with hypertension and depression. | C: Cyclosporin<br>D: Depression |

Fig. 18. Free-form text RE via NER prompting results using the model checkpoint from the BioGPT paper. This method did not produce a list of entities, but rather elaborated on certain terms found in the text that the model had likely seen during training.

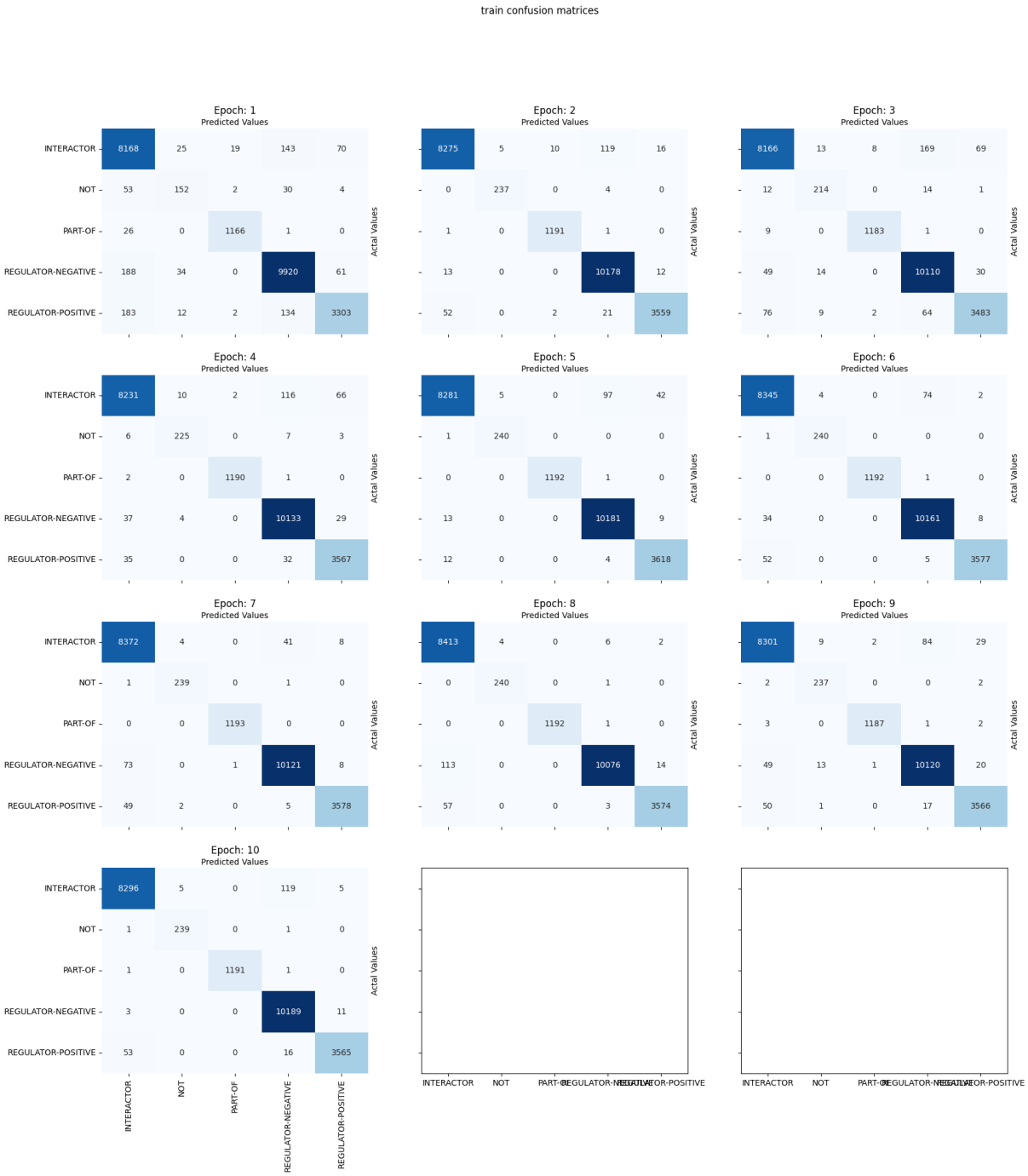


Fig. 19. Confusion matrix for the training set at each of the 10 epochs during the SciBERT fine-tuning on the combined ChemProt and DrugProt corpus.

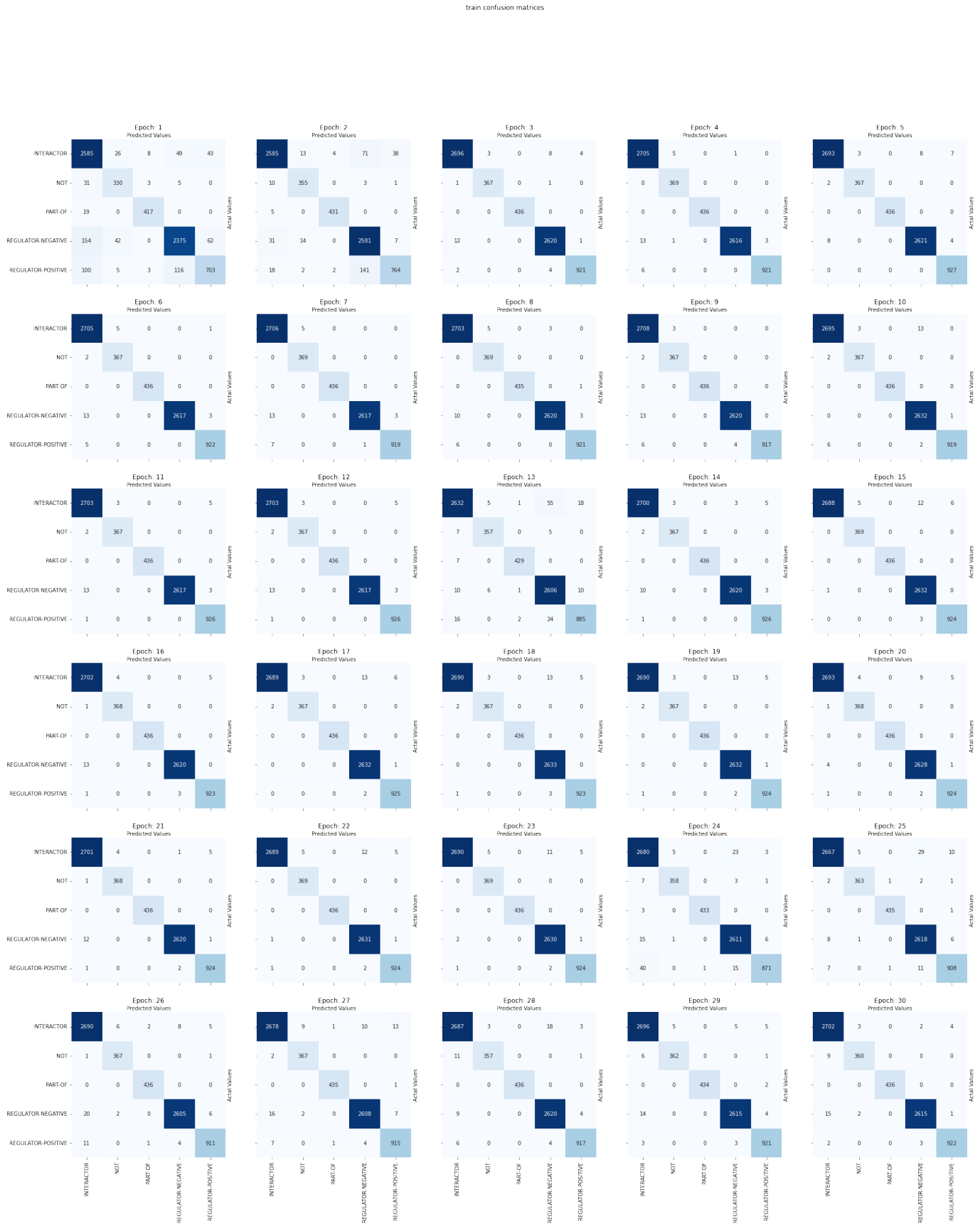


Fig. 20. Confusion matrix for the training set at each of the 20 epochs during the SciBERT fine-tuning on only the ChemProt corpus.

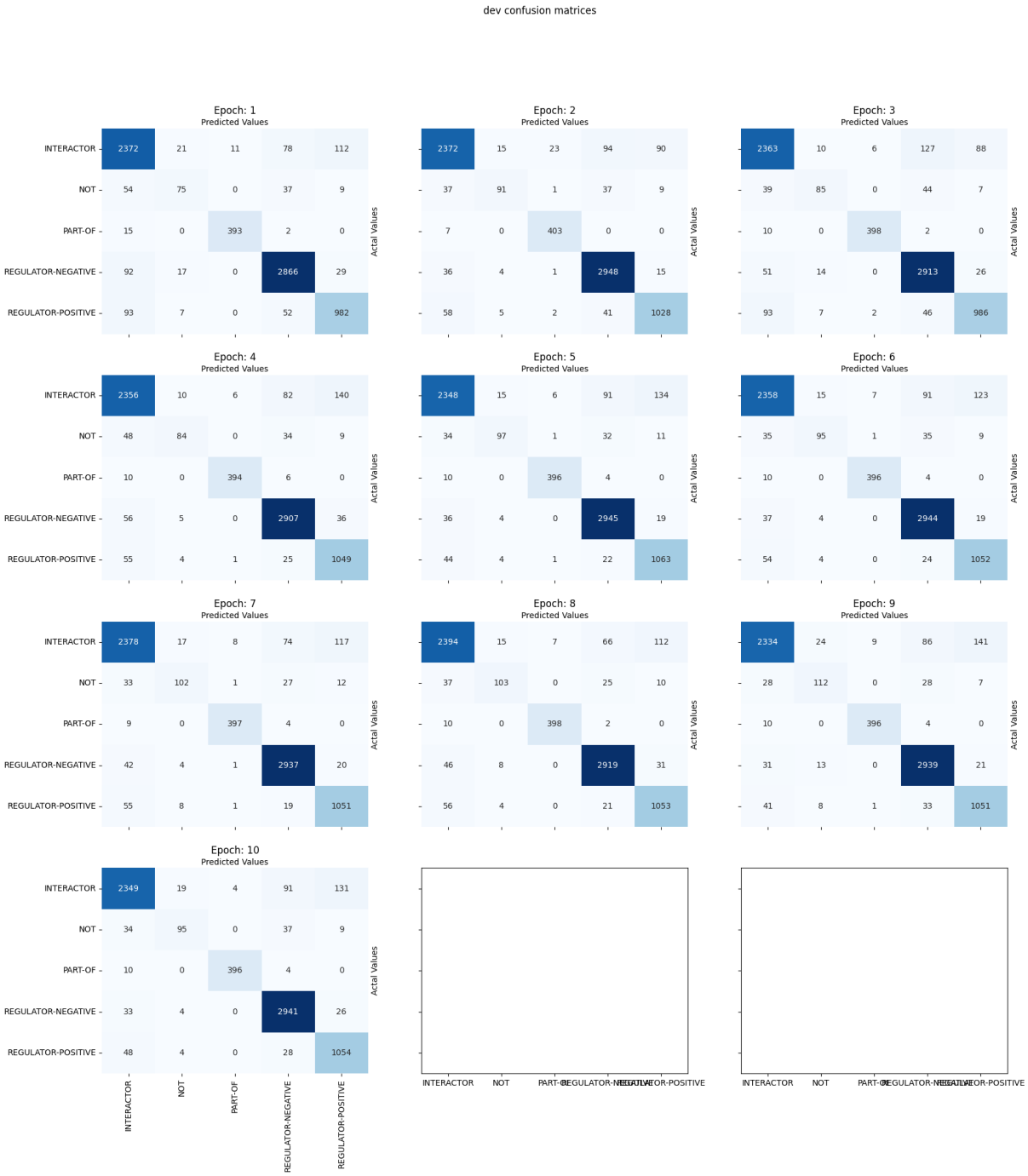


Fig. 21. Confusion matrix for the development set at each of the 10 epochs during the SciBERT fine-tuning on the combined hemProt and DrugProt corpus.

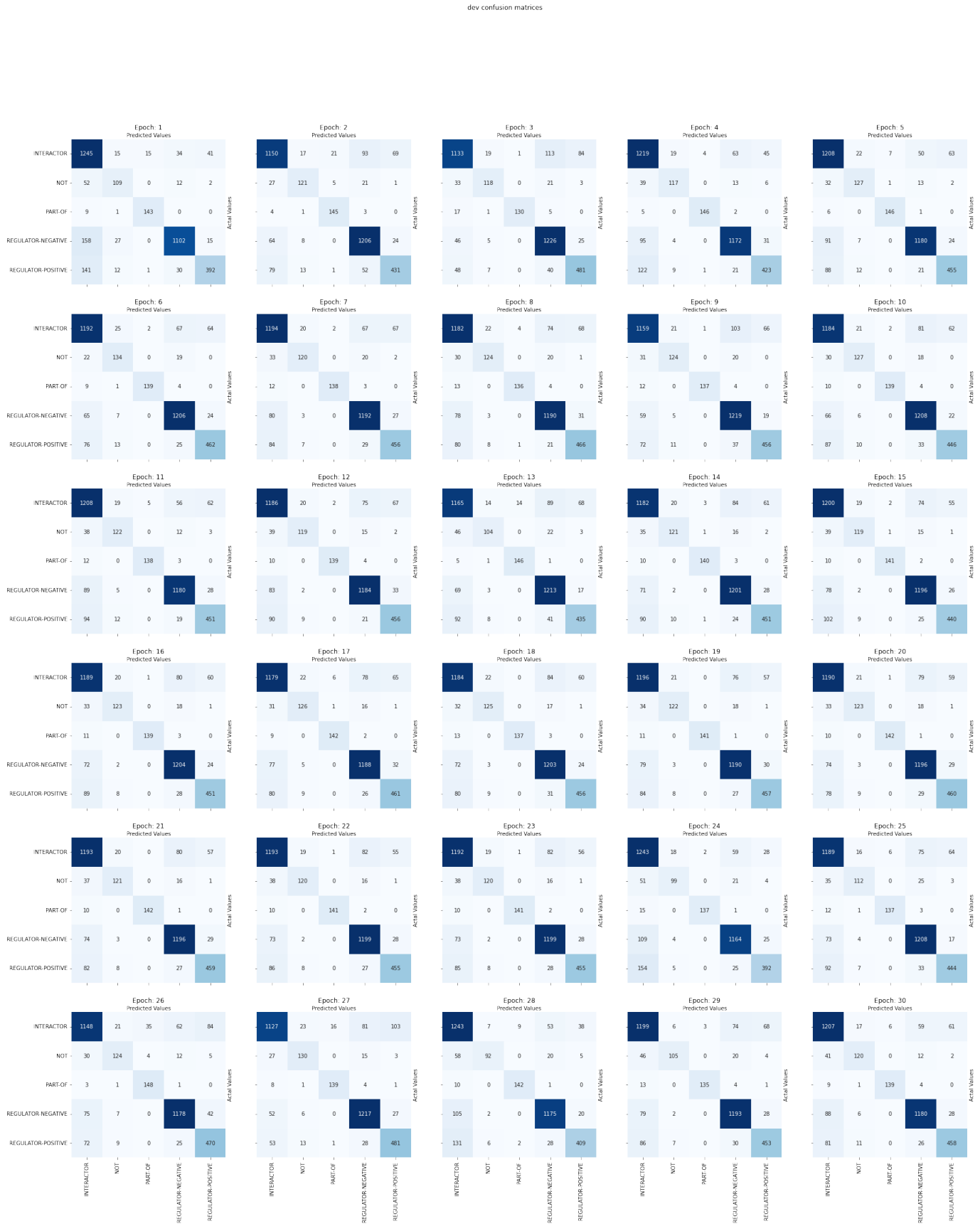


Fig. 22. Confusion matrix for the development set at each of the 20 epochs during the SciBERT fine-tuning on only the ChemProt corpus.

## C. Oversampling Results

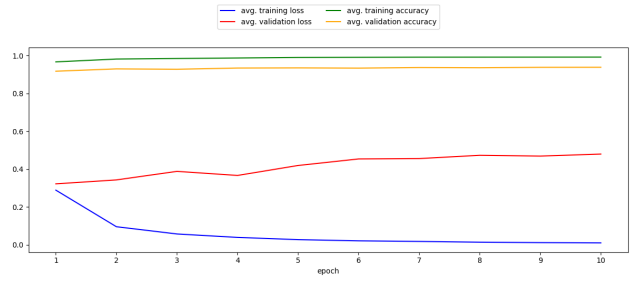


Fig. 23. Train and validation loss and accuracy curves from fine-tuning the SciBERT model on the *oversampled* combined ChemProt and DrugProt corpus for 10 epochs.

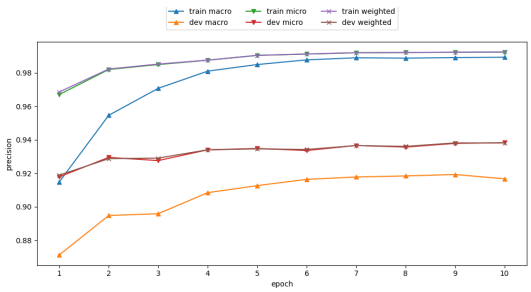


Fig. 24. Macro, micro, and weighted precision scores for each epoch when fine-tuning the SciBERT model on the *oversampled* combined ChemProt and DrugProt corpus for 10 epochs.

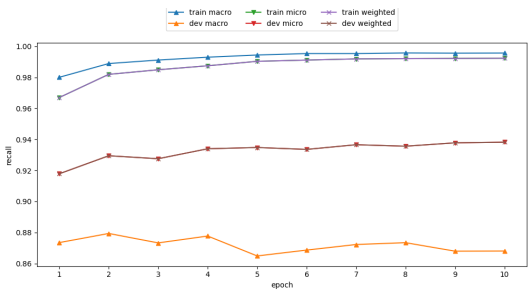


Fig. 25. Macro, micro, and weighted recall scores for each epoch when fine-tuning the SciBERT model on the *oversampled* combined ChemProt and DrugProt corpus for 10 epochs.

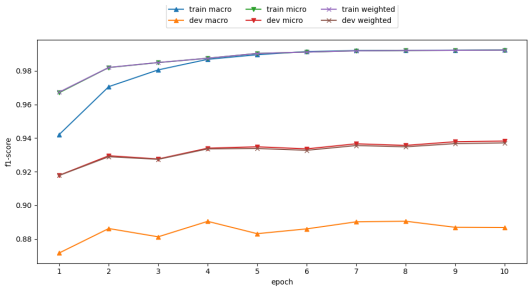


Fig. 26. Macro, micro, and weighted F1 scores for each epoch when fine-tuning the SciBERT model on the *oversampled* combined ChemProt and DrugProt corpus for 10 epochs.


Fig. 27. Confusion matrix for the training set at each of the 10 epochs during the SciBERT fine-tuning on the *oversampled* combined ChemProt and DrugProt corpus.


Fig. 28. Confusion matrix for the development set at each of the 10 epochs during the SciBERT fine-tuning on the *oversampled* combined ChemProt and DrugProt corpus.

Below are the performance summary results for SciBERT fine-tuning on the oversampled combined corpus.

TABLE IV
MACRO, MICRO, AND WEIGHTED PRECISION, RECALL AND F1 SCORES FOR THE SCIBERT MODEL TRAINED ON THE *oversampled* COMBINED CHEMPROT AND DRUGPROT CORPUS AND THE BASELINE MODEL TRAINED ON THE CHEMPROT ONLY CORPUS TRAINED FOR AN EXTENDED 40 EPOCHS (METRICS FROM 20 EPOCH TRAINING WERE NOT AVAILABLE).

| | **Combined Model** | | | **Baseline Model** | |
|---|---|---|---|---|---|
| **Metric** | Train | Dev | Dev change | Train | Dev |
| **Precision** | | | | | |
| macro | 0.9893 | 0.9193 | +0.0080 | 0.9925 | 0.9113 |
| micro | 0.9923 | 0.9382 | -0.0003 | 0.9924 | 0.9385 |
| weighted | 0.9924 | 0.9381 | -0.0001 | 0.9924 | 0.9382 |
| **Recall** | | | | | |
| macro | 0.9957 | 0.8794 | -0.0028 | 0.9943 | 0.8822 |
| micro | 0.9923 | 0.9382 | -0.0003 | 0.9924 | 0.9385 |
| weighted | 0.9923 | 0.9382 | -0.0003 | 0.9924 | 0.9385 |
| **F1-score** | | | | | |
| macro | 0.9924 | 0.8905 | +0.0003 | 0.9928 | 0.8902 |
| micro | 0.9923 | 0.9382 | -0.0003 | 0.9924 | 0.9385 |
| weighted | 0.9923 | 0.9371 | -0.0011 | 0.9924 | 0.9382 |