Chemical–protein relation extraction in PubMed abstracts using BERT and neural networks

Rui Antunes[1], Tiago Almeida[1], João Figueira Silva[1], and Sérgio Matos[1§]
[1]DETI/IEETA, University of Aveiro, Aveiro, Portugal
§ Corresponding author. E-mail: aleixomatos@ua.pt.

***Abstract*—Automatically extracting relations from scientific literature is a major task in biomedical text mining that is helpful in database curation. Particularly, the automatic extraction of chemical–protein interactions is relevant for accelerating drug discovery. This work describes the participation of the BIT.UA team from the University of Aveiro: we use PubMedBERT to create embedding representations for candidate pairs which are then forwarded through a neural network classifier. Our best system achieved a 0.7114 micro-averaged F1-score which is above the official mean by 9 percentage points.**

Keywords—relation extraction; chemical–protein interactions; deep learning; transformer based model.

I. Introduction

Biomedical relation extraction (RE) aims to detect relationships between specific entities in the life sciences domain (such as chemicals, genes, and diseases). The automatic extraction of these interactions assists human experts during their manual curation labor and supports biomedical research. Particularly, finding interactions between drugs (chemicals) and genes (proteins) can be relevant for drug discovery, drug repurposing, and chemical health risk assessments (1).

The BioCreative VII DrugProt (Track 1) challenge (2), similarly to the ChemProt initiative (3), promoted the development of RE systems that can identify interactions between drugs (chemical compounds) and GPROs (gene and protein related objects) in PubMed abstracts. In comparison to the preceding challenge, the DrugProt track organizers prepared a larger dataset and allowed the prediction of more relation types. Also, the organizers formulated an additional sub-track—DrugProt Large Scale—where a much larger dataset containing more than two million PubMed records was built to call for teams that could deliver systems able to make predictions in large scale. Herein, we describe the methods from our participation in the main task (DrugProt) of BioCreative VII Track 1. Due to resource and time limitations, we did not apply our model in the DrugProt Large Scale dataset, but we aim to improve the robustness and execution performance of our system in future work.

II. Data

In this section we detail the data used for development and official evaluation of our system. BioCreative VII Track 1 organizers prepared the DrugProt corpus (2) which was built upon the existent ChemProt corpus (3). It contains PubMed abstracts annotated with chemical and gene entities, and their interactions. The DrugProt dataset has a total of 5000 documents—where 2432 documents are from the ChemProt dataset and 2568 are new documents—and is split into three subsets: *training*, *development*, and *test* with 3500, 750, and 750 documents respectively. However, at the time of the challenge the *test* subset was mixed with 10000 additional 'background' documents to fend against manual annotation. The *training* and *development* subsets were used for developing our model, and final predictions were made in the *test* subset to be evaluated officially by the organizers.

For this task, entities are provided and participants must only focus on the RE problem. In the *training* and *development* subsets gold standard entities and relations are given, whereas in the *test* subset participants only have access to the entities, having to predict the relations.

Table I presents statistics about the DrugProt dataset (only *training* and *development* subsets, since the *test* subset is mixed

TABLE I. DrugProt *training* and *development* subset statistics.

		Training	**Development**
Documents	With no relations	1067	208
	With one or more relations	2433	542
Entities	Chemical	46274	9853
	Protein	43254	9005
Relations	Indirect-downregulator	1329	332
	Indirect-upregulator	1378	302
	Direct-regulator	2247	458
	Activator	1428	246
	Inhibitor	5388	1150
	Agonist	658	131
	Antagonist	972	218
	Agonist-activator	29	10
	Agonist-inhibitor	13	2
	Product-of	920	158
	Substrate	2003	494
	Substrate_product-of	24	3
	Part-of	885	257

RA, TA, and JFS are funded by the FCT—Foundation for Science and Technology (Portuguese funds)—under the grants SFRH/BD/137000/2018, 2020.05784.BD, and PD/BD/142878/2018, respectively.

with background records). It is noticeable that a significant number of documents do not contain any relation, yet these are useful to create negative instances (chemical–protein pairs without any relation) for training a machine learning model. The dataset comprises 13 relation types and is highly unbalanced (scarce relations include 'Agonist-activator', 'Agonist-inhibitor', and 'Substrate_product-of'). The 'Inhibitor' relation is the most frequent relation by a considerable margin. Around 1% of chemical–protein pairs (231 out of 20804) were associated with two relation types, while most positive instances were only associated with one relation.

III. Methods

In this section we describe our methods (pre-processing and deep learning approach) and the five submitted runs. A schematic overview of the base model architecture used in this work is provided in Fig. 1. We implemented the system in Python programming language using the TensorFlow and Hugging Face frameworks.

A. *Pre-processing*

The first step was to pre-process the PubMed records (each document only contained the title and the abstract). We performed sentence splitting using the scispaCy library (4) which is specific for biomedical, scientific and clinical text. To simplify the problem, we decided to target only chemical–protein relations within the same sentence, since relations between entities from different sentences are rare in the dataset. Similarly, we ignored relations between overlapping entities, which are also scarce. Apart from overlapping entities, we considered all combinations between chemicals and proteins within the same sentence to create candidate pairs.

We denote a sample as (x, e_1, e_2), where x is the input sentence, e_1 corresponds to the chemical entity and e_2 to the protein entity. Note that it is possible to have different samples sharing the same input sentence, x, when the sentence has multiple chemical–protein candidate pairs. We restricted the length of the input sentence to 256 tokens, and therefore left and right truncation is applied when the maximum length is exceeded (we centered sentences according to the position of the target entities). Also, if the target entities are not present within a span of 256 tokens, then the corresponding candidate pair is ignored.

B. *Problem formulation*

First and foremost, we provide a formal description of the problem and solution. Relation extraction can be defined as finding a function f that maps each sample to a set of relations, $r = f(x, e_1, e_2)$, where $r \subset R$ and R represents the set that contains all valid relation classes. We relied on neural networks to approximate f and considered it as a composition of f_e and f_c, thus $r = f_c\big(f_e(x, e_1, e_2)\big)$. More precisely, we denote f_e as our encoder function, that aims to create a dense representation of the input which is then classified by f_c, our classifier function. Furthermore, in all our experiments we consider f_e to be fixed, *i.e.*, we do not train any parameter of f_e, and only f_c is trained. An important note is that f_c can be formulated as either a multi-class single-label or multi-label problem: the former case requires the creation of an additional class for representing the absence of a relation, whereas the latter does not.

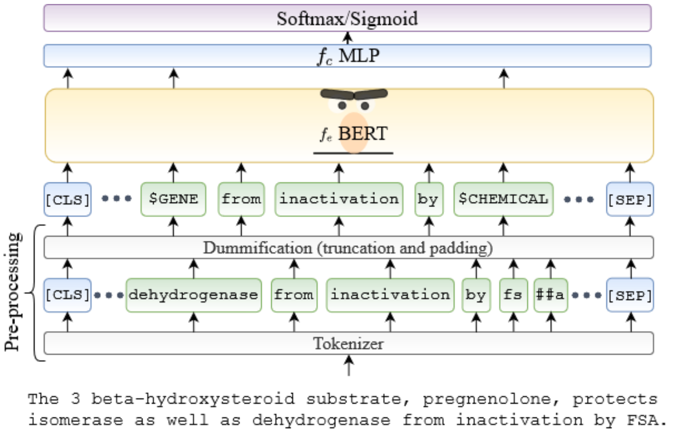
Softmax/Sigmoid
f_c MLP
f_e BERT
[CLS] ••• $GENE from inactivation by $CHEMICAL ••• [SEP]
Dummification (truncation and padding)
[CLS] ••• dehydrogenase from inactivation by fs ##a ••• [SEP]
Tokenizer
Pre-processing
The 3 beta-hydroxysteroid substrate, pregnenolone, protects isomerase as well as dehydrogenase from inactivation by FSA.

Fig. 1. Overview of the base model architecture (BERT-MLP). Sentence example from PubMed identifier 1911436.

C. *Encoder f_e*

For implementing f_e we considered the transformer architecture of BERT (Bidirectional Encoder Representations from Transformers) (5), more precisely the PubMedBERT model that, at the time of writing, presents state-of-the-art results in many biomedical downstream tasks including relation extraction on the ChemProt dataset (6). Furthermore, when feeding x to the transformer model, we experimented different input and output strategies inspired by previous works (6, 7).

1) *Simple case*

The input corresponds to x (no changes) and the output uses the `[CLS]` token and (or) the entities representations.

2) *Dummification case*

Chemical and protein mentions present in input x are replaced with the special tokens `$CHEMICAL` and `$GENE`, respectively, and the output uses the `[CLS]` token and (or) the entities special token representations.

3) *Entity markers case*

We added a special token before (`[E1]`) and after (`[/E1]`) the chemical entity mention, and similarly we added `[E2]` and `[/E2]` before and after the protein mention; for the output we considered the same approach from the simple case 1) where the special tokens are also included.

Note that in cases 1) and 3) an entity can be represented by multiple sub-tokens. Therefore, we compute the maximum or average pooling over all the sub-token representations to derive the final entity representation.

After preliminary experiments we settled for the dummification technique since it presents a simpler and straightforward approach while being as competitive as the entity markers technique, and the output was fixed to the concatenation of the `[CLS]` token and the entities special token representations (`$CHEMICAL`, `$GENE`).

D. Classifier f_c

Regarding f_c, we considered three classifier variants, each with increasing complexity levels:

1) Base

The first variant consisted of a simpler approach, denoted here as 'base', which adopts a two-layer MLP (multi-layer perceptron) that outputs probabilities for each relation class.

2) Attention

The next approach, denoted 'attention', tries to find relation mentions directly on the input sentence by leveraging the searching capabilities of the multi-head attention mechanism $MA(Q, K, V)$. More precisely, we firstly create relation class embedding representations using the textual definitions from the ChemProt annotation guidelines (the textual definition was fed to the transformer model and the resulting `[CLS]` token representation was used). Then, by considering each previously created class representation as queries, Q, and the input sentence, x, as key and values, K and V, we leverage the multi-head mechanism to create a new representation that condenses the input sentence information most related to our classes' representations. The intuition behind this mechanism is to give the model an idea of what type of information we are looking for in each class. The resulting representation is concatenated with the `[CLS]` and entities representations, being then forwarded through a two-layer MLP to predict the final class probabilities.

3) Last layer

The final approach, denoted 'last layer', uses the same architecture from the first method, but the last layer of the transformer model is also trained, *i.e.*, f_e represents the transformer except for the last layer, and f_c starts with this last layer of the transformer model followed by the MLP.

Softmax and sigmoid activations were used for single-label and multi-label predictions, respectively. In both cases, we used class weights inversely proportional to class frequency, and we fine-tuned a multiplicative factor for the weight of the negative class which was set to 1.5. Additionally, we applied a further step to attempt a maximization of the F1-score metric by balancing precision and recall. For that, in the single-label approach we multiplied the predicted probability of the negative class by 0.60 (then, the class with the highest probability is chosen), whereas in the multi-label approach we set the prediction threshold to 0.40 (classes with predicted probabilities above this threshold are chosen). All these values were adjusted according to evaluation on the *development* subset during early experiments.

E. Submitted runs

We made five variants for submission which are detailed here. Runs 1 and 2 use the 'last layer' approach, Run 3 used the 'base' approach, and Runs 4 and 5 used the 'attention' approach. Runs 1–4 were trained as a multi-class single-label classification problem, while Run 5 was trained as a multi-label classification problem. We used a batch size of 64 and a total number of 30 epochs. In Run 1 only the *training* subset was used for model training, and the model from the best epoch on *development* subset was selected, whilst for Runs 2–5 the *training* and *developments* subsets were used for training and the model from the last epoch was used. Lastly, for each run we made an average of the predicted probabilities from four experiments using different random initializations.

IV. Results and discussion

Table II presents the official results shared by the organizers, including our five submitted runs and average statistics from the submissions of all participating teams. Our best method (Run 2) achieved a F1-score of 0.7114 which is above the mean performance by around 9 percentage points. Runs 1 and 2 had a close performance showing that using the *development* subset for monitoring (as validation data for selecting the best epoch) or as additional training data yields similar models. Our baseline system (Run 3) obtained the lowest performance (0.6059 F1-score) but a balancing between precision and recall could prove beneficial, since recall is much higher than precision (0.7080 vs 0.5296). Runs 4 and 5 obtained superior performance when compared to our baseline method (Run 3), demonstrating the effectiveness of using relation class embedding representations.

From these results we conclude that training the last layer of BERT (Runs 1 and 2) brought a great performance improvement, surpassing our baseline model that did not train any BERT layer (Run 3) in about 10 percentage points. We hypothesize that training more layers from the BERT model could further improve system performance, but at a higher computational cost and risk of overfitting.

Our multi-label system (Run 5) achieved 0.6628 F1-score, which is slightly below the similar yet single-label system (Run 4) leading us to believe that predicting multiple relations for a candidate pair is hard and ends up hurting the performance. We suspect this is also because only a small number of pairs were associated with more than one relation type.

Table III presents the detailed results from our best model (Run 2) with the metrics obtained for each relation type. The 'Antagonist' and 'Inhibitor' relation types achieved the highest results with F1-scores above 0.80. Regarding rare relation types (Table I), the model was unable to find 'Agonist-activator' and 'Substrate_product-of' interactions but was able to successfully predict 'Agonist-inhibitor' relations (0.6667 F1-score). At the

TABLE II. Official results in the *test* subset using micro-averaged metrics. The best scores are highlighted in bold.

	Precision	**Recall**	**F1-score**
Run 1[b]	0.6916	**0.7298**	0.7102
Run 2	**0.7003**	0.7229	**0.7114**
Run 3	0.5296	0.7080	0.6059
Run 4	0.6623	0.6794	0.6707
Run 5	0.6750	0.6510	0.6628
Mean[a]	0.6430	0.6291	0.6196
SD[a]	0.1962	0.2473	0.2317

[a] Mean and SD (standard deviation) values calculated from the submissions of all teams.

[b] The same model achieved a 0.7139 F1-score in the *development* subset showing stable generalization.

TABLE III. OFFICIAL GRANULAR RESULTS, PER RELATION TYPE, OF OUR BEST PREDICTIONS (RUN 2). SCORES PRESENTED IN DESCENDING ORDER.

	Precision	**Recall**	**F1-score**
Antagonist	0.8363	0.9346	0.8827
Inhibitor	0.7656	0.8392	0.8007
Activator	0.7584	0.7425	0.7504
Agonist	0.7576	0.7426	0.7500
Indirect-downregulator	0.6295	0.7434	0.6818
Agonist-inhibitor	0.6667	0.6667	0.6667
Indirect-upregulator	0.6493	0.6751	0.6619
Direct-regulator	0.6494	0.6433	0.6494
Part-of	0.6023	0.6974	0.6463
Product-of	0.6193	0.6022	0.6106
Substrate	0.6391	0.5155	0.5707
Agonist-activator	0.0000	0.0000	0.0000
Substrate_product-of	0.0000	0.0000	0.0000

of writing, we do not have access to the *test* subset relation statistics, but we suspect these scarce interactions appear only a few times and it is challenging for the model to successfully detect them. Concerning the remaining relation types, the model achieved F1-scores between approximately 0.57 and 0.75. Excluding the rare relation types, the model obtained the lowest F1-score in the 'Substrate' relation type which indicates the increased difficulty in predicting this interaction.

V. CONCLUSIONS

Our best method achieved a competitive performance, considerably above the official mean, which demonstrates the encoding ability of PubMedBERT for representing biomedical scientific text. However, during the development of our models many aspects were left aside with only a small number of preliminary experiments being carried out. Therefore, we acknowledge that various mechanisms can be further investigated. In future work, we suggest evaluating more thoroughly the impact of using entity markers (7) against entity dummification, exploring other encoder models such as BioELECTRA (8), and addressing the possibility of including cross-sentence relations and relations between overlapping entities.

In this work, we used the tokens from the whole sentence for representing a chemical–protein candidate pair. However, a different approach would be to use only the tokens from the shortest dependency path between the chemical and protein entities, which has been a traditional technique in relation extraction—this approach proved valuable in our past work on the ChemProt task (9).

An important aspect of a RE system is the trade-off between the quality of its predictions and its computational performance, since prompt automatic predictions may be required for user interaction. Following this idea, an interesting research direction would be to make our system less resource-hungry and more scalable to be applied to the DrugProt Large Scale dataset. Finally, a more realistic scenario is the case where the system would only have access to raw text, with gold standard entities not being provided. As such, we leave for future work the development of an end-to-end approach, using joint learning, for entity and relation extraction.

REFERENCES

1. Krallinger,M., Rabal,O., Lourenço,A., et al. (2017) Information retrieval and text mining technologies for chemistry. *Chemical Reviews*, 117(12), pp.7673-7761.

2. Miranda,A., Mehryary,F., Luoma,J., et al. (2021) Overview of DrugProt BioCreative VII track: quality evaluation and large scale text mining of drug-gene/protein relations. *Proceedings of the seventh BioCreative challenge evaluation workshop*.

3. Krallinger,M., Rabal,O., Akhondi,S.A., et al. (2017) Overview of the BioCreative VI chemical-protein interaction track. *Proceedings of the BioCreative VI Workshop*, pp. 141-146.

4. Neumann,M., King,D., Beltagy,I., and Ammar,W. (2019) ScispaCy: fast and robust models for biomedical natural language processing. *Proceedings of the 18th BioNLP Workshop and Shared Task*, pp. 319-327.

5. Devlin,J., Chang,M.W., Lee,K., and Toutanova,K. (2019) BERT: pre-training of deep bidirectional transformers for language understanding. *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)*, pp. 4171-4186.

6. Gu,Y., Tinn,R., Cheng,H., et al. (2020) Domain-specific language model pretraining for biomedical natural language processing. *arXiv:2007.15779*.

7. Soares,L.B., FitzGerald,N., Ling,J., and Kwiatkowski,T. (2019) Matching the blanks: distributional similarity for relation learning. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2895-2905.

8. Kanakarajan,K.R., Kundumani,B., Sankarasubbu,M. (2021) BioELECTRA: pretrained biomedical text encoder using discriminators. *Proceedings of the 20th Workshop on Biomedical Language Processing*, pp. 143-154.

9. Antunes,R. and Matos,S. (2019) Extraction of chemical–protein interactions from the literature using neural networks and narrow instance representation. *Database*, 2019(baz095).