Team RobertNLP at BioCreative VII LitCovid Track: Neural Document Classification Using SciBERT

Subhash Chandra Pujari[1,2], Tim Tarsi[3,4], Jannik Strötgen[1], Annemarie Friedrich[1]
[1]Bosch Center for Artificial Intelligence, Renningen, Germany
[2]Heidelberg University, Germany [3]Robert Bosch GmbH, Germany [4]DHBW, Stuttgart , Germany

***Abstract*—This paper describes our submission to the *BioCreative VII - LitCovid track Multi-label topic classification for COVID-19 literature annotation.* Our system generates embeddings for title, abstract, and keywords using the transformer-based pre-trained language model SciBERT. The classification layer consists of several multi-layer perceptrons, each predicting the applicability of a single label. Our approach, originally developed for hierarchical patent classification, shows a strong performance on the LitCovid shared task, outperforming roughly 75% of the participating systems.**

Keywords—document representation; multi-task learning; multi-label classification.

I. Introduction

Information is of utmost importance in the fight against COVID-19, influencing the decisions of medical professionals and policy makers. Research publications document information related to crucial topics, such as prevention, diagnosis, or treatment. Due to the large number of publications appearing every day, manual curation and search is infeasible. A first step in organizing document collections with the aim of supporting search systems consists in assigning meaningful labels to new documents. The *BioCreative VII shared task on Multi-label topic classification for COVID-19 literature annotation* [10] targets the classification of scientific articles in LitCovid, a literature database of COVID-19 related articles in PubMed. The task requires assigning one or several of seven labels (*Treatment*, *Mechanism*, *Prevention*, *Case Report*, *Diagnosis*, *Transmission*, and *Epidemic Forecasting*) to a document, assuming the text of title and abstract is given along with meta-data information including, for instance, the journal or author-defined keywords.

We participate in the shared task with a system building on prior work on hierarchical multi-label patent classification [5]. In a nutshell, we use the neural transformer-based language model SciBERT [4], which has been pre-trained on scientific text, to encode a document's title, abstract, and keywords as vectors. Finally, the classification layer uses several heads, each consisting of a three-layer perceptron predicting whether a particular label applies to the document or not.

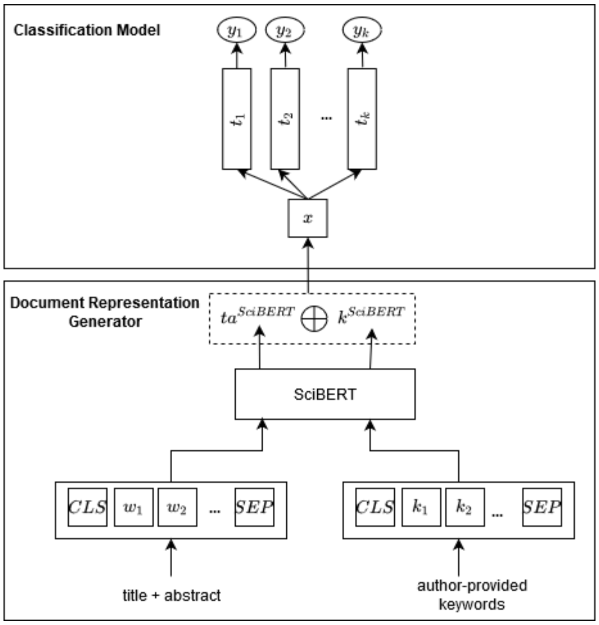

Fig. 1. **System architecture**, showing the best performing document representation setting, concatenating the SciBERT-encoded title and abstract ($\text{ta}^{\text{SciBERT}}$) and the SciBERT-encoded keywords ($\text{k}^{\text{SciBERT}}$).

In this paper, we describe our model architecture and implementation details. In preliminary experiments using several meta-data fields, encoded as multi-hot vectors, we find that these representations are in general informative with regard to the classification task, but also that SciBERT-based embeddings are more effective. The architecture of our best-performing system is shown in Fig.1.

Our system performs almost on par with the third quartile (Q3) in terms of macro- and micro-average F1. Hence, we outperformed roughly 75% of participating systems, demonstrating that our system, originally developed for patent classification, also performs well in the LitCovid domain. Compared to the ML-Net baseline provided by the task organizers, our model performs better across labels, improving macro-average F1 by 13% in relative terms.

II. System Architecture

In this section, we describe our system as submitted to the *BioCreative VII – LitCovid track shared task*.

A. Task Definition and Overview

Labeling articles in the LitCovid dataset constitutes a supervised multi-label classification problem, i.e., the goal is to train an estimator which can map a document d_i to a non-empty set of labels y_i. For training and development, a labeled set of PubMed articles $\{(d_1, y_1), (d_2, y_2), \ldots\}$ is provided. Each document d_i consists of various fields, including the textual content fields *title* and *abstract*, as well as additional meta-data fields such as *publication type* (p_i), *journal name* (j_i), and *keywords* (k_i). We encode the set of labels y_i using a multi-hot vector, i.e., a vector with binary values whose dimensions correspond to labels.

In the following, we describe our model in terms of two steps. First, a document representation generator Ψ computes an *n*-dimensional feature vector x_i for each document d_i. Second, a classifier layer Φ maps the document representation x_i to a label vector y_i. The full model can be described as $y_i = \Phi(\Psi(d_i))$.

B. Document Representation Generator (Ψ)

Our document representation generator takes a document as input and generates an *n*-dimensional representation *x*. A document representation can be generated with only content, or only meta-data, or combining both content and meta-data information. We use the transformer-based pre-trained language model SciBERT [4] for generating text sequence embeddings. SciBERT is pre-trained on a large volume of scientific text including PubMed articles and computer science publications.

When generating sequence embeddings, we first tokenize text into a sequence of word-piece tokens, with a maximum sequence length of 510. Two special tokens ([CLS] and [SEQ]) are added at the start and at the end of the sequence, respectively. Hence, the total maximum input size is 512 word-piece tokens. By means of its self-attention mechanism, SciBERT assigns a 768-dimensional embedding to the [CLS] token that can be considered to represent the meaning of the entire input sequence [6]. We fine-tune SciBERT during training.

For the content fields, i.e., the concatenated text of *title* and *abstract*, we generate an embedding $ta^{SciBERT}$. We also experiment with the meta-data fields *pub_type*, *journal*, and *keywords* as possible sources of information to be included in our document representations. We represent them using multi-hot or one-hot vectors, with dimensionalities equal to the respective number of possible values (see Table II), i.e., $p^{multi-hot}$, $j^{one-hot}$, $k^{multi-hot}$ are of sizes 50, 4251 and 35,766, respectively. As each article can only appear in a single journal, *journal* is encoded as a one-hot vector ($j^{one-hot}$). A document can be associated with multiple *pub_type* values (e.g., *Journal Article, Randomized Controlled Trial,* or *Research Support Non-U.S. Gov't),* hence, *pub_type* is represented as a multi-hot encoded vector ($p^{multi-hot}$). Similarly, since a document might contain multiple author-provided *keywords*, we represent them as a multi-hot encoded vector ($k^{multi-hot}$). The keyword's overall terminology lacks a common taxonomy. For example, "severe acute respiratory syndrome coronavirus 2" and "sars-2" refer to the same entity. The total vocabulary size is 35,700, i.e., the multi-hot encodings of these keywords are rather sparse. We hence project $k^{multi-hot}$ to 768 dimensions using a linear layer. In our experiments, we compare the $k^{multi-hot}$ embeddings to an approach embedding the keywords using SciBERT ($k^{SciBERT}$). We concatenate the keywords, pass them through SciBERT and use the resulting 768-dimensional [CLS] embedding as their representation.

For generating document representations, we first experiment with using $ta^{SciBERT}$, $k^{multi-hot}$, $k^{SciBERT}$, $p^{multi-hot}$, or $j^{one-hot}$ separately. When combining several of these vectors for the document representation *x*, we concatenate them. The lower part of Fig 1. shows the best-performing document representation, consisting of the 1536-dimensional concatenation of $ta^{SciBERT}$ and $k^{SciBERT}$.

TABLE I. META-DATA: NUMBER OF VALUES

Meta-data type	# of unique values	Avg. per instance
pub_type	50	1.55
journal	4,251	1.0
keywords	35,766	4.18

C. Classification Layer (Φ)

In the upper part of Fig. 1, the classification layer takes as input the document representation *x* and trains a separate classification head for each label in the target label set. The classification heads consist of multi-layer perceptrons with three layers each. The dense layers all use ReLU activation. Finally, a two-dimensional softmax layer is used as output, predicting "applicability" vs. "non-applicability" of the corresponding label. This model architecture corresponds to the Transformer-based Multi-Task model (TMM) described by Pujari et al. [5].

III. Experiments

A. Dataset Splits

The dataset provided by the shared task organizers consists of 24,960 training instances, as well as a dev set of 6,239 instances. In our experiments, we report model performance on this dev set. We split the training data into a training set of 22,464 instances and a validation set of size 2,496.

B. Experimental Settings

For the classification layer, we use the hyperparameter settings as defined in Pujari et al. [5], where the dense layers are of size 256 and use ReLU activations. The end-to-end training for the document representation and classification layers is performed with a learning rate of 10^{-5} and a batch size of 64. A dropout of 0.25 is applied across layers. Each of the models is trained on a single Nvidia TeslaV100 GPU, stopping early if the F1-score on the validation set does not improve for 5 epochs. Training the transformer-based models takes 100 to 120 hours, whereas the models using only meta-data information can be trained in one to two hours.

TABLE III. SUBMISSION RESULTS ON BLIND TEST SET AND SHARED TASK STATISTICS.

		Label-based						**Instance-based**		
		Macro-avg.			**Micro-avg.**					
		P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Task Statistics	Mean	86.7	80.1	81.9	89.7	86.2	87.8	89.9	88.9	89.3
	Std	6.0	7.9	7.0	5.4	4.8	4.3	5.2	4.5	4.6
	Q1	84.6	75.5	76.5	88.0	84.5	85.4	87.0	86.2	86.7
	Median	88.4	83.9	85.3	91.1	88.4	89.3	91.9	91.0	91.3
	Q3	**90.8**	85.6	**86.7**	**92.5**	89.7	**90.8**	**93.5**	**91.9**	**92.5**
Models	Baseline (ML-Net) [1]	83.6	73.1	76.6	87.6	81.4	84.4	88.5	85.1	86.8
	TMM - $ta^{SciBERT}$ [5]	85.7	86.1	85.6	91.2	88.9	90.0	93.3	91.8	**92.5**
	TMM - $ta^{SciBERT} + k^{multi\text{-}hot}$	84.7	**87.3**	85.4	89.3	**90.2**	89.7	91.5	92.2	91.9
	TMM - $ta^{SciBERT} + k^{SciBERT}$	89.0	84.8	86.6	92.2	88.6	90.3	**93.5**	91.3	92.4

TABLE II. RESULTS ON DEVELOPMENT SET

Document Representation	**Label-based**					
	Macro-avg.			**Micro-avg.**		
	P	*R*	*F1*	*P*	*R*	*F1*
$p^{multi\text{-}hot}$	46.2	29.1	31.7	59.1	39.7	47.5
$j^{one\text{-}hot}$	59.6	33.4	41.5	70.6	45.9	55.6
$k^{multi\text{-}hot}$	61.3	42.0	49.6	71.3	51.3	59.7
$k^{SciBERT}$	69.7	49.4	57.7	78.3	57.8	66.5

C. Evaluation Metrics

Results are evaluated using the following metrics. The *label-based* evaluation metrics compute precision, recall, and F1 for each label using sklearn.metrics. For macro-average scores, precision and recall are computed separately for each label, and then unweighted averages are formed. The label-based macro-average F1 is computed as the average of per-label F1-scores. Label-based micro-average precision and recall are computed by globally counting true positives, false negatives, and false positives of label assignments. Instance-based scores compute precision, recall and F1-score per instance and then average.

D. Experimental Results

Table II reports our results on the development set when using one document representation vector at a time. We find that out of the meta-data fields, the keywords are most informative with regard to the classification task, and that the SciBERT-based embeddings outperform the multi-hot encodings. Table III reports results on the blind test set using the shared task statistics as provided by the organizers. The last three rows show the results of our submissions with three different document representation settings: only encoding title and abstract with SciBERT ($ta^{SciBERT}$), SciBERT-encoded title and abstract concatenated with a multi-hot encoded keyword vector ($ta^{SciBERT} + k^{multi\text{-}hot}$), as well as SciBERT-encoded title and abstract concatenated with the SciBERT-encoded keywords embedding ($ta^{SciBERT} + k^{SciBERT}$). Among our models, $ta^{SciBERT} + k^{SciBERT}$ performs best on the blind test set, at par with Q3 statistics of the task, which means that it is better than roughly 75% of the participating systems. Also, when comparing with the ML-Net [1] baseline, we observe that all of our submitted runs are better. Particularly, there is a substantial gain in both instance-based and label-based F1-scores.

IV. Related Work

The terms text classification and document classification are often used interchangeably. However, the former is concerned with classifying content fields, whereas the latter can also incorporate meta-data information. Kowsari et al. [7] provide a comprehensive survey on text classification. Classifying a document into a set of predefined labels or categories is useful for search-related tasks, e.g., the labels can be used for filtering documents [9]. The shared task dataset is taken from the LitCovid database [3], a collection of PubMed articles related to COVID-19 research, which is updated every day, adding new documents with manually labeled categories. In addition, the task provides ML-NET [1] as a baseline, which uses a BiLSTM and ELMo for classifying biomedical documents. Using a transformer model pretrained on the biomedical domain text, i.e., BioBERT, Gutierrez et al. [8] showed improved topic classification performance over the LitCovid dataset compared to their baselines using LSTMs, CNNs, BERT-base, and BERT-large. Outside of the biomedical domain, Pujari et al. [5] proposed a transformer-based multi-task model (TMM) for classifying patents into target labels belonging to a patent classification taxonomy. Their model trains a single task for each label in the taxonomy with a shared SciBERT layer. We extend this work by generating a document representation combining content and meta-data fields.

V. Conclusion

In this paper we have described our submission to the BioCreative VII - LitCovid track, which shows a significant improvement over the baseline system (ML-Net) [1] and is at par with the Q3 task submission statistics. We reuse the model setting from our previous work on patent classification [5] and enrich the document representation of the textual content (title + abstract) embedding further by incorporating an embedding of the author-provided keywords, which are part of each document's meta-data. Hence, we have demonstrated the cross-domain applicability of our neural system originally developed in the context of multi-label patent classification.

Acknowledgments. We thank Michael Gertz, Mark Giereth, and Fryderyk Mantiuk for insightful discussions related to this work. We also thank Sophie Henning and Stefan Grünewald for their useful comments on this paper, as well as the shared task organizers for their kind support.

REFERENCES

1. Du, J., Chen, Q., Peng, Y., Xiang, Y., Tao, C. and Lu, Z., 2019. ML-Net: multi-label classification of biomedical texts with deep neural networks. Journal of the American Medical Informatics Association, 26(11), pp.1279-1285.
2. Chen, Q., Allot, A. and Lu, Z., 2020. Keep up with the latest Coronavirus research. Nature, 579(7798), pp.193-194.
3. Chen, Q., Allot, A. and Lu, Z., 2021. LitCovid: an open database of COVID-19 literature. Nucleic acids research, 49(D1), pp.D1534-D1540.
4. Beltagy, Iz, Kyle Lo and Arman Cohan, 2019. SciBERT: A Pretrained Language Model for Scientific Text. In Proceedings of Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics.
5. Pujari, S. & Friedrich, A. & Strötgen, J. 2021. A Multi-task Approach to Neural Multi-label Hierarchical Patent Classification Using Transformers. European Conference on Information Retrieval (ECIR). Springer.
6. Devlin, J., Chang, M., Lee, K., & Toutanova, K., 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 17th meeting of the North American Association for Computational Linguistics (NAACL). Association for Computational Linguistics.
7. Kowsari, K., Meimandi, K.J., Heidarysafa, M., Mendu, S., Barnes, L.E., & Brown, D.E., 2019. Text Classification Algorithms: A Survey. Inf., 10, 150.
8. Gutierrez, B.J., Zeng, J., Zhang, D., Zhang, P., & Su, Y., 2020. Document Classification for COVID-19 Literature. ArXiv, abs/2006.13816.
9. Manning, C. D., Raghavan, P., Schütze, H., 2008. Introduction to Information Retrieval. Cambridge, UK: Cambridge University Press. ISBN: 978-0-521-86571-5.
10. Chen, Q., Allot, A., Robert, L., Doğan, R. I. and Lu, Z., 2021. Overview of the BioCreative VII LitCovid Track: multi-label topic classification for COVID-19 literature annotation. Proceedings of the seventh BioCreative challenge evaluation workshop.