Multi-label classification of COVID-19-related articles with an autoML approach

Team E8@IJS at BioCreative VII LitCovid Track 5

Ilija Tavchioski[2,3], Boshko Koloski[1,2], Blaž Škrlj[1,2], Senja Pollak[1,2]
Affiliation: [1]Jožef Stefan Int. Postgraduate School, Ljubljana, Slovenia
[2]Jožef Stefan Institute, Ljubljana, Slovenia
[3]Facutly of Computer and Information Science,University of Ljubljana, Slovenia

***Abstract*— The rapid growth of literature related to the COVID-19 pandemic results in a multitude of articles which cannot be manually labeled due to the lack of human resources, In this work we present a solution to the shared task titled LitCovid track Multi-label topic classification for COVID-19 literature annotation. Our proposed solution constructs classifiers for each class by using an autoML system for text named autoBOT. Albeit the proposed system performed sub-optimally in terms of recall, it offered better-than-baseline (macro) precision, indication that automated representation learning is a promising approach to multilabel classification of COVID-19-related texts.**

Keywords—component; autoML; COVID-19 literature classification, multi-label classification

I. Introduction

The COVID-19 pandemic has drastically changed our daily life. Challenges including funding healthcare resources, developing a vaccine, research of new possible treatments for coronavirus have brought together governments and researchers from all over the world, resulting in a very high and rapid production of research output from the scientific community.

The number of biomedical articles in the last decade has been steadily increasing. This trend has been continuing during the COVID-19 pandemic where the number of COVID-19 related articles started to increase exponentially. Every month there are 10,000 new COVID-19 related articles(1). The LitCovid database(2) of COVID-19 related articles has already grown to over 100,000 articles. LitCovid database is adding new COVID-19 related articles on a daily basis, creating an increased need for automated annotation of articles with predefined topics information (e.g. Treatment, Diagnosis etc.), which is crucial for better understanding of the literature. Annotation of topics to articles from LitCovid database is a standard multi-label classification (MLC) problem, where each article can be annotated with one or more labels (topics). Increasing the predictive accuracy of automated topic prediction would be beneficial for curators, as it would speed up the annotation process, and to researchers, facilitating the experts' insights into the ever-growing body of literature(1).

The remainder of this paper is structured as follows: in section II we are discussing about approaches for solving MLC problems and related work with COVID-19 related literature, in Section III we present the statistics of our given data, in section IV we explain the methods we used in order to solve the given problem, in Section V we present and explain the obtained results by our experiments, in Section VI we present the obtained results from our final submission on the track and the conclusion and future work is present in Section VIII.

II. Background and related work

Text mining methods have been significantly improved over the past decades, and this need for automation during the COVID-19 pandemic and the rapid growth of COVID-19 text corpora is an opportunity for researchers to test them(3). Such an example is also our experiment on the multi-label classification problem related to the literature from the LitCovid database, performed in the scope of the *LitCovid track Multi-label topic classification for COVID-19 literature annotation*(1). In this section, we cover selected related work, from COVID-19 literature mining, multi-label classification and automated machine learning.

In terms of *automated approaches to COVID-19 literature,* there are many text mining tools being developed(4) addressing a wide range of tasks. For automated literature exploration, the tools include bobble-like visualization of the COVID-19 literature using keyword groups(5), Watson Annotator of Clinical data[1] for highlighting the key terms data[2] , a Google search engine[3] that can identify publications based on a natural language-based query, a literature prioritization system COVID19 Explorer(6), automated knowledge discovery methods from COVID-19 literature(7), etc. A more comprehensive overview of related literature databases and tools(3) is provided by CDC[4].

[1] https://www.ibm.com/cloud/watson-annotator-for-clinical-data

[2] https://www.ibm.com/cloud/watson-annotator-for-clinical-data

[3] https://covid19-research-explorer.appspot.com/

[4] cdc.gov/library/researchguides/2019novelcoronavirus

The focus of this paper is on automated topic assignment where the most similar task has been addressed by Jimenez et al. (8). They also used the data from the COVID-19 literature databases (LitCovid(2)) and used traditional machine learning models like Support Vector Machines (SVM) and Linear Regression (LR) along with several Neural Network's based models such as: Long-Short Term Memory (LSTM)(9), recurrent neural networks, XML-CNN (10), BERT (11) and BioBERT (12). The BioBERT model has shown a state-of-art result with an F1-score of 0.86(8).

Next we cover background and related work in *multi-label classification.* Multi-label classification (MLC) task is defined as follows: given an article and a set of possible labels, in our case denoting topics to which an article can belong, an algorithm is asked to predict which of these labels should be assigned to an article to describe its topics best. There are several approaches in order to solve multi-label classification problems using machine learning. The **Binary Relevance** approach transforms an MLC problem into multiple binary classification problems. For each label, a corresponding binary classifier predicts whether it can be assigned to an article or not(13). Similarly, in a **Classifier Chain** approach, we transform our MLC problem into a binary classification, and for each label, we have a corresponding binary classifier, but we add the predictions of previous classifiers as new features to consider the relationship between labels(13). In **Ranking by Pairwise Comparison**, for each pair of labels we train a classifier to predict which of both labels is more likely to be assigned to the instance (for training, we only use instances that do not have the same label). After the prediction of all classifiers using a voting technique the classifier predicts the labels(13). **Label Powerset** transforms a MLC problem into a multi-class classification problem. The possible classes are all possible sets that can be formed from possible labels in order to predict which set of labels will be assigned to the instance. Finally, **Multi-label Decision Tree** approach adopts the decision tree algorithm for multi-label classification(13). **Sequence Generation Model for MLC** is an approach that transforms the MLC problem into a Sequence Generation problem that is structured with encoder, decoder, and attention mechanism using bidirectional Long-Short Term Memory recurrent neural networks.

In our work, we have opted to use the Binary Relevance approach since it has the simplest implementation in order to solve the presented MLC problem. Using other, potentially more computationally expensive options was left for further work. Finally, as we opted for an *automated machine learning approach*, we also briefly introduced this paradigm. The key idea of Auto-ML is that parts of the learning procedure are modularized and automatically explored. Development of approaches for automatic learning renders possible fast prototyping---instead of spending days in deciding to what extent the current data is suitable for learning---autoML systems offer quick and effortless answers to such questions, greatly speeding up the machine learning development and deployment process. Automatic learning of machine learning pipelines has been thoroughly explored for tabular data (e.g. AutoWEKA(14) and auto-sklearn(15)). For example, AutoWEKA and auto-sklearn employ Bayesian optimization(16) for scalable and efficient exploration of such hyperparameter spaces. Another example of automated learning is conducted with TPOT(17), a tool for automatic construction of Scikit-learn workflows. In our work, we use autoBOT (18), a recently introduced AutoML system suitable for the classification of texts in a language-agnostic setting. Even though the initial version of autoBOT was not aimed at multi-class classification, we saw this task as an opportunity to explore its out-of-the-box performance.

In this paper, we thus employed the autoBOT for the first time on a multi-label classification problem and contributed to automated approaches for labelling COVID-19 literature.

III. The LitCovid data set

The data set consists of articles collected from the LitCovid database, a collection of COVID-19 related literature. We were provided three data subsets. One for training, one for development and another one for testing. We have used the development set for internally evaluating our models. In each of the data sets provided by the *LitCovid track Multi-label topic classification for COVID-19 literature annotation* task organizers, each article consists of the title, abstract and additional meta information (such as journal of publication, DOI of the journal, present keywords), and finally the label. Based on their content, the articles were labeled with up to seven labels. Table I breaks down the distribution of labels in the training and development set respectively.

TABLE I. Label distributions.

	Training data set	**Dev data set**
Size (documents)	24960	6239
Prevention	11102 (44.48 %)	2750 (44.08 %)
Treatment	8717 (34.2 %)	2207 (35.37 %)
Diagnosis	6193 (24.81 %)	1546 (24.78 %)
Mechanism	4438 (17.78 %)	1073 (17.2 %)
Case Report	2063 (8.27 %)	482 (7.72 %)
Transmission	1088 (4.35 %)	256 (4.1 %)
Epidemic Forecasting	645 (2.58 %)	192 (3.08 %)

IV. Methodology

The following section presents a description of our method with corresponding steps, as well as evaluation measures used.

A. Transforming MLC into Binary Classifications

Since this task is defined as a multi-label classification problem, one of the techniques to tackle this type of problem

is to consider each label with a separate (binary) classifier. For each label we define two possible outputs 1 or 0, where one means that the corresponding label is assigned to that instance in the training data and 0 means that the label is not assigned to that instance. Hence, seven different binary classifiers are used to obtain the final output space.

B. *The autoML approach used*

In our work for binary classification, we used the autoBOT (Automatic Bags-Of-Tokens) system(18), with some task-specific modifications. autoBOT is a system that can efficiently learn from multiple representations of a given document set[5]. The main idea underlying autoBOT is *representation evolution* -- by learning to re-weight different representations, including token, sub-word and sentence-level features (contextual and non-contextual), the system identifies the final representation suitable for a given task. This system requires minimal user input -- minimally, only specification of which representations are to be considered and the evolution's time. We considered three different autoBOT's configurations:

- **[N] Neural.** This autoBOT variant includes two doc2vec-based latent representations, each of dimension 512.
- **[NS-I] Neurosymbolic-0.1** This autoBOT variant includes both symbolic and sub-symbolic features. The symbolic features include features based on words, characters, part-of-speech tags and keywords. The sparsity of 0.1 implies that the dimension of symbolic sub-spaces will be 5,120, because the dense dimension is set to 512 and the sparsity presents the quotient of dense dimension and final dimension.
- **[NS-II] Neurosymbolic-0.02** As the name suggests this configuration is similar to the previous variant with only one difference, i.e. the sparsity parameter, which is set up to 0.02 and accordingly the dimension of symbolic features is 25,600.

C. *Evaluation measures*

For evaluation, we used the measures defined by the competition organizers[6]. These include: the precision, recall and F1, average in micro, macro, weighted and `samples` manner. For additional information, please consult the main paper describing this competition [7].

Across all experimental settings, we set the time-constraint of the search to 8 hours.

[5] See https://skblaz.github.io/autobot/features.html

[6] https://github.com/ncbi/biocreative_litcovid/blob/main/biocreative_litcovid_eval.py

[7] https://biocreative.bioinformatics.udel.edu/tasks/biocreative-vi/track-5/

V. Internal evaluation

We used only COVID-19 papers' abstracts as input for training; since autoBOT already performs keyword detection, we did not consider keywords at this time. The considered autoML approach was learned only from the training data, and its predictions were evaluated on the development data (serving as an internal test set, unseen during the training).

TABLE II. Overview of the results (internal evaluation on the development set) - The model column denotes an autoBOT configuration.

Average	Model	Precision	Recall	F1-score
Label-based Micro	N	0.8783	0.7964	0.8353
	NS-1	**0.8834**	0.8040	0.8418
	NS-II	0.8730	**0.8197**	**0.8455**
Label-based Macro	N	0.8544	0.6070	0.6456
	NS-1	**0.8716**	0.6271	0.6701
	NS-II	0.8587	**0.6421**	**0.6770**
Label-based Weighted	N	0.8744	0.7964	0.8170
	NS-1	**0.8809**	0.8040	0.8263
	NS-II	0.8704	**0.8197**	**0.8301**
Label-based Samples	N	0.8559	0.8353	0.8290
	NS-1	0.8637	0.8428	0.8364
	NS-II	**0.8674**	**0.8557**	**0.8442**
Instance-based Mean	N	0.8559	0.8353	0.8455
	NS-1	0.8637	0.8428	0.8531
	NS-II	**0.8674**	**0.8557**	**0.8615**

A. *Evaluation of different configurations*

In Table II, we present an overview of the configurations' performances with respect to a different weighting of Precision, Recall and F1 on label-based setting. The total amount of present labels across all documents is 8506 (the Support). It can be seen that the second configuration, i.e. *Neurosymbolic-0.1*, performed on average better when

considering precision, even though in terms of other metrics, the third configuration *Neurosymbolic-0.02* performed best.

B. Analysing performance for different topics

In this section, we analyse how the autoBOT's configuration with the highest precision (*Neurosymbolic*-0.1) performs on different topics (labels).

TABLE III. RESULTS PER LABEL: AUTOBOT CONFIGURATION NEUROSYMBOLIC-0.1 (INTERNAL EVALUATION ON THE DEVELOPMENT SET)

Label	**Precision**	**Recall**	**F1-score**	**Support**
Prevention	0.9120	0.9196	0.9158	2750
Treatment	0.8761	0.8351	0.8546	2207
Diagnosis	0.8560	0.7652	0.8081	1546
Mechanism	0.8600	0.7903	0.8237	1073
Case Report	0.8811	0.7842	0.8299	482
Transmission	0.8333	0.0195	0.0382	256
Epidemic Forecasting	0.8833	0.2760	0.4206	192

Based on the results provided in Table III, where Precision, Recall and F1-scores are provided, we can conclude that we have a similar precision score for all labels. This is not the case with the recall because the score for labels "Transmission" and "Epidemic Forecasting" are multiple times lower than the recall of other labels. Similar is true for F1-score for those labels. The reason can be in a lower number of instances that are labelled with those topics in the training set. As it can be seen from Table I, "Transmission" represents only 4.35% and "Epidemic Forecasting" only 2.58% of training set examples.

VI. EVALUATION ON THE OFFICIAL TEST SET

In the following table are shown the results achieved on the final test set for our three model configurations. For comparison we also provide the organizers' baseline model (19). Our *Nerosymbolic-01* autoBOT configuration outperforms the baseline model at label-based precision, but achieves lower recall, and in consequence also lower F1-score. The results are listed in Table IV.

TABLE IV. OVERVIEW OF THE RESULTS (INTERNAL EVALUATION ON THE DEVELOPMENT SET)

Average	**N**	**NS-I**	**NS-II**	**Baseline** (19)
Label-based micro precision	0.8788	**0.8930**	0.8771	0.8756
Label-based micro recall	0.7757	0.7826	0.8113	**0.8142**
Label-based micro f1	0.8240	0.8342	0.8430	**0.8437**
Label-based macro precision	0.8720	**0.9175**	0.7611	0.8364
Label-based macro recall	0.6832	0.6185	0.6435	**0.7309**
Label-based macro	0.7382	0.6724	0.6799	**0.7655**
Instance-based precision	0.8457	0.8517	0.8589	**0.8849**
Instance-based recall	0.8121	0.8200	0.8449	**0.8514**
Instance-based f1	0.8286	0.8355	0.8518	**0.8678**

In conclusion, the baseline dominates with respect to most metrics apart from *macro precision*, where our method achieves almost **8%** improvement. We believe this result is an imminent tradeoff of how the used system assigns labels; there is a substantially lower amount of labels assigned, implying lower rate of false positives (higher precision). Lower recall indicates that not all labels are retrieved (too few are perhaps predicted).

VII. CONCLUSIONS AND FUTURE WORK

With our best autoBOT configuration, we have achieved precision above the baseline provided by the task organisers, but did not improve over the baseline's recall and F1-scores. Nevertheless, we have shown that the results are still competitive by using the autoBOT automated machine learning approach with nearly no task-specific adaptations. Increasing amounts of text corpora yield multiple interesting text mining problems. The potential of autoBOT is its capacity of adaptation to a wide range of tasks with minimal human

effort. Moreover, by exploiting genetic algorithm-based feature representation search, the considered approach learns feature type-level weights which can potentially be transferred across tasks.

As further work, we plan to explore the effect of including different sources of background knowledge into the learning process, as well as information about the authors of the articles. We did not test this feature in our solution, but as autoBOT supports simultaneous inclusion of multiple triplet bases, offering the opportunity to systematically investigate the effects of different types of background knowledge, this seems a promising extension of our work.

Availability

The code and the experiments are available at: *https://gitlab.com/iletavcioski/biocreative-vii-5*

Acknowledgment

The work was supported by the Slovenian Research Agency through a young researcher grant of (BŠ) Knowledge technologies core research program (P2-0103) and SDM-Open research project (ERC Complementary scheme, N2-0078). This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media) and project TAILOR (952215).

References

1. Qingyu Chen, Alexis Allot, Robert Leaman, Rezarta Islamaj Doğan, and Zhiyong Lu. Overview of the BioCreative VII LitCovid Track: multi-label topic classification for COVID-19 literature annotation. Proceedings of the seventh BioCreative challenge evaluation workshop. 2021.

2. Chen, Q., Allot, A., & Lu, Z. (2021). LitCovid: an open database of COVID-19 literature. Nucleic acids research, 49(D1), D1534-D1540.

3. Wang, L. L., & Lo, K. (2021). Text mining approaches for dealing with the rapidly expanding literature on COVID-19. Briefings in Bioinformatics, 22(2), 781-799.

4. Hutson, M. (2020). Artificial-intelligence tools aim to tame the coronavirus literature. Nature.

5. Le Bras, P., Gharavi, A., Robb, D. A., Vidal, A. F., Padilla, S., & Chantler, M. J. (2020). Visualising covid-19 research. arXiv preprint arXiv:2005.06380, 1.

6. Škrlj, B., Jukič, M., Eržen, N., Pollak, S., & Lavrač, N. (2021, October). Prioritization of COVID-19-related literature via unsupervised keyphrase extraction and document representation learning. In International Conference on Discovery Science (pp. 204-217). Springer, Cham.

7. Martinc, M., Škrlj, B., Pirkmajer, S., Lavrač, N., Cestnik, B., Marzidovšek, M., & Pollak, S. (2020, October). Covid-19 therapy target discovery with context-aware literature mining. In International Conference on Discovery Science (pp. 109-123). Springer, Cham.

8. Jiménez Gutiérrez, B., Zeng, J., Zhang, D., Zhang, P., & Su, Y. (2020). Document classification for COVID-19 literature. arXiv e-prints, arXiv-2006.

9. Adhikari, A., Ram, A., Tang, R., & Lin, J. (2019, June). Rethinking complex neural network architectures for document classification. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (pp. 4046-4051).

10. Liu, J., Chang, W. C., Wu, Y., & Yang, Y. (2017, August). Deep learning for extreme multi-label text classification. In Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval (pp. 115-124).

11. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805..

12. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., & Kang, J. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, 36(4), 1234-1240.

13. Ahmed, N. A., Shehab, M. A., Al-Ayyoub, M., & Hmeidi, I. (2015, April). Scalable multi-label Arabic text classification. In 2015 6th International Conference on Information and Communication Systems (ICICS) (pp. 212-217). IEEE.

14. Thornton, C., Hutter, F., Hoos, H. H., & Leyton-Brown, K. (2013, August). Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms. In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 847-855).

15. Feurer, M., Klein, A., Eggensperger, K., Springenberg, J. T., Blum, M., & Hutter, F. (2019). Auto-sklearn: efficient and robust automated machine learning. In Automated Machine Learning (pp. 113-134). Springer, Cham.

16. Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. Advances in neural information processing systems, 25.

17. Olson, R. S., & Moore, J. H. (2016, December). TPOT: A tree-based pipeline optimization tool for automating machine learning. In Workshop on automatic machine learning (pp. 66-74). PMLR.

18. Škrlj, B., Martinc, M., Lavrač, N., & Pollak, S. (2021). autoBOT: evolving neuro-symbolic representations for explainable low resource text classification. Machine Learning, 110(5), 989-1028.

19. Du, J., Chen, Q., Peng, Y., Xiang, Y., Tao, C., & Lu, Z. (2019). ML-Net: multi-label classification of biomedical texts with deep neural networks. Journal of the American Medical Informatics Association, 26(11), 1279-1285.