Extreme Multi-Label Classification applied to the Biomedical and Multilingual Panorama

Andre Neves, Andre Lamurias and Francisco M. Couto

LASIGE, Faculdade de Ciências, Universidade de Lisboa, Portugal
aneves@lasige.di.fc.ul.pt
alamurias@lasige.di.fc.ul.pt
fcouto@di.fc.ul.pt

Abstract. This paper presents our participation on the BioASQ tasks 8a and MESINESP. Our approach was based on X-BERT, a state-of-the-art deep learning algorithm developed for Extreme Multi-Label Classification (XMLC). We adapted this algorithm and its most recent version to the biomedical and multilingual biomedical panorama which, to the best of our knowledge, is the first XMLC-solution to be applied to both tasks. In addition, we combined this algorithm with a named-entity recognition tool to find entities from DeCS and MeSH in the text in order to improve the assignment of labels by the XMLC algorithm. In both challenges, our results in the Micro F-Measure, which was the most relevant evaluation measure, were low. However, our submissions achieved top results in the precision measures, reaching the 1st place in the last two weeks of task 8a in the Micro Precision measure and 2nd place in MESINESP in Micro and Macro Precision, as well as Example Based Precision.

Keywords: Extreme Multi-Label Classification, Deep Learning, Named-Entity Recognition, Semantic Indexing, Multilingual Semantic Indexing

1 Introduction

BioASQ is a series of international challenges focused in promoting the development of state-of-the-art solutions for NLP tasks, namely semantic indexing and question answering, for the biomedical domain. This year, in addition to the usual annual competitions, BioASQ presented a new task called MESINESP (Medical Semantic Indexing in Spanish) focused on the multilingual panorama, in this case, the Spanish language. In this paper, we describe the approach of our submissions for both BioASQ task 8a and BioASQ MESINESP task.

The BioASQ task 8a consisted in indexing English biomedical abstracts with terms from MeSH (Medical Subject Headings) related with the content of biomedical abstracts retrieved from PubMed. The BioASQ MESINESP task was similar, but instead of using MeSH, it used DeCS (Health Sciences Descriptors), a hierarchy of terms closely related to MeSH that can be used to label biomedical articles in Spanish, Portuguese and English. The biomedical abstracts to index were in Spanish and were retrieved from the IBECS and LILACS databases.

In both competitions, each abstract could have more than one term indexed to it, making this a multi-label classification challenge. However, since there are more than 29,000 MeSH terms and more than 33,000 DeCS terms to choose from, our approach consisted in using an Extreme Multi-Label Classification (XMLC) algorithm to classify the abstracts complemented by a Named Entity Recognition tool. XMLC algorithms can provide the best subset of labels to index data, choosing the labels from an extremely large label set, which can reach hundreds of thousands or even millions of labels [1]. Our approach was motivated by three reasons. The first one was the fact that many XMLC algorithms achieved high precision scores in the last years using large datasets with thousands of labels [2–5]. The second reason was the fact that, to the best of our knowledge, state-of-the-art XMLC solutions have not yet been applied to the biomedical domain nor to the multilingual panorama. Finally, the third reason was to develop a XMLC model for both English and Spanish and compare their results. Therefore, we thought that both BioASQ task 8a and task MESINESP would be an appropriate setting to test our approach.

We have chosen the XMLC algorithm X-BERT [5], a state-of-the-art solution that achieved top results in benchmark datasets, surpassing other current XMLC algorithms. Its architecture could also be applied to other languages too, which is essential to the MESINESP task. Furthermore, during the time of the competition, a new version of X-BERT was developed and made available by the authors, which allowed us to adapt and apply it to this challenge. This way, we could compare not only the performance between an English and Spanish model, but also the performance between the older and the newer version of the algorithm.

In addition, the algorithm was also combined with MER (Minimal Named-Entity Recognition) [6, 7], a named-entity recognition tool that given a lexicon and an input text, is able to identify the entities of the lexicon in the text, including their exact location. With this combination, we aimed at improving the assignment of labels by the XMLC algorithm, due to identification of key concepts in the text.

2 Related Work

2.1 Language Models

Recently, many deep learning solutions developed for NLP tasks use pre-trained language models in their core, such as BERT (Bidirectional Encoder Representations for Transformers) [8], which has greatly changed the solutions applied to most NLP tasks. With the success of BERT, many other models based on BERT were developed for specific domains, such as SciBERT [9], a BERT model trained with scientific papers from the corpus of `semanticscholar.org`. SciBERT was developed with the goal of improving the performance of deep learning models in scientific related NLP tasks and, thanks to its characteristics, SciBERT surpassed BERT and achieved state-of-the-art results in several NLP tasks from different scientific domains, such as biomedical sciences.

2.2 Multilingual NLP

Pre-trained language models also constitute an important part of the resources available to develop deep learning models for multilingual NLP task. BERT has also been applied to the multilingual panorama through three different models: BERT Base Multilingual Cased, BERT Base Multilingual Uncased and BERT Base Chinese. The larger model contains 104 languages, and it is composed by the text of the Wikipedia pages for each of those languages, excluding the user and talk pages. These BERT models surpassed the highest accuracy scores achieved in the XNLI dataset [10], a corpus designed for language transfer and cross-lingual sentence classification translated in 15 languages by human experts.

Another great source of multilingual pre-trained models is the Hugging Face Transformers library[1] [11]. Transformers is a Python library that provides several pre-trained deep learning models, such as BERT, for several NLP and language generation tasks. The library is enriched by a vast community of collaborators that adapt the existing models or contribute with novel pre-trained models, many of them in non-English languages. However, multilingual biomedical models are still lacking from this library.

Finally, another example of multilingual improvement, namely in the biomedical domain, is a recent work that aimed at the development of word embeddings for the Spanish biomedical domain [12]. The authors use a combination of data in the Spanish language from the biomedical domain retrieved from the SciELO database, along with text data from specific topics of Wikipedia, such as pharmacy, medicine or biology. When evaluated and compared with embeddings from a much larger but general domain corpus, their embeddings model achieved better performance values, thus showing the importance of domain-specific data.

2.3 Extreme Multi-Label Classification

As to XMLC, several machine learning solutions have been developed in the last decade [1], however, only more recently there have been deep learning approaches developed specifically for this task. One of the first attempts to apply deep learning to the XMLC task was XML-CNN [2], a convolutional neural network that was adapted from a state-of-the-art approach to a multi-class classification task [13]. The architecture of the neural network was adapted with additional layers, one to capture features more precisely from across different regions of text and another to reduce the model size, increasing performance. The loss function was also changed so it could better rank the labels. The XML-CNN

[1] https://github.com/huggingface/transformers

architecture was successful in applying deep learning to XMLC, surpassing most state-of-the-art algorithms in several datasets.

Another successful approach was AttentionXML [3], which used a BiLSTM (bidirectional long short-term memory) recurrent neural network with a multi-label attention layer to capture the most relevant parts of the text. However, AttentionXML could not scale well with the largest datasets, so HAXMLNet was created, adding to the same architecture a hierarchical clustering algorithm to divide the labels into smaller clusters, thus being able to work on larger datasets effectively where AttentionXML failed [4].

Lastly, one of the most recent approaches is X-BERT [5], the first deep learning approach to scale pre-trained language models, such as BERT [8], to XMLC. X-BERT uses a three-stage framework that firstly, semantically indexes all the possible labels in clusters using ELMo [14]. Then, using a deep learning Transformer model, it indexes each text instance to the most relevant cluster. Finally, a linear ranker is trained to rank the labels retrieved from the previous cluster indices, modeling the relevance between each text instance and the retrieved labels, and then calculating the corresponding label scores. X-BERT surpassed other state-of-the-art methods in XMLC using benchmark datasets, such as the Eurlex-4K, AmazonCat-13K or the Wikipedia-500K, all of them available in the Extreme Classification Repository [15].

More recently, a newer version of X-BERT has been released, renamed X-Transformer[2][16]. X-Transformer includes more Transformer models, such as RoBERTa [17] and XLNet [18] and scales them to XMLC. The ranking phase was also improved by combining additional sampling strategies to reduce computational complexity and improve the algorithm performance. It also surpasses the previous results achieved by X-BERT and other XMLC algorithms in the same benchmark datasets. However, at the time of the BioASQ competitions, X-Transformer was still under development, thus we did not have access to official results nor comparisons between this version and other state-of-the-art algorithms.

3 Methods

3.1 X-BERT and X-Transformer modifications

For both tasks, modifications in the algorithm code were required. The first one was made in the vectorization of the labels of the training, test and validation sets. We have chosen to use all possible labels, including the labels that were not present in the train, test or validation sets. This change was needed since the algorithm would fail to work correctly if the number of labels between sets did not match. Another modification was the inclusion of SciBERT in the choices of models to train X-BERT, so that we could use this model in task 8a. However, this inclusion was not made in X-Transformer, since we only used X-Transformer for the MESINESP task.

[2] `https://github.com/OctoberChang/X-Transformer`

X-Transformer required additional changes since, at the time of the competition, the algorithm was still being developed by the X-BERT team and it could only process the four datasets in which X-BERT was tested [5]. To surpass this limitation, we modified some preprocessing scripts from the original X-BERT code and implemented them in X-Transformer, so it could process custom datasets, namely the MESINESP datasets.

Finally, in addition to these modifications, both X-BERT and X-Transformer were also changed so that the algorithms could process input data containing diacritical marks, such as accents, that are common in the Spanish language.

3.2 Pipeline

A common pipeline was developed for both tasks, with some changes according to the algorithm used or to the task, as it can be seen in Figures 1 and 2. The first step, was to retrieve the data given by the competition organizers. A total of 318,658 articles abstracts and titles were used for both task 8a and MESINESP for comparison purposes. The articles were then split into training, test and validation sets using a 60%-20%-20% proportion to be used by X-BERT. The validation set would be used for hyperparameter tuning such as the number of training epochs, since X-BERT can perform an evaluation during training and then save the hyperparameters in a checkpoint if the evaluation results surpassed the results of the previous checkpoint. The values differ when using X-Transformer, because it only requires training and test sets. The test set will also be used by X-Transformer to save the model checkpoints during training evaluation. Therefore, we decided to use a proportion of 70%-30% for X-Transformer. A summary of this distribution can be seen in Table 1.

Table 1. Total number of unique DeCS and MeSH term codes and the total number of articles used in the models along with the corresponding proportion used for the train, test and validation sets. In X-BERT, it was 60%-20%-20%. In X-Transformer it was 70%-30%, since there is no validation set.

System	Total Articles	Train Set	Test Set	Validation Set	#MeSH Terms	#DeCS Terms
X-BERT	318,658	191,195	63,732	63,731	29,640	33,702
X-Transformer		223,060	95,598	-		

The next step was the creation of a vocabulary file, which is required by both X-BERT and X-Transformer and that contains the labels used to classify the data and a corresponding numerical identifier. For that, each line of the vocabulary file has a DeCS or MeSH code, according to the task, and its corresponding internal identifier, which corresponds to a number from 0 to N, where N is the total number of DeCS or MeSH Terms minus 1. This internal numerical identifier is the characteristic that allows X-BERT and X-Transformer to work with any type of labels and with any kind of language, since the algorithms will use these numeric identifiers to classify the text. In addition, a label mapping file was created for each task. For MESINESP, it contains the correspondence between the DeCS term, its code and its numeric identifier in the vocabulary file, while for task 8a, the file has the same structure but using the MeSH terms instead. For example, the term ‘Temefós’, which has the corresponding DeCS code ‘2’, is the first element in the vocabulary file, thus its numeric identifier will be ‘0’. This label mapping file will later be used to map the predictions from their numeric identifiers to the corresponding DeCS or MeSH codes required for the task.

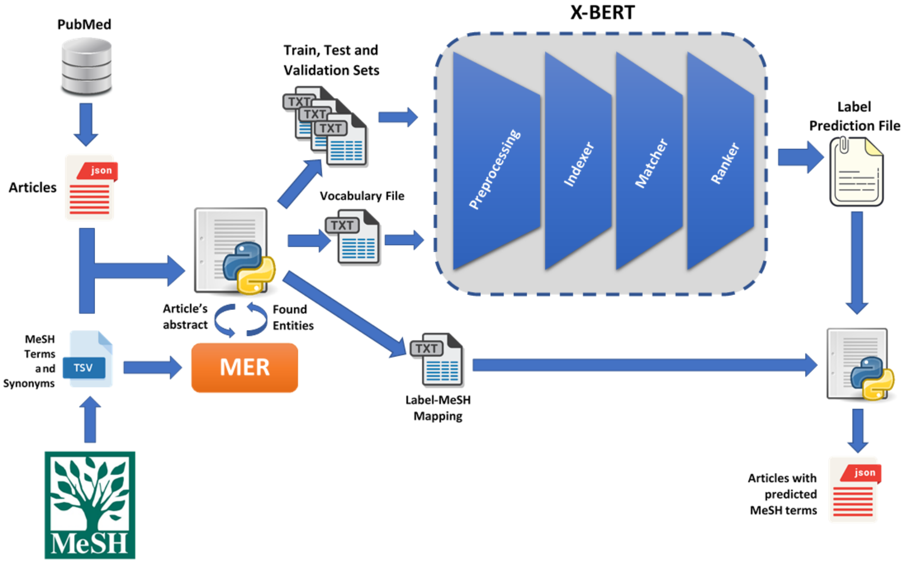

Fig. 1. Developed pipeline to process the articles before running them through X-BERT and finally converting the resulting predictions in the required format for the competition. This pipeline was used for both BioASQ task 8a and task MESINESP.

After converting the codes to numeric identifiers, we used MER [6, 7], which given a lexicon and an input text, is able to identify the entities of the lexicon in the text, including their exact location. For task 8a, we created a lexicon composed by the MeSH terms and their synonyms present in the MeSH 2020 XML file. For task MESINESP, we created a lexicon composed by the DeCS terms and their synonyms, which were given by the competition, but we decided to also include their corresponding direct ancestors as an improvement over the choice made for task 8a. MER was used to recognize the lexicon terms in the article's abstracts in both tasks. The identified terms were then added to the end of the titles or the abstracts, depending on the model that was being trained.

Repeated terms were removed, and the resulting string was then stemmed using a snowball stemmer[3].

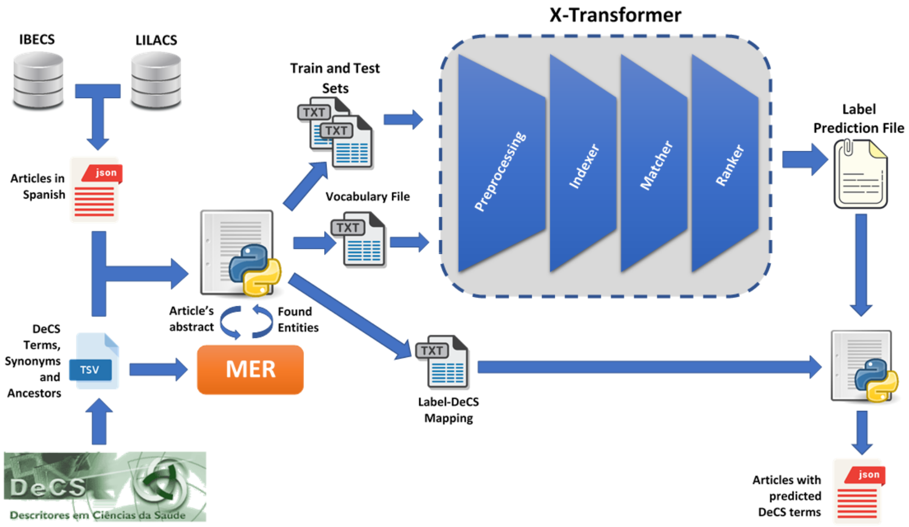

Fig. 2. Pipeline used in the MESINESP task to process the articles before running them through X-Transformer and finally converting the resulting predictions in the required format for the competition.

We chose to train some models using the abstracts and others using the titles in order to compare the performance of the algorithm according to the amount of text given in each article. In cases where the title for a specific article was not present, we used its abstract instead, and in cases where the abstract of a specific article was not present, we used its title instead. In the end, the files to be given as input to X-BERT or X-Transformer, contained, for each article, the list of corresponding DeCS or MeSH terms indexed to that article converted to their corresponding internal identifier, the article's title or abstract, and the terms identified by MER.

The results of X-BERT and X-Transformer are given in the form of sparse matrices, with a number of rows equal to the number of articles that compose the test set, and the columns corresponding to the possible labels. The prediction for each article is retrieved, comprising a top K of most relevant labels and their corresponding confidence values. We used K=20 labels per article for both X-BERT and X-Transformer. For a prediction to be chosen as correct, we discarded every label with a confidence value under a threshold, which will be more detailed in Section 4. Each label is then converted to the corresponding DeCS or MeSH term by using the previously created label mapping file, so that in the end,

[3] https://www.nltk.org/_modules/nltk/stem/snowball.html

the JSON file containing the predictions has the MeSH or DeCS codes for each article id.

In the test sets given by competition organizers, there were no MeSH or DeCS terms indexing the articles, so we had to put a placeholder label on each article, because the code was not prepared to run on unlabeled data. We also had to artificially adapt the size of the given test sets by adding extra articles to them, so that they could have the same size as the ones used on the trained X-BERT or X-Transformer models, otherwise the algorithm would fail to work. In this case, the set needed to have a total of 63,732 articles for X-BERT models and 95,598 articles for X-Transformer models, which corresponds to the number of articles used in the test set of those models. Since in both competitions the number of articles given to classify was under that value, we padded the test set with additional articles equal to the difference. For task 8a, we included indexed PubMed articles from the 14 million articles given by the competition as training data. For MESINESP, we included translated and indexed PubMed articles that came from a larger dataset given as additional material by the task organizers to the participants. The additional articles in both tasks were used as an additional validation set to define the confidence threshold values of our submissions and were not used in neither set to train nor evaluate the models, so that the results would not come biased.

4 Evaluation Script

To evaluate the results achieved by our models, we used a script developed by us to measure the Macro and Micro Precision, Recall and F1 scores, which were some of the measures used in both BioASQ competitions. For each model, two evaluations were made: one using the test set, and another using the additional validation set composed by the articles that were padded to the test sets given by the competition organizers which acted like a preliminary evaluation for the competition.

The evaluation script was focused on choosing the confidence score threshold that achieved the highest Micro F1 (MiF) score and the highest Micro Precision (MiP). The focus on MiF was chosen, since MiF was defined by the organization as the most important measure to both challenges. The focus on MiP was motivated by the improved precision of X-BERT over other XMLC algorithms [5], thus we wanted to see how high it could reach with this type of data.

The confidence score interval given by X-BERT to its predicted labels ranged from 0 to 1. However, in X-Transformer, we noticed that this interval was changed from 0 to 1 to -0.99 to 1, thus encompassing negative confidence values. To facilitate the comparisons between X-Transformer and X-BERT, we have normalized this scale to fit the interval from 0 to 1. Then, to determine the best threshold for the confidence score that achieved the highest MiF or MiP values, we tested the values between 0.01 and 1, incrementing by 0.01.

5 Developed Models

5.1 Titles vs Abstracts

Given the token limit of 512 tokens of the used BERT models, we considered using only the articles or the titles to train the models since that, if we combined both articles and titles along with the entities recognized by MER, in most cases the limit of 512 tokens would be surpassed, thus the articles would be truncated and consequently relevant information and the list of entities found by MER would be lost. Therefore we decided to check which combination could achieve better results: if using the titles and the MER terms or if using the abstracts and the MER terms.

The main hypothesis was that, due to the increased objectiveness of the titles in the core topics of the article in a less amount of words than the abstract, along with a list of key terms and synonyms given by MER, the model would be able to achieve better results since it a had a continuous string of words that were more related to the topics of the article than a larger amount of words with the key topics more diluted. To test the hypothesis, we used X-BERT using the BERT Base Multilingual Cased model and the MESINESP dataset, which was made available first by the organizers. The results are available in Table 2.

Analyzing the results of this experiment, we can see that the difference between using the titles and the abstracts is minimal. If evaluation was focused on achieving the highest MiF value, the model that used the abstracts achieved slightly higher MiP and MiF values. However, if the evaluation was focused on attaining the highest MiP value, the model using the titles achieved better scores in all measures than its abstract counterpart, which confirms our hypothesis.

Table 2. Comparison between the results of two X-BERT models, one trained using the abstracts and the other the titles. Both models were evaluated with a focus on MiF and MiP. The threshold corresponds to the minimum confidence score that obtained the highest MiF or MiP score.
Bold values correspond to the measures where the titles model surpassed the abstracts.

Model	**Focus**	**Measures**						**Threshold**
		MiP	**MiR**	**MiF**	**MaP**	**MaR**	**MaF**	
Abstracts + MER	MiF	0.4780	0.4123	0.4427	0.4766	0.4094	0.4028	0.10
	MiP	0.7117	0.1574	0.2578	0.6812	0.1642	0.2480	0.93
Titles + MER	MiF	0.4685	**0.4142**	0.4397	0.4704	**0.4229**	**0.4072**	0.11
	MiP	**0.7154**	**0.1636**	**0.2663**	**0.6901**	**0.1769**	**0.2617**	0.92

5.2 BioASQ task 8a Models

Only two models were trained for the BioASQ task 8a with the following characteristics:

- Model 1: MER in conjunction with X-BERT, using BERT Base Uncased and finetuned using article's titles.
- Model 2: MER in conjunction with X-BERT, using SciBERT and finetuned using the article's titles.

Both models used the titles due to results shown in the previous section and due to time restrictions, since the competition had already begun when we started training the models, so we could only train 2 models. Therefore, Model 1 was trained for Batch 2 of the competition, while Model 2 was trained for Batch 3.

For both models, the parameters given as default to train and evaluate were kept, except for the eval and train batch sizes which were changed from their corresponding default values of 64 and 32 to 3 due to hardware limitations. The number of train epochs was also changed to 7, since the best checkpoint would usually be achieved before this epoch. The models were trained on a single NVIDIA Tesla P4 GPU, taking 1 week for each model to be fully trained.

5.3 BioASQ task MESINESP Models

In total, four models were trained for the BioASQ MESINESP task with the following characteristics:

- Model 1: MER in conjunction with X-BERT, using BERT Multilingual Base Cased and finetuned using the article's abstracts.
- Model 2: MER in conjunction with X-BERT, using BERT Multilingual Base Cased and finetuned using the article's titles.
- Model 3: X-Transformer using BERT Multilingual Base Cased and finetuned using the article's abstracts.
- Model 4: X-Transformer using BERT Multilingual Base Cased and finetuned using the article's titles.

From these four models, we decided to train two using X-BERT and the other two using X-Transformer since it was made available during the time of this competition. We also decided to train two models using the abstracts and the other using the titles, so that we could not only compare the differences between the two versions of the algorithm, but also to check if our hypothesis from Section 5.1 would also be confirmed in the results achieved in the competition.

For all models, we have kept the default parameters to train and evaluate the models, except for the eval and train batch sizes, due to hardware limitations. In X-BERT they were both changed from their corresponding default values of 64 and 32 to 3. In X-Transformer, both were changed to 4 and we also set the number of gradient accumulation steps to 2 to compensate for the small batch size. The X-BERT models were trained on 7 epochs, since the best checkpoint would usually be achieved before this epoch, like in the BioASQ task 8a models. The X-Transformer models were trained during 3 epochs, the default value. All models were trained on a set of 4 GPUs composed by 1 NVIDIA Tesla P4 GPU and 3 NVIDIA Tesla M10 GPUs. Each model took approximately 4 to 5 days to be fully trained.

6 BioASQ task 8a Results

6.1 Preliminary Evaluation

The results of the preliminary evaluation using the models test sets can be seen in Table 3. When focused on MiF, both Model 1 and Model 2 achieved similar results for their best scoring threshold values, which was the same. However Model 2, the one trained using SciBERT, achieved slightly higher values, which was expected since SciBERT was especially created to be used with scientific text. MiP and Macro Precision (MaP) have the highest values, and both Recall and F1 scores have a relatively similar score and are both higher than 0.50.

When focused on MiP, the threshold values differ, but the results of both models were again similar, with Model 2 achieving slightly higher values in all measures, except for the Macro Recall, which was 0.01 lower compared to Model 1. Both MaP and MiP increased significantly to values over 0.83 and reaching a maximum of 0.89. However, this gain in precision comes with the cost of the recall and F1 values dropping between 0.17 and 0.27.

Table 3. Results achieved by the models in the different measures using their corresponding test sets. The values of each measure correspond to the values achieved by the model using the value present in the Threshold column.
Bold values correspond to the highest achieved scores in each measure.

Model	**Focus**	**Measures**						**Threshold**
		MiP	**MiR**	**MiF**	**MaP**	**MaR**	**MaF**	
Model 1	MiF	0.6179	0.4911	0.5472	0.6168	0.4972	0.5293	0.18
	MiP	0.8722	0.2467	0.3846	0.8650	0.2526	0.3695	0.86
Model 2	MiF	0.6270	**0.5095**	**0.5622**	0.6265	**0.5175**	**0.5460**	0.18
	MiP	**0.8894**	0.2408	0.3790	**0.8832**	0.2480	0.3664	0.89

6.2 Competition Results

We competed in both batch 2 and 3 of BioASQ Task 8a. The results achieved in each week of the batch can be found on Tables 4 and 5, where our scores are compared with the solution that achieved the highest MiF score on the same week.

In batch 2, we submitted results for weeks 3, 4 and 5. The submitted solutions were given by Model 1, the model trained using BERT Base uncased, with the confidence score threshold that achieved the highest MiF score. The results in those weeks were suboptimal, with our model staying in the last position every week and in all evaluation measures.

For batch 3, we submitted results from weeks 2 to 5 using Model 2. In week 2, we used the same strategy of the previous batch, focusing of achieving the highest MiF score. However, the results were very similar, with only a small

Table 4. Results achieved by our submitted models compared to the best scoring model in the Micro F-measure (MiF) in the same week. Only the flat measures are presented.
The bold values correspond to values where the model surpassed the best scoring models and reached the 1st place in the corresponding measure.
Measures: MiF – Micro F-measure. MiP – Micro Precision. MiR – Micro Recall. MaF – Macro F-measure. MaP – Macro Precision. MaR – Macro Recall. Acc. – Accuracy. EBP – Example Based Precision. EBR – Example Based Recall. EBF – Example Based F-measure.

Batch	**Week**	**System**	**Flat Measures**									
			MiF	**MiP**	**MiR**	**MaF**	**MaP**	**MaR**	**Acc.**	**EBP**	**EBR**	**EBF**
2	**3**	deepmesh_dmiip_fdu	0.708	0.758	0.664	0.592	0.721	0.590	0.558	0.755	0.684	0.701
		X-BERT BioASQ (BERT - MiF)	0.316	0.465	0.239	0.066	0.350	0.054	0.184	0.462	0.238	0.294
	4	deepmesh_dmiip_fdu	0.707	0.768	0.655	0.588	0.732	0.584	0.558	0.768	0.673	0.703
		X-BERT BioASQ (BERT - MiF)	0.328	0.496	0.245	0.067	0.352	0.054	0.195	0.498	0.247	0.310
	5	deepmesh_dmiip_fdu_	0.709	0.760	0.663	0.591	0.725	0.591	0.557	0.757	0.678	0.701
		X-BERT BioASQ (BERT - MiF)	0.320	0.477	0.240	0.066	0.343	0.054	0.187	0.480	0.239	0.399
3	**2**	dmiip_fdu	0.769	0.814	0.730	0.678	0.789	0.674	0.647	0.809	0.744	0.762
		X-BERT BioASQ (SciBERT - MiF)	0.330	0.494	0.248	0.067	0.345	0.055	0.195	0.495	0.247	0.309
	3	dmiip_fdu	0.779	0.830	0.735	0.677	0.810	0.669	0.664	0.825	0.751	0.774
		X-BERT BioASQ (SciBERT - MiP)	0.265	0.803	0.159	0.022	0.640	0.015	0.156	0.796	0.162	0.255
	4	deepmesh_dmiip_fdu	0.710	0.771	0.659	0.594	0.729	0.595	0.563	0.769	0.680	0.706
		X-BERT BioASQ (SciBERT - MiP)	0.245	**0.781**	0.145	0.024	0.639	0.017	0.140	0.750	0.145	0.232
	5	deepmesh_dmiip_fdu	0.706	0.758	0.661	0.589	0.714	0.591	0.556	0.756	0.677	0.700
		X-BERT BioASQ (SciBERT - MiP)	0.260	**0.792**	0.155	0.023	0.618	0.016	0.151	**0.772**	0.157	0.248

increase in some precision measures. From week 3 an onward, we decided to started submitting the predictions focusing on the highest MiP value, since it was our best scoring measure in the previous submissions as well as our best scoring measure in the preliminary evaluation. As a result, we achieved lower accuracy, recall and MiF scores, although the loss in MiF was less than 0.1. However, all precision measures increased significantly making the model reach the 3rd, 1st and again 1st place in the MiP measure on weeks 3, 4 and 5, respectively, and a 1st place in the Example Based Precision (EBP) in the final week. The only precision measures where the model did not have a significant increase were the MaP and the Lowest Common Ancestor Precision (LCA-P) measures.

7 BioASQ task MESINESP Results

7.1 Preliminary Evaluation

Contrarily to the task 8a preliminary evaluation, for task MESINESP we added a new evaluation parameter, which was the confidence score threshold of 0.50.

Table 5. Results achieved by our submitted models compared to the best scoring model in the Micro F-measure (MiF) in the same week. Only the hierarchical measures are presented.
Measures: LCA-F – Lowest Common Ancestor F-measure. LCA-P – Lowest Common Ancestor Precision. LCA-R – Lowest Common Ancestor Recall. HiF – Hierarchical F-measure. HiP – Hierarchical Precision. HiR – Hierarchical Recall.

Batch	**Week**	**System**	**Hierarchical Measures**					
			LCA-F	**LCA-P**	**LCA-R**	**HiF**	**HiP**	**HiR**
2	3	deepmesh_dmiip_fdu	0.565	0.608	0.550	0.782	0.839	0.762
		X-BERT BioASQ (BERT - MiF)	0.271	0.422	0.218	0.389	0.619	0.316
	4	deepmesh_dmiip_fdu	0.570	0.624	0.546	0.786	0.855	0.752
		X-BERT BioASQ (BERT - MiF)	0.281	0.448	0.221	0.400	0.652	0.319
	5	deepmesh_dmiip_fdu_	0.567	0.615	0.547	0.785	0.844	0.759
		X-BERT BioASQ (BERT - MiF)	0.273	0.434	0.217	0.394	0.641	0.316
3	2	dmiip_fdu	0.657	0.696	0.641	0.829	0.877	0.809
		X-BERT BioASQ (SciBERT - MiF)	0.280	0.445	0.221	0.400	0.651	0.321
	3	dmiip_fdu	0.676	0.715	0.658	0.839	0.889	0.815
		X-BERT BioASQ (SciBERT - MiP)	0.194	0.526	0.125	0.275	0.867	0.175
	4	deepmesh_dmiip_fdu	0.576	0.627	0.555	0.789	0.852	0.760
		X-BERT BioASQ (SciBERT - MiP)	0.182	0.509	0.115	0.251	0.826	0.157
	5	deepmesh_dmiip_fdu	0.568	0.613	0.550	0.784	0.845	0.758
		X-BERT BioASQ (SciBERT - MiP)	0.189	0.510	0.121	0.262	0.840	0.165

We decided to add this value since it was the middle of the confidence interval scale and it should provide a baseline performance.

The results achieved in the first evaluation using the test set can be seen in Table 6. For both Model 1 and Model 2, the scores achieved by each model were very similar, with slightly higher scores achieved by Model 2. As for Models 3 and 4, we noticed that they achieved higher values than the ones achieved by the older version of the algorithm. Also, and contrarily to Models 1 and 2, Model 3, which used the abstracts, achieved higher values in all measures when compared with Model 4, except when focused on MiP, where Model 4 achieved higher Precision values.

However, in the second evaluation which used the additional validation set, the results achieved were different as can be seen in Table 7. First, there was a drop in the scores, especially in the recall measures thus dropping the F-score. When focusing on MiP, the best performing model was Model 2 and not Model 4 as it was in the first evaluation. This small difference might be due to the usage of MER in Model 2, while Model 4 did not. As to the focus on MiF, the best scoring model was Model 4 with the highest Macro and Micro recall and F-scores.

After this second evaluation, we have decided to submit a total of 4 prediction files, two from a X-BERT model and the other two from a X-Transformer model

Table 6. Results achieved by the models in the different measures using their corresponding test sets. The values of each measure correspond to the values achieved by the model using the threshold value present.
Bold values correspond to the highest achieved scores in each measure.

Model	**Focus**	**Measures**						**Threshold**
		MiP	**MiR**	**MiF**	**MaP**	**MaR**	**MaF**	
Model 1	Base	0.6570	0.2622	0.3748	0.6295	0.2620	0.3386	0.50
	MiF	0.4780	0.4123	0.4427	0.4766	0.4094	0.4028	0.10
	MiP	0.7117	0.1574	0.2578	0.6812	0.1642	0.2480	0.93
Model 2	Base	0.6492	0.2663	0.3777	0.6307	0.2771	0.3512	0.50
	MiF	0.4685	0.4142	0.4397	0.4704	0.4229	0.4072	0.11
	MiP	0.7154	0.1636	0.2663	0.6901	0.1769	0.2617	0.92
Model 3	Base	0.7026	0.2819	0.4024	0.6890	0.2922	0.3773	0.50
	MiF	0.4826	**0.4971**	**0.4898**	0.4830	**0.5046**	**0.4587**	0.24
	MiP	0.7249	0.2539	0.3761	0.7081	0.2652	0.3555	0.55
Model 4	Base	0.6928	0.2785	0.3974	0.6836	0.2984	0.3798	0.50
	MiF	0.5137	0.4318	0.4693	0.5140	0.4513	0.4437	0.28
	MiP	**0.7765**	0.1465	0.2465	**0.7595**	0.1674	0.2567	0.79

Table 7. Results achieved by our models when using the validation set composed by additional indexed articles from PubMed translated to Spanish.
Bold values correspond to the highest achieved values in the corresponding measures.
Gray lines correspond to the models which predictions were submitted to the MESINESP challenge.

Model	**Focus**	**Measures**						**Threshold**
		MiP	**MiR**	**MiF**	**MaP**	**MaR**	**MaF**	
Model 1	Base	0.6132	0.0787	0.1395	0.6117	0.0838	0.1400	0.50
	MiF	0.2937	0.1506	0.1991	0.3163	0.1545	0.1898	0.05
	MiP	0.6737	0.0550	0.1018	0.6627	0.0603	0.1082	0.97
Model 2	Base	0.6290	0.0759	0.1355	0.6323	0.0814	0.1377	0.50
	MiF	0.3029	0.1572	0.2070	0.3204	0.1608	0.1975	0.05
	MiP	**0.6903**	0.0556	0.1028	**0.6823**	0.0613	0.1103	0.95
Model 3	Base	0.5610	0.0824	0.1437	0.5848	0.0886	0.1437	0.50
	MiF	0.3198	0.1730	0.2245	0.3314	0.1797	0.2143	0.15
	MiP	0.6468	0.0489	0.0910	0.6471	0.0550	0.0998	0.99
Model 4	Base	0.6448	0.0832	0.1473	0.6374	0.0896	0.1499	0.50
	MiF	0.3786	**0.2041**	**0.2652**	0.3761	**0.2107**	**0.2506**	0.14
	MiP	0.6823	0.0549	0.1017	0.6752	0.0613	0.1103	0.77

so that the performance of the models could be compared in the competition. The results of Model 2 focusing on MiF and MiP were chosen, since the focus on MiP presented the highest precision scores, and the focus on MiF had better recall and F-scores than Model 1. From Model 4, the results focusing on MiF and the baseline were chosen, since the focus on MiF had the achieved the highest recall and F1 score values, and the baseline since the precision gains from the MiP focus were not so significant and had heavier losses on the remaining measures.

7.2 Competition Results

In Table 8, we present the results achieved by our four submissions, the BioASQ baseline and the winning system. Like in BioASQ task 8a, our results in the MiF measure were not optimal, with the four submissions staying in bottom positions in the classification. However, in the precision measures, the scores achieved by two of our submissions were higher than most systems, even surpassing the winning system.

The predictions of Model 4 focused on the baseline achieved the 2nd place in the MiP, MaP and EBP measures, while the predictions of Model 2 focused on MiP achieved the 3rd place in MiP as well as a 5th place in EBP. As to the models focused of MiF, they achieved balanced scores between all measures, but they were not enough to surpass the BioASQ baseline. We can also notice that the submissions focused on MiF achieved higher accuracy scores, which was expected.

Table 8. Results achieved by the models submitted for the MESINESP task (gray lines) compared to the best scoring system in the Micro F-measure (MiF) and the BioASQ baseline.
Bold values correspond to the measures where our submissions surpassed both the BioASQ baseline and the winning system.
Measures: MiF – Micro F-measure. MiP – Micro Precision. MiR – Micro Recall. MaF – Macro F-measure. MaP – Macro Precision. MaR – Macro Recall. Acc. – Accuracy. EBP – Example Based Precision. EBR – Example Based Recall. EBF – Example Based F-measure.

System	**Flat Measures**									
	MiF	**MiP**	**MiR**	**MaF**	**MaP**	**MaR**	**Acc.**	**EBP**	**EBR**	**EBF**
Winning System	0.4254	0.4374	0.4140	0.3194	0.3989	0.3380	0.2786	0.4382	0.4343	0.4240
BioASQ Baseline	0.2695	0.2337	0.3182	0.2816	0.3733	0.3220	0.1659	0.2681	0.3239	0.2754
X-BERT BioASQ F1 (Model 2 focus MiF)	0.1430	**0.4577**	0.0847	0.0220	0.3095	0.0186	0.0787	**0.5057**	0.0867	0.1397
X-BERT BioASQ (Model 2 focus MiP)	0.0909	**0.5449**	0.0496	0.0045	0.3422	0.0036	0.0503	**0.5415**	0.0508	0.0916
LasigeBioTM TXMC F1 (Model 4 focus MiF)	0.2507	0.3559	0.1936	0.0858	0.3646	0.0799	0.1440	0.3641	0.1986	0.2380
LasigeBioTM TXMC P (Model 4 focus baseline)	0.1271	**0.6864**	0.0701	0.0104	**0.6989**	0.0081	0.0708	**0.6609**	0.0716	0.1261

8 Discussion

8.1 Achieved Results

Overall, the results achieved in both competitions showed that the usage of deep learning XMLC solutions in the biomedical and multilingual panorama can achieve promising results, especially with the high MiP values achieved in MESINESP and in the last three weeks of task 8a. However, there is also a great room for improvement. For example, in both task 8a and in MESINESP, the models struggled in all recall measures, with scores lower than most of the competing systems, although achieving higher scores in the precision measures. We think that these low recall values can possibly be explained by several factors.

The amount of data used to train the models could have impacted our results. We have chosen to use the same number of articles in both competitions, which was the number of available articles in the pre-processed train set from task MESINESP, so that we could compare the performance of a biomedical XLMC based model across multiple languages. There was much more data available for both task 8a and task MESINESP that we could have used to train larger models. However, this was not possible to do in time due to the hardware limitations, which lead to each model requiring approximately 1 week to train.

Another limitation was the fact that we did not use more complex lexicons in MER to identify entities in the text. The lexicon we used in task 8a was rather simple, only comprising the MeSH terms and their synonyms. We could have used more complex lexicons such as one with the term's ancestors or with ontology information, which could possibly increase the final scores. In task MESINESP, the fact that we did not use MER in the X-Transformer models could have negatively influenced our results.

Finally, another reason for our low recall values could be caused by the fact that XMLC algorithms tend to achieve higher precision values due to an increased focus on precision measures, namely in Precision at top K (P@K). P@K is widely used in the evaluation and comparison between XMLC algorithms [1–5, 15, 16], as it considers the number of most relevant labels retrieved by the algorithm among the top K documents. The choice of using this measure to evaluate the performance of these algorithms makes sense when dealing with datasets with thousands of documents, from which the objective is to retrieve the most relevant labels from a set of thousands or millions of possible labels. However, this increased focus in precision can affect negatively the evaluation of the systems in other measures, such as the F1 score and recall, which were evaluated in these two tasks.

8.2 English vs Multilingual

One of our objectives with our participation in these two tasks was to compare the performance of an English model versus its multilingual counterpart in a biomedical domain. Comparing the results achieved by our submissions in BioASQ task 8a and in task MESINESP, we notice that there is a great difference between the performances of an English model and a multilingual model. In task 8a, Model 1 and 2 shared the same characteristics as task MESINESP Model 2. The three of them were X-BERT models trained with the same number of articles, all used the titles and were combined with MER to find key terms in the abstracts. The major difference between them was the pre-trained language models used. In task 8a, the models used BERT Base Uncased and SciBERT, while in task MESINESP the model used BERT Base Multilingual Cased.

Table 9. Results of the different versions of the common Model (Article's titles + terms identified by MER) used in the BioASQ competitions, when evaluated using the corresponding validation set.
Bold values correspond to the highest values achieved in the corresponding measures.

Model	**Focus**	**Measures**						**Threshold**
		MiP	**MiR**	**MiF**	**MaP**	**MaR**	**MaF**	
BERT Base Uncased	MiF	0.6179	0.4911	0.5472	0.6168	0.4973	0.5293	0.18
	MiP	0.8722	0.2467	0.3846	0.8650	0.2526	0.3695	0.86
SciBERT	MiF	0.6270	**0.5095**	**0.5622**	0.6265	**0.5175**	**0.5460**	0.18
	MiP	**0.8894**	0.2408	0.3790	**0.8832**	0.2480	0.3665	0.89
BERT Multilingual	MiF	0.4685	0.4142	0.4397	0.4704	0.4229	0.4072	0.11
	MiP	0.7154	0.1636	0.2663	0.6901	0.1769	0.2617	0.92

Analyzing the scores achieved by these three models, we can notice that there is a significant difference in the scores achieved in both our evaluations and in the competition evaluations. The clearest is that in task 8a evaluation, our worst scores in the competition surpass our best submission in task MESINESP in the non-precision measures, as well as the enormous difference in the scores between the first classified systems in task 8a and the MESINESP winner. The reason for this difference can be caused by the number of documents to classify in each task, which in MESINESP was more than 24,000 articles in total, whereas in task 8a it was about 5,000 to 7,000 articles in each week.

However, in our evaluations using the model's test set, we can also see that the English-based models achieve better results in all measures, as can be seen in Table 9. The reason for the difference in the results of the evaluations is probably caused by the amount of data used to train the pre-trained language models. The BERT models were trained using mostly text data retrieved from Wikipedia. However, the amount of Wikipedia text varies between different languages. For example, in terms of words, the BERT Base model was trained with more than 3,000 million words, with about 2,500 million coming from the English Wikipedia [8]. In comparison, the number of words in the Spanish Wikipedia, as of June 2020, was less than 900 million words. This enormous gap between the number of words is very significant and consequently is reflected on the results achieved by the non-English models.

8.3 Future Work

We could not submit in time a X-Transformer model that used MER in the abstracts to identify relevant keywords. Therefore, in the future we will develop a model using X-Transformer in conjunction with MER for both the English language and the Spanish language and check if the usage of MER with X-Transformer can improve the results in the different evaluation measures. In addition, we intend to implement additional terms to the MER lexicon and use other ontologies other than DeCS and MeSH. Afterwards, we can also run MER in both titles and abstracts, since the titles might contain relevant keywords that are not present in the abstracts.

We could also have developed a model using BioBERT [19], a BERT model developed with English biomedical text for NLP tasks, but due to time and hardware constrains, we were not able to adapt and train a model using BioBERT, which could have lead to improved results in task 8a. In the future, we intend to develop a X-BERT and a X-Transformer model using BioBERT and compare its performance with the models trained using SciBERT.

Finally, as a way of further improving the results, we could use semantic similarity solutions to find which labels are more related with the abstract, thus hoping to reduce the number of false negatives and consequently improve the recall and F-measures.

9 Conclusion

With our participation in these two competitions, we successfully adapted and applied a state-of-the-art deep learning XMLC solution to the biomedical and multilingual biomedical panorama. As far as we know, deep learning XMLC-based solutions had not yet been applied to both these scenarios. The results achieved in the competitions were promising, with our submissions achieving top classifications in the precision measures when compared with most competing models. Although our results were not optimal, we achieved a 1st place in MiP in the last two weeks of task 8a and 2nd place in MiP, MaP and EBP in task MESINESP. However, the performance of multilingual models is still far from the performance of English models in NLP tasks, especially in the biomedical domain. Therefore, we can conclude that there is a great need for additional research and development of non-English deep learning models and corpora, especially for specific domains such as the biomedical sciences.

10 Acknowledgements

This project was supported by FCT through funding of the DeST: Deep Semantic Tagger project, ref. PTDC/CCI-BIO/28685/2017, and the LASIGE Research Unit, ref. UIDB/00408/2020

References

1. Bhatia, K., Jain, H., Kar, P., Varma, M., and Jain, P.: Sparse local embeddings for extreme multi-label classification. Advances in Neural Information Processing Systems. pp. 730–738. (2015)
2. Liu, J., Chang, W. C., Wu, Y., and Yang, Y.: Deep learning for extreme multi-label text classification. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17). pp. 115—124. Association for Computing Machinery, New York, NY, USA. (2017) https://doi.org/10.1145/3077136.3080834

3. You, R., Zhang, Z., Wang, Z., Dai, S., Mamitsuka, H., and Zhu, S. : AttentionXML: Label Tree-based Attention-Aware Deep Model for High-Performance Extreme Multi-Label Text Classification. Retrieved from: `http://arxiv.org/abs/1811.01727` (2018)
4. You, R., Zhang, Z., Dai, S., and Zhu, S.: HAXMLNet: Hierarchical Attention Network for Extreme Multi-Label Text Classification. Retrieved from: `http://arxiv.org/abs/1904.12578` (2019)
5. Chang, W. C., Yu, H. F., Zhong, K., Yang, Y., and Dhillon, I.: X-BERT: eXtreme Multi-label Text Classification with BERT. Retrieved from: `https://arxiv.org/abs/1905.02331v2` (2019)
6. Couto, F. M., and Lamurias, A.: MER: a shell script and annotation server for minimal named entity recognition and linking. J Cheminform **10** (58), (2018) https://doi.org/10.1186/s13321-018-0312-9
7. Couto, F. M., Campos, L. F., and Lamurias, A.: MER: a Minimal Named-Entity Recognition Tagger and Annotation Server. In: Proceedings of the BioCreative V.5 Challenge Evaluation Workshop. pp. 130-137. (2017)
8. Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Retrieved from: `http://arxiv.org/abs/1810.04805` (2018)
9. Beltagy, I., Lo, K., and Cohan, A.: SciBERT: A Pretrained Language Model for Scientific Text. Retrieved from: `http://arxiv.org/abs/1903.10676` (2019)
10. Conneau, A., Rinott, R., Lample, G., Williams, A., Bowman, S., Schwenk, H., and Stoyanov, V.: XNLI: Evaluating Cross-lingual Sentence Representations. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 2475–2485. Association for Computational Linguistics, Brussels, Belgium. (2018) https://doi.org/10.18653/v1/d18-1269
11. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., and Brew, J.: HuggingFace's Transformers: State-of-the-art Natural Language Processing. Retrieved from: `http://arxiv.org/abs/1910.03771` (2019)
12. Soares, F., Villegas, M., Gonzalez-Agirre, A., Krallinger, M., and Armengol-Estapé, J.: Medical Word Embeddings for Spanish: Development and Evaluation. In: Proceedings Ofthe 2nd Clinical Natural Language Processing Workshop. pp. 124–133. Association for Computational Linguistics, Minneapolis, Minnesota, USA. (2019) https://doi.org/10.18653/v1/W19-1916
13. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751. Association for Computational Linguistics, Doha, Qatar. (2014) https://doi.org/10.3115/v1/d14-1181
14. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L.: Deep Contextualized Word Representations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 2227–2237. Association for Computational Linguistics, New Orleans, Louisiana, USA. (2018) https://doi.org/10.18653/v1/n18-1202
15. The extreme classification repository: Multi-label datasets & code. `http://manikvarma.org/downloads/XC/XMLRepository.html`. Last accessed 30 Jun 2020
16. Chang, W. C., Yu, H. F., Zhong, K., Yang, Y., and Dhillon, I.: Taming Pretrained Transformers for Extreme Multi-label Text Classification. Retrieved from: `http://arxiv.org/abs/1905.02331v4` (2020)

17. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanovand, V.: RoBERTa: A Robustly Optimized BERT Pretraining Approach. Retrieved from: `http://arxiv.org/abs/1907.11692` (2019)
18. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V.: XL-Net: Generalized Autoregressive Pretraining for Language Understanding. Retrieved from: `http://arxiv.org/abs/1906.08237` (2019)
19. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J.: BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics **36** (4), pp. 1234—1240. (2020) https://doi.org/10.1093/bioinformatics/btz682