LasigeBioTM at CANTEMIST: Named Entity Recognition and Normalization of Tumour Morphology Entities and Clinical Coding of Spanish Health-related Documents

Pedro Ruas[a], Andre Neves[a], Vitor D.T. Andrade[a] and Francisco M. Couto[a]

[a] *LASIGE, Faculdade de Ciências da Universidade de Lisboa, Portugal*

Abstract

The CANTEMIST track included three subtasks for the automatic assignment of codes related with tumour morphology entities to Spanish health-related documents: CANTEMIST-NER, CANTEMIST-NORM and CANTEMIST-CODING. For CANTEMIST-NER, we trained Spanish biomedical Flair embeddings on PubMed abstracts and then trained a BiLSTM+CRF Named Entity Recognition tagger on the CANTEMIST corpus using the trained embeddings. For CANTEMIST-NORM, we adapted a graph-based model that uses the Personalized PageRank algorithm to rank the eCIE-O-3.1 candidates for each entity mention. As for CANTEMIST-CODING, we adapted X-Transformer, a state-of-the-art deep learning Extreme Multi-Label Classification algorithm, to classify the clinical cases with a ranked list of eCIE-O-3.1 terms in a multilingual and biomedical panorama. The results obtained were a F1-score of 0.749 and 0.069 for the CANTEMIST-NER and the CANTEMIST-NORM subtasks, respectively, and our best scoring submission achieved a MAP score of 0.506 in the CANTEMIST-CODING subtask.

Keywords

CANTEMIST, Named Entity Recognition, Normalization, Coding, Text Mining, Natural Language Processing, Clinical Text, Extreme Multi-Label Classification

1. Introduction

There are several benefits arising from the application of Natural Language Processing (NLP)/Text Mining approaches to clinical text, like for example, the improvement of the decision-making process in clinical context. The use of electronic health records is associated with less doctor-patient interaction [1], so tools that are able to automatically extract relevant information from clinical notes can free up the doctors to contact directly with patients. Besides, these tools have the potential to improve biomedical [2, 3] and pharmaceutical research [4] and to democratise the access to clinical information for the layman user [5].

In the present work, we describe the participation of the LasigeBioTM team in CANTEMIST ("CANcer TExt Mining Shared Task – tumor named entity recognition") competition [6], which

Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020)

EMAIL: psruas@fc.ul.pt (P. Ruas); aneves@lasige.di.fc.ul.pt (A. Neves); fc49005@alunos.fc.ul.pt (V.D.T. Andrade); fcouto@di.fc.ul.pt (F.M. Couto)
ORCID: 0000-0002-1293-4199 (P. Ruas); 0000-0003-0627-1496 (F.M. Couto)

included a corpus of Spanish health-related documents and three different subtasks with the following goals:

- CANTEMIST-NER: Automatically recognise and locate tumour morphology mentions.
- CANTEMIST-NORM: Returning and normalising all tumour morphology mentions along with their respective codes from the eCIE-O-3.1 ("Clasificación Internacional de Enfermedades para Oncología - 3ª edición, 1ª revisión"[1]) terminology.
- CANTEMIST-CODING: Classification of clinical cases by returning a list of ranked eCIE-O-3.1 codes for each document.

For the CANTEMIST-NER substask we used the Flair framework [7] to train new Flair embeddings over Spanish translated PubMed abstracts and to train a NER tagger with BiLSTM+CRF architecture on the CANTEMIST Corpus leveraging the trained embeddings. For the CANTEMIST-NORM substask, we used the PPR-SSM model [8] to normalise the entities recognised by the NER tagger. This model builds a disambiguation graph for each document, where the nodes are the retrieved candidate codes from the eCIE-O-3.1 terminology for the present entities and the relations are based on the hierarchy of eCIE-O-3.1 terminology. A variation of this model additionally retrieves candidates from other terminologies, like CIE-10-ES and DeCS, and extracts relations between concepts from these terminologies and the codes from the eCIE-O-3.1 terminology to improve the edge structure in the graph. The Personalised PageRank algorithm (PPR) assign weights to each candidate and the highest scored one is the selected code for the respective entity. As for the CANTEMIST-CODING subtask, we adapted and built a pipeline using X-Transformer, a deep learning Extreme Multi-Label Classification (XMLC) algorithm, to the multilingual biomedical panorama, so that it could successfully process and classify each clinical case with the eCIE-O-3.1 terms more related with each document.

1.1. Related work

In the Named Entity Recognition (NER) task, state-of-the-art approaches usually have a BiLSTM-CRF architecture, which was initially proposed by Huang et al. [9]. LSTM (Long-Short Term Memory) networks are Recurrent Neural Networks (RNN), which means that these networks have a recurrent layer connecting different features at different time frames. In fact, BiLSTM networks can leverage the past features and the future features for a given time frame. An input layer represents a given set of features, in this case text tokens, at a given time and the output layer represents the probability distribution for each label at that time. The CRF (Conditional Random Fields) models, in turn, focus on sentence level tag information. More recently, pre-trained language models, like BERT [10] or ELMo [11], have been fine-tuned to the NER task. BERT has a multilayer bidirectional transformer encoder architecture and, contrarily to RNNs, employs an attention mechanism to establish the dependency between the input features and the output. The original BERT implementation has been trained in general corpora, such as the BookCorpus and the English Wikipedia, but since then a plethora of domain-specific versions have been proposed, including BioBERT [12] and ClinicalBERT [13].

[1] https://eciemaps.mscbs.gob.es/ecieMaps/browser/index_o_3.html

The state-of-the-art approaches in Entity Normalization (also called Disambiguation or Linking) include graph-based models [14, 15], neural networks-based models [16, 17] and, similarly to what happens for the NER task, more recently the fine-tuned pre-trained language models [18, 19]. The graph-based models usually focus on building a graph containing the candidates for the entity mentions and then on ranking the candidates according to the relevance or coherence of each candidate in the graph. These are global models, since the disambiguation decision of a given entity mention is dependant of the other disambiguations in the same graph, but there is also sometimes a module responsible for the determination of the local similarity between candidates and mentions or the candidate retrieval. The neural network-based models usually take into account both the global coherence of the candidates and the local similarity, however, the candidates and the entities are typically represented by word embeddings, which are then integrated in the neural network. On the other hand, BERT-based models like the one proposed by Ji et al. [18] focus on the generation of contextualised word embeddings for candidates and entities and then on candidate ranking, which is considered a sequence-pair classification task. In this case, the model performs disambiguation of each entity independently based on word representations and other local features.

As to XMLC, several machine learning solutions have been developed in the last decade [20], but only more recently there have been deep learning solutions applied to XMLC. One of the first attempts to was the XML-CNN [21], a convolutional neural network that was adapted from a state-of-the-art approach to a multi-class classification task [22], with some changes on the neural network layers that allowed it to capture features more precisely from different regions of text. There was also HAXMLNet [23] which used a BiLSTM RNN with a multi-label attention layer to capture the most relevant parts of the text, along with a hierarchical clustering algorithm to divide labels through clusters, which proved efficient on larger datasets. Lastly, there is X-Transformer [24] the first deep learning approach to scale pre-trained Transformer models, such as BERT [10], RoBERTa[25] or XLNet[26] to XMLC. The algorithm uses a three-stage framework that firstly, semantically indexes all the possible labels in clusters using ELMo [11]. Then, using a deep learning Transformer model, it indexes each text instance to the most relevant cluster and, finally, ranks the labels retrieved from the previous cluster indices. X-Transformer surpassed other state-of-the-art methods in XMLC in four benchmark datasets and it was also applied to a query recommendation dataset from Amazon, where it showed improvements of more than 10% over Parabel [27], one of the most commonly used and competitive XMLC algorithms.

Our team has already made an adaptation of X-Transformer for the biomedical panorama in the BioASQ MESINESP competition that occurred earlier this year. In MESINESP, the goal was indexing a large dataset of biomedical articles written in Spanish using DeCS terms. In the final scoreboard, our approach using X-Transformer has achieved high scores in the precision measures, surpassing most competing systems in those measures.

2. Methodology

2.1. CANTEMIST-NER

The goal of this task was to recognise and locate tumour morphology entities in Spanish health-related documents. We used the Flair framework [7] to develop a Spanish biomedical NER tagger.

2.1.1. Training of Flair embeddings

We trained new Flair contextualised embeddings [28] in Spanish biomedical text, more concretely, in translated abstracts of PubMed articles available at https://temu.bsc.es/mesinesp/index.php/download/translated-pubmed-articles/. We considered 4 subsets of articles, each one with 80%/10%/10% of the articles included in the train, validation and test files, respectively:

1. 32,500 articles, 40,987,614 tokens
2. 32,500 articles, 35,352,727 tokens
3. 32,500 articles, 39,021,229 tokens
4. 32,500 articles, 40,005,075 tokens

The 4 splits contained a total of 130,000 articles and 155,366,645 tokens, of which 143,387,385 corresponded to training tokens.

We generated a language model for each split with hidden_size = 1024, nlayers = 1, dropout = 0.1, using the following training parameters:

- sequence_lenght = 250
- mini_batch_size = 100
- max_epochs = 2000
- patience = 25

We trained forward and backward embeddings in each split using a single NVIDIA Tesla P4 GPU, since Flair does not have multi-GPU support in the current version, and interrupted the training after a variable number of epochs that ranged from 71 in the backward embeddings training in split 1 and 99 in the backward embeddings training in split 2.

2.1.2. Pre-processing of the CANTEMIST corpus

We converted the train, development 1 (dev1) and development 2 (dev2) sets of the CANTEMIST corpus[2] to the IOB2 format[3] using the Flair sentence segmenter jointly with the Flair tokenizer. Each token was tagged with the label “B-MOR_NEO" if corresponded to the beginning of an annotation, the label “I-MOR_NEO" if corresponded to the inside of an annotation, and the label “O" if it was outside of any annotation. The content present in the train, dev1 and dev2 sets originated, respectively, the files “train.txt", “dev.txt" and “test.txt". The corpus was then loaded into a Flair “ColumnCorpus" object to allow the further training of the NER tagger.

[2] https://temu.bsc.es/cantemist/?p=4338

[3] https://en.wikipedia.org/wiki/Inside%E2%80%93outside%E2%80%93beginning_(tagging)

2.1.3. Training of the Spanish biomedical NER tagger

The following models were considered:

1. “base": uses Flair embeddings (es-forward and es-backward) trained on Spanish Wikipedia + Spanish FastText embeddings.
2. “medium": uses Flair embeddings (es-forward and es-backward) trained on Spanish Wikipedia + Spanish FastText embeddings + PubMed Flair embeddings trained in 1 PubMed split.
3. “large": uses Flair embeddings (es-forward and es-backward) trained on Spanish Wikipedia + Spanish FastText embeddings + PubMed Flair embeddings trained in 2 PubMed splits.
4. “pubmed": uses PubMed Flair embeddings trained in 4 PubMed splits.

We considered the default architecture for the sequence tagger: BiLSTM with a CRF decoding layer, hidden_size = 256. The training parameters were set to:

- learning_rate = 0.1
- mini_batch_size = 32
- max_epochs = 55
- patience = 3

Due to the lack of time, we only selected the “medium" model for training, as we considered it was the safest approach: to leverage trained biomedical embeddings jointly with general available embeddings. After the training, we applied the model to predict the labels in the 5232 documents belonging to the background + test set of the CANTEMIST corpus and to create an annotation file in the BRAT format for each text document.

2.2. CANTEMIST-NORM

The goal of this task was to perform NER and the normalization or disambiguation of the recognised entities to the eCIE-O-3.1 terminology. We applied the model previously developed for the NER task to recognise the entities and adapted the PPR-SSM model [8] to assign the entities a eCIE-O-3.1 code.

2.2.1. Pre-processing of the NER output

In each document, the first step was to apply the NER tagger to generate the NER output files and then to retrieve the ten best eCIE-O-3.1 candidates for each recognised entity through string matching, more concretely, according to the edit distance. The model then built a disambiguation graph with the eCIE-O-3.1 candidates for all present entities in the document. Two candidates/nodes were considered linked in the graph if they were linked in the eCIE-O-3.1 hierarchy. For each candidate was calculated the extrinsic information content (IC), which is a measure of rareness: the IC of a given entity is high if that entity has few entries in an external dataset [29]. In this case, we considered the external dataset the train, dev1 and dev2 sets of the CANTEMIST corpus.

2.2.2. Entity disambiguation

The model applied the Personalized PageRank (PPR) algorithm over each disambiguation graph. PPR is a variation of PageRank [30], which was originally proposed as an algorithm to rank the relative importance of web pages. It considers the web a graph where each node is a page and has links to other pages (forward links) and links from other pages (backlinks). The PageRank algorithm simulates the behaviour of a "random surfer" in the web: from a given page the surfer can either follow one of the forward links in that page or jump to a random page belonging to the graph. In the personalised variation, this jump is not random and instead always occur to a chosen page. After successive iterations, the algorithm returns the probability distribution of reaching each node in the graph. Nodes containing more links will be reached more times, so they will have more relevance in the context of the graph. PPR have also been applied in the normalization of entities, but in this case the web graph is replaced by the disambiguation graph containing the candidates for all entities in a given document. PPR traverses the graph and then assigns weights to each candidate according to its coherence or relevance to the graph: more connected nodes will have higher weight. Additionally, in our model, the IC of each candidate was also considered in the candidate ranking. The model selected the highest scored candidate for each entity and added the eCIE-O-3.1 codes to the annotation files outputted by the NER tagger.

2.2.3. Multiple terminologies

We also explored the use of more than one terminology in the candidate retrieval and graph building. We considered the CIE-10-ES (“Classificación Internacional de Enfermedades 10.ª Revisión, Modificación Clínica"[4]) and the Spanish DeCS (“Descriptores en Ciencias de la Salud")[5] terminologies. For each entity, besides the ten best eCIE-O-3.1 candidates, we also retrieved the five best CIE-10-ES and the five best DeCS candidates through string matching and built the disambiguation graph accordingly. We considered that a eCIE-O-3.1 candidate and a CIE-10-ES or a DeCS candidate were linked if they were present in the same sentence of any document belonging to the CANTEMIST corpus. We applied the python implementation[6] of MER [31] for the fast recognition of the entities in the sentences. With these additional candidates, the disambiguation graph was denser and contained more semantic information, which we expected that would improve the precision of the disambiguation process. After the application of the PPR algorithm, the model only selected eCIE-O-3.1 candidates to normalise the mentions.

2.2.4. Models

The following models were considered:

1. “single-ont": this model only used candidates retrieved from the eCIE-O-3.1 terminology.
2. “multi-ont": besides eCIE-O-3.1, this model additionally retrieved candidates from CIE-10-ES and DeCS terminologies to improve the disambiguation graph.

[4] https://eciemaps.mscbs.gob.es/ecieMaps/browser/index_10_mc.html
[5] http://decs.bvs.br/E/homepagee.htm
[6] https://pypi.org/project/merpy/

2.3. CANTEMIST-CODING

The goal of this task was the classification of clinical cases in Spanish by returning a list of ranked eCIE-O codes related to the content of each clinical case. To tackle this challenge, we decided to use X-Transformer, a state-of-the-art deep learning XMLC solution and apply it to multilingual biomedical panorama.

2.3.1. X-Transformer modifications

Some modifications in the algorithm code were required. The first one was made in the vectorization of the labels of the training and test sets. We have chosen to use all possible labels, including the labels that were not present in the train or test 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 BETO [32][7] in the choices of models to train X-Transformer, since we considered that using a Transformer model specifically designed for the Spanish language could lead to improved results over the Multilingual version of BERT. Finally, we have also adapted the algorithm so that it could process input data containing diacritical marks, such as accents, that are common in the Spanish language.

2.3.2. Pipeline

A pipeline was developed for this task as it can be seen in Figure 1. After retrieving the data from the competition organisers, the first step of this pipeline was merging each separate text file that composed the training and development datasets into a single file for each dataset, so that in the end we could have only two files, one for train and another for test. Then, using the '.ann' files given for the other two tasks of the CANTEMIST competition and that were associated with each clinical case, we extracted all labels that were attributed to each document and appended them to the beginning of the corresponding clinical case separated by a tab character ('\t'). This way, X-Transformer could distinguish between the labels and the text. The text was then stemmed using a Snowball stemmer[8].

The next step was creating a vocabulary file containing all labels used in the datasets, which was required as input by X-Transformer. Each line of this file had an eCIE-O code and its corresponding internal identifier, which corresponds to a number from 0 to N, where N is the total number of eCIE-O codes minus 1. This internal identifier is a label standardization method that allows X-Transformer to classify the text using labels from any kind or domain. For the creation of this file, we used a file containing a list of valid eCIE-O codes that was provided by the competition in their evaluation scripts folder, from which we retrieved all codes and corresponding descriptions. Then, we included the eCIE-O codes descriptions that were present in the '.ann' files and that were not present in the list of codes retrieved, adding them to existing descriptions if they had differences. Then, the codes without any description were removed, since they would be of no use to classify the clinical cases using X-Transformer. In the end, our vocabulary file consisted of a total of 4360 eCIE-O codes.

[7] https://github.com/dccuchile/beto

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

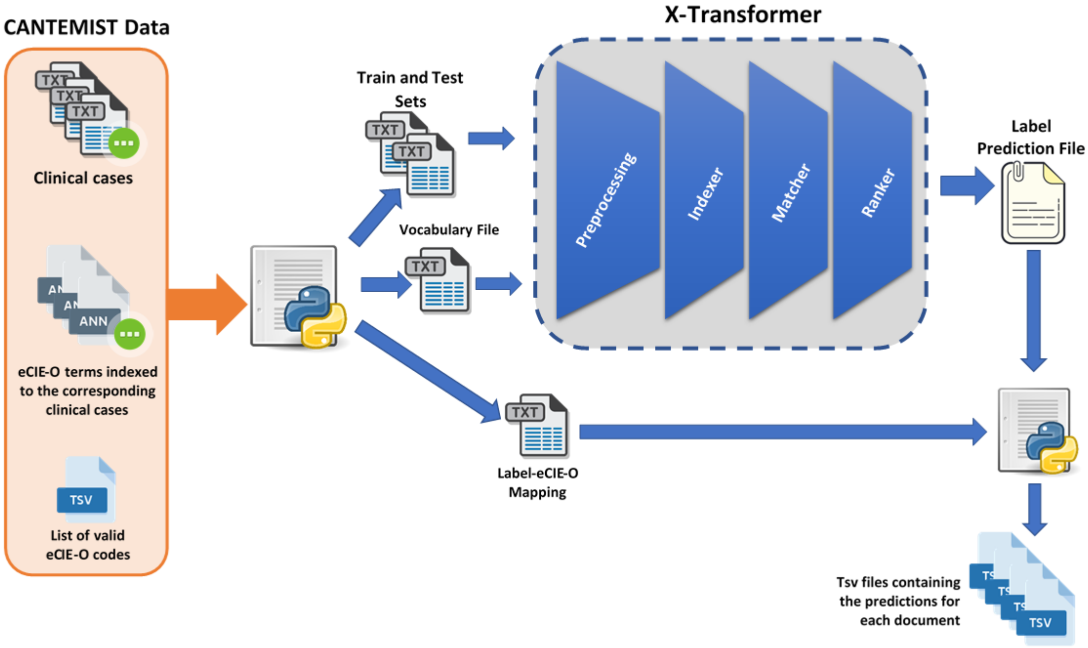

Figure 1: Developed pipeline for CANTEMIST-CODING task. It processes the CANTEMIST data before running it through X-Transformer. In the end, the predictions are converted to the format required by the competition.

In addition, we have also created a label mapping file that contained the correspondence between the eCIE-O code and its numeric identifier in the vocabulary file. For example, the term ‘Células tumorales benignas’, which has the corresponding eCIE-O code ‘8001/0’, is the eleventh element in the vocabulary file, thus it’s numeric identifier will be ‘10’. This label mapping file will later be used to map the predictions from their numeric identifiers to the corresponding eCIE-O codes required for the task.

The results of X-Transformer are given in the form of sparse matrices, with a number of rows equal to the number of clinical cases that compose the test set, and the number of columns corresponding to the possible labels. The prediction for each clinical case was retrieved, comprising a top K of most relevant labels and their confidence values ranging from -0.99 to 1. We used K=20 labels per clinical case. Then, using a Python script, we converted the predicted labels from their numeric identifiers to their corresponding eCIE-O codes using the label mapping file previously created. The script also discarded each label with a confidence score under a threshold chosen to achieve the highest Precision, Recall or F1 scores. For each of these measures, a ’.tsv’ file was created with the predictions for each clinical case in the format required by the competition. A fourth ’.tsv’ file was also created for the score threshold equal to 0, which was used as a baseline score. In the end, the files were used as input for the evaluation script given by the CANTEMIST competition so that we could retrieve the Mean Average Precision (MAP) scores for the predictions of our models.

In the test and background sets given by competition organizers, there were no eCIE-O codes indexing the clinical cases, so we had to put a placeholder label on each document, because X-Transformer was not prepared to run on unlabelled data. We also had to artificially adapt the size of the given test and background sets by splitting the sets into a total of 48 smaller sets of 250 clinical cases each. This procedure was necessary so that the files could have the same size as the test sets used on the trained X-Transformer models, which had a total of 250 articles. The first 109 lines of each of those files was composed by 109 clinical cases from the test and background sets to classify, and the remaining 141 came from the dev1 set, which was already classified with eCIE-O codes. These 141 clinical cases were used as an additional validation set to define the confidence threshold values of our submissions.

2.3.3. Developed Models

In a first iteration, we trained 4 models using the 501 indexed clinical cases that composed the train set, and the dev2 set that comprised a total of 250 indexed clinical cases, was used as test set. One of our models was trained using BERT Base Multilingual Cased and another was trained using BETO, the Spanish version of BERT.

The other two models were trained with two X-Transformer models that were previously developed by us for the Spanish biomedical domain using biomedical articles in Spanish retrieved from the IBECS, LILACS and PubMed databases, along with a list of keywords identified for each article using MER[31], a NER software. The major difference between the two models was that one of them was developed using 318,658 articles, while the other one used 50% more data, with a total of 637,316 articles. We shall call this two models Spanish Biomedical X-Transformer and Spanish Biomedical X-Transformer large, correspondingly. Summarizing, the 4 models had the following characteristics:

- Model 1: BERT base Multilingual Cased finetuned with the clinical records.
- Model 2: BETO finetuned with the clinical records.
- Model 3: Spanish Biomedical X-Transformer finetuned with the clinical records.
- Model 4: Spanish Biomedical X-Transformer large finetuned with the clinical records.

In a second iteration, we trained 4 additional models following the same characteristics of the previous ones, but using a larger train set composed by the 501 clinical cases that composed the CANTEMIST-CODING train set, plus 249 additional clinical cases from the dev1 set that were already classified. This way, we expected to achieved better results, since the models were trained with additional clinical cases. Summarizing, these four models had the following characteristics:

- Model 5: BERT base Multilingual Cased finetuned with 750 clinical records.
- Model 6: BETO finetuned with 750 clinical records.
- Model 7: Spanish Biomedical X-Transformer finetuned with 750 clinical records.
- Model 8: Spanish Biomedical X-Transformer large finetuned with 750 clinical records.

All models were trained using the default parameters of X-Transformer, except for the eval and train batch sizes which were both changed from their original values of 64 and 32 to 4 due to hardware constrains. We have also set the number of gradient accumulation steps to 2 to compensate for the small batch size. Each model was trained for 12 epochs, on a single NVIDIA Tesla P4 GPU.

2.3.4. Preliminary Results

In order to choose which model predictions to submit, we decided to evaluate the predictions made by each model on the dev2 set using the evaluation script given by the competition organizers. As was explained before, each model had 4 '.tsv' files as output, with each file containing the predictions with a confidence score superior to the confidence score threshold defined to best precision, recall or F1-score, and the baseline score, which corresponds to the confidence score threshold set to 0, which was the middle of the X-Transformer confidence score scale. Then, each file was used as input for the evaluation script given by the competition organizers.

The results for each model can be seen in Table 1. As it can be seen, the highest MAP scores are achieved when the predictions are focused on achieving the highest recall scores especially when using the models trained with the Spanish Biomedical X-Transformer models. We can notice that the models that use BETO seem to achieve higher scores when compared with the ones that used BERT Multilingual. In addition, we notice that there is not a clear difference between the usage of more clinical cases to train the models, with some models achieving slightly higher scores in MAP if the evaluation was focused on precision or in the F1-score, while in other models the score was inferior when compared with the models that used lesser articles.

Taking this into consideration, we decided to choose the predictions focused on recall of 5 distinct models to submit for the CANTEMIST-CODING task so we could also compare their performance in the competition. The chosen models were Models 2, 3, 4, 5 and 7. We then retrieved the predictions made by the models for each of the 48 text files that contained the test and background sets data. Then, for each of the resulting prediction files, the first 109 lines corresponding to the predictions made for the test and background sets were stored in the '.tsv' files, while the other 141 lines which corresponded to the predictions of the labelled cases from the first development set, were used to find the confidence score threshold that achieved the best recall score, and that would be used to choose which predictions would be stored in the final '.tsv' file.

3. Results and discussion

The results for the CANTEMIS-NER subtask are available in Table 2.

Our model obtained a F1-score of 0.749 in this subtask. Some errors prevented a higher performance, such as those related with the span of some detected entities. For example, in the document "cc_onco94.ann", the mention "linfoma" is correctly recognised by the NER tagger, but the processing script attributed the span "1690 1697", whereas the correct span would be "1691 1698". Besides, in some cases the NER tagger only recognised incomplete entity mentions.

Model	Focus	MAP	Threshold
Model 1 **BERT Base Multilingual Cased**	Baseline	0.222	0
	F1	0.366	-0,66
	Precision	0.18	0,22
	Recall	0.384	-0,81
Model 2 **BETO**	Baseline	0.267	0
	F1	0.385	-0,48
	Precision	0.188	0,48
	Recall	0.438	-0,83
Model 3 **Spanish Biomedical X-Transformer**	Baseline	0.293	0
	F1	0.378	-0,34
	Precision	0.191	0,52
	Recall	**0.446**	-0,82
Model 4 **Spanish Biomedical X-Transformer large**	Baseline	0.281	0
	F1	0.396	-0,45
	Precision	0.18	0,6
	Recall	**0.448**	-0,82
Model 5 **BERT Base Multilingual Cased** **(750 CC)**	Baseline	0.203	0
	F1	0.372	-0,46
	Precision	0.186	0,12
	Recall	0.407	-0,83
Model 6 **BETO** **(750 CC)**	Baseline	0.224	0
	F1	0.371	-0,46
	Precision	0.18	0,39
	Recall	0.417	-0,82
Model 7 **Spanish Biomedical X-Transformer** **(750 CC)**	Baseline	0.25	0
	F1	0.392	-0,4
	Precision	0.2	0,32
	Recall	**0.442**	-0,82
Model 8 **Spanish Biomedical X-Transformer large** **(750 CC)**	Baseline	0.268	0
	F1	0.379	-0,43
	Precision	0.194	0,29
	Recall	0.427	-0,81

Table 1

Preliminary results of the trained models for the CODING task using the second development set.
Bold values correspond to the three highest values achieved in the Mean Average Precision (MAP) measure using the evaluation script given by the competition organizers.
Green lines correspond to the models and corresponding evaluation focus whose predictions were chosen to submit for the CANTEMIST-CODING task.

For example, in the document "cc_onco89.ann", the NER tagger recognised the entity "implantes mediastínicos", whereas the full entity mention would be "implantes mediastínicos pleurales".

For future work, it would be interesting to train the other models beside the "medium" ("base", "large" and "pubmed") in the CANTEMIST corpus and to apply them to the test set to verify if they obtain a higher performance. Besides, we only trained the Flair embeddings during less than 100 epochs, so with more epochs, probably the performance of the NER tagger would be higher. The NER tagger itself could also be trained for more epochs, up until 150, according to a

Model	P	R	F1
medium	0.787	0.714	0.749

Table 2
Results for the CANTEMIST-NER subtask. P, R and F1 refer, respectively, to Precision, Recall and F1-score.

Model	P	R	F1	P-No-M	R-No-M	F1-No-M
1.single-ont	0.063	0.057	0.060	**0.059**	**0.082**	**0.069**
2.multi-ont	**0.064**	**0.058**	**0.061**	**0.059**	0.080	0.068

Table 3
Results for the CANTEMIST-NORM subtask. P, R and F1 refer, respectively, to Precision, Recall and F1-score, and P-No-M, R-No-M and F1-No-M refer, respectively, to the Precision, Recall and F1-score calculated without considering the mentions to metastasis (8000/6 code). Bold values correspond to the highest achieved scores in our submissions

suggestion by the authors of Flair. We will also address the errors associated with the span of the recognised entities.

The results for the CANTEMIS-NORM subtask are available in Table 3.

The model “multi-ont" obtained a F1-score of 0.061, which represents a slight improvement of +0.001 comparing to the “single-ont" model. Still, the performance of the model was too low, which is mainly related with the candidate retrieval step. Since we used string matching for candidate retrieval and selected the top candidates according to the edit distance between candidate/entity mention, most of the candidates lists did not contain the correct codes for the respective entity mentions. For example, the correct code for the entity mention “cáncer" would be the eCIE-O-3.1 code 8000/6, corresponding to the concept “Neoplasia metastásica". However, neither the “single-ont" model nor the “multi-ont" model were able to retrieve this candidate code for the entity mention, because the edit distance between “cancer" and “Neoplasia metastásica" is too high. Besides, the lack of a synonyms list in the eCIE-O-3.1 terminology further exacerbates the problem. Another aspect that is worth mentioning is the fact that the performance of the normalization model is always dependant on the performance of the NER tagger, so an incorrect output returned by the latter will hinder the results outputted by the former.

In order to improve the normalization model we will explore alternative methods for candidate generation, like for example, the use of word embeddings, both for entity mentions and for the terminology concepts. Instead of just considering that two entities in the same sentence are related, we will also use a proper relation extraction tool to get more semantically meaningful relations between concepts, either belonging to the same terminology, or belonging to different terminologies (for example, a relation between a eCIE-O-3 and a DeCS concept), which will densify the disambiguation graph and improve the disambiguation precision.

The results for the CANTEMIS-CODING subtask are available in Table 4.

Looking at our results, we can observe that, contrarily to what we expected, the best scoring model was Model 5 which, in our preliminary evaluation, had achieved the lowest MAP score of the five models submitted for this task. The lowest MAP scores were achieved by Models 3, 4 and 7, which were trained using the Spanish Biomedical X-Transformer models and that

Model	MAP	P	R	F1	MAP-No-M	P-No-M	R-No-M	F1-No-M
Model 2	0.463	0.157	0.549	0.244	0.350	0.119	0.466	0.189
Model 3	0.449	0.159	0.517	0.243	0.333	0.118	0.427	0.184
Model 4	0.455	0.151	0.532	0.235	0.344	0.113	0.445	0.180
Model 5	**0.506**	**0.211**	**0.601**	**0.312**	**0.399**	**0.167**	**0.527**	**0.254**
Model 7	0.459	0.197	0.541	0.289	0.346	0.151	0.456	0.226

Table 4
Results for the CANTEMIST-CODING subtask. MAP, P, R and F1 refer, respectively, to Mean Average Precision, Precision, Recall and F1-score, and MAP-No-M, P-No M, R-No-M and F1-No-M refer, respectively, to the Mean Average Precision, Precision, Recall and F1-score calculated without considering the mentions to metastasis (8000/6 code).
Bold values correspond to the highest scores for each metric in our submissions

had achieved the highest MAP scores in the preliminary evaluation. We can also notice that the precision and F1 scores were lower than the recall scores, but that was expected, since our submissions contained the predictions that used a confidence score threshold focused on achieving the highest recall scores.

As a proposal for a future work, we could try to develop additional models with a Transformer architecture, like the Biomedical X-Transformer models, and use them to train other X-Transformer models expecting to improve the number of correctly identified eCIE-O codes. These models could be trained using scientific biomedical articles in Spanish or even the clinical cases given by the CANTEMIST competition.

Another possible solution could pass by preprocessing the clinical case files before running them through X-Transformer by reducing the amount of text from each clinical case. This could be achieved by using automatic text summarization tools to leave only the essential information about each clinical case. Then, using NER tools, we could retrieve key entities and/or related terms and include them in the summarized text. This way, by reducing the original clinical case to a smaller and more objective text, along with identified key terms and entities given by the NER tools, it is expected that the X-Transformer model will be able to achieve better results since it has a smaller and more concise string of words, than a larger amount of text with the key topics more diluted.

4. Conclusion

We obtained a F1-score of 0.749 and 0.069 for the CANTEMIST-NER and the CANTEMIST-NORM subtasks, respectively, and a MAP of 0.506 for the CANTEMIST-CODING subtask. The code to run the developed models is available in our GitHub page: https://github.com/lasigeBioTM/CANTEMIST-Participation.

To improve the NER tagger, we intend to resume the training of the Flair embeddings up until 2000 epochs and to generate a larger language model, with 2048 hidden layers instead of 1024, and to train the tagger over more text, by including the development 2 set in the training process. For the normalization model, we intend to explore the use of word embeddings in the candidate generation process and the use of a relation extraction tool to build better disambiguation graphs. As to improvements in the classification model, we intend to explore additional X-Transformer models trained with new models using biomedical data in Spanish and through the combination with other NLP techniques, such as automatic text summarization and NER, in order to further improve the results achieved by the algorithm.

Acknowledgements

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

References

[1] O. Asan, P. Smith, E. Montague, More Screen Time, Less Face time – Implications for EHR Design, Journal of Evaluation in Clinical Practice 20 (2014) 896–901. doi:10.1111/jep.12182.

[2] P. Sfakianaki, L. Koumakis, S. Sfakianakis, G. Iatraki, G. Zacharioudakis, N. Graf, K. Marias, M. Tsiknakis, Semantic biomedical resource discovery : a Natural Language Processing framework, BMC Medical Informatics and Decision Making 15 (2015). URL: http://dx.doi.org/10.1186/s12911-015-0200-4. doi:10.1186/s12911-015-0200-4.

[3] P. Ernst, A. Siu, D. Milchevski, J. Hoffart, G. Weikum, DeepLife: An Entity-aware Search, Analytics and Exploration Platform for Health and Life Sciences, in: Proceedings ofthe 54th Annual Meeting ofthe Association for Computational Linguistics—System Demonstrations, Association for Computational Linguistics, Berlin, Germany, August 7-12, 2016, 2016, pp. 19–24. URL: http://www.aclweb.org/anthology/P16-4004{%}0Ahttp://aclweb.org/anthology/P16-4004. doi:10.1111/j.1348-0421.2010.00272.x.

[4] M. Vazquez, M. Krallinger, F. Leitner, A. Valencia, Text mining for drugs and chemical compounds: Methods, tools and applications, Molecular Informatics 30 (2011) 506–519. doi:10.1002/minf.201100005.

[5] J. He, M. de Rijke, M. Sevenster, R. van Ommering, Y. Qian, Generating Links to Background Knowledge: A Case Study Using Narrative Radiology Reports, in: CIKM'11, ACM, October 24–28, 2011, Glasgow, Scotland, UK., 2011, p. 1867. doi:10.1145/2063576.2063845.

[6] A. Miranda-Escalada, E. Farré, M. Krallinger, Named entity recognition, concept normalization and clinical coding: Overview of the cantemist track for cancer text mining in spanish, corpus, guidelines, methods and results, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020), CEUR Workshop Proceedings, 2020.

[7] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, R. Vollgraf, FLAIR: An easy-to-use framework for state-of-the-art NLP, in: NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Demonstrations Session, 2019, pp. 54–59.

[8] A. Lamurias, P. Ruas, F. M. Couto, PPR-SSM: Personalized PageRank and semantic similarity measures for entity linking, BMC Bioinformatics 20 (2019) 1–12. doi:10.1186/s12859-019-3157-y.

[9] Z. Huang, W. Xu, K. Yu, Bidirectional LSTM-CRF Models for Sequence Tagging (2015). URL: http://arxiv.org/abs/1508.01991. arXiv:1508.01991.

[10] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, CoRR abs/1810.04805 (2018). URL: http://arxiv.org/abs/1810.04805. arXiv:1810.04805.

[11] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, 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), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 2227–2237. URL: https://www.aclweb.org/anthology/N18-1202. doi:10.18653/v1/N18-1202.

[12] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics (2019) 1–7. doi:10.1093/bioinformatics/btz682. arXiv:1901.08746.

[13] E. Alsentzer, J. Murphy, W. Boag, W.-H. Weng, D. Jindi, T. Naumann, M. McDermott, Publicly Available Clinical, in: Proceedings ofthe 2nd Clinical Natural Language Processing Workshop, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 72–78. doi:10.18653/v1/w19-1909.

[14] M. Pershina, Y. He, R. Grishman, Personalized Page Rank for Named Entity Disambiguation, in: Human Language Technologies: The 2015 Annual Conference ofthe North American Chapter ofthe ACL, Section 4, Association for Computational Linguistics, Denver, Colorado, May 31 – June 5, 2015, 2015, pp. 238–243.

[15] Z. Guo, D. Barbosa, Robust named entity disambiguation with random walks, Semantic Web 9 (2018) 459–479. doi:10.3233/SW-170273.

[16] Y. Cao, L. Hou, J. Li, Z. Liu, Neural Collective Entity Linking (2018). URL: http://arxiv.org/abs/1811.08603. arXiv:1811.08603.

[17] O.-E. Ganea, T. Hofmann, Deep Joint Entity Disambiguation with Local Neural Attention, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Copenhagen, Denmark, September 7–11, 2017, 2017, pp. 2619–2629. URL: http://arxiv.org/abs/1704.04920. doi:10.18653/v1/d17-1277. arXiv:1704.04920.

[18] Z. Ji, Q. Wei, H. Xu, BERT-based Ranking for Biomedical Entity Normalization (2019). URL: http://arxiv.org/abs/1908.03548. arXiv:1908.03548.

[19] X. Yin, Y. Huang, B. Zhou, A. Li, L. Lan, Y. Jia, Deep Entity Linking via Eliminating Semantic Ambiguity With BERT, IEEE Access 7 (2019) 169434–169445. doi:10.1109/ACCESS.2019.2955498.

[20] K. Bhatia, H. Jain, P. Kar, M. Varma, P. Jain, Sparse local embeddings for extreme multi-label classification, in: Advances in Neural Information Processing Systems, 2015.

[21] J. Liu, W. C. Chang, Y. Wu, Y. Yang, Deep learning for extreme multi-label text classification, in: SIGIR 2017 - Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017. doi:10.1145/3077136.3080834.

[22] Y. Kim, Convolutional neural networks for sentence classification, in: EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 2014. doi:10.3115/v1/d14-1181. arXiv:1408.5882.

[23] R. You, Z. Zhang, S. Dai, S. Zhu, Haxmlnet: Hierarchical attention network for extreme multi-label text classification, CoRR abs/1904.12578 (2019). URL: http://arxiv.org/abs/1904.

12578. arXiv:1904.12578.

[24] W.-C. Chang, H.-F. Yu, K. Zhong, Y. Yang, I. Dhillon, Taming Pretrained Transformers for Extreme Multi-label Text Classification, 2020. URL: http://arxiv.org/abs/1905.02331v4. arXiv:1905.02331.

[25] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019). URL: http://arxiv.org/abs/1907.11692. arXiv:1907.11692.

[26] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, CoRR abs/1906.08237 (2019). URL: http://arxiv.org/abs/1906.08237. arXiv:1906.08237.

[27] Y. Prabhu, A. Kag, S. Harsola, R. Agrawal, M. Varma, Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising, in: 2018 World Wide Web Conference, International World Wide Web Conferences Steering Committee, International World Wide Web Conferences Steering Committee, 2018, pp. 993–1002.

[28] A. Akbik, D. Blythe, R. Vollgraf, Contextual String Embeddings for Sequence Labeling, in: Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 1638–1649. URL: https://github.com/zalandoresearch/flair.

[29] F. M. Couto, A. Lamurias, Semantic Similarity Definition, Reference Module in Life Sciences (2018) 0–16. URL: http://linkinghub.elsevier.com/retrieve/pii/B9780128096338204019. doi:10.1016/B978-0-12-809633-8.20401-9.

[30] L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web, Technical Report, Stanford InfoLab, 1998. URL: http://ilpubs.stanford.edu:8090/422/.

[31] F. M. Couto, A. Lamurias, MER: a shell script and annotation server for minimal named entity recognition and linking, Journal of Cheminformatics 10 (2018) 58. URL: https://jcheminf.biomedcentral.com/articles/10.1186/s13321-018-0312-9. doi:10.1186/s13321-018-0312-9.

[32] J. Cañete, G. Chaperon, R. Fuentes, J. Pérez, Spanish pre-trained bert model and evaluation data, in: to appear in PML4DC at ICLR 2020, 2020.