Vicomtech at LivingNER2022

Elena Zotova[1,2,†], Aitor García-Pablos[1,†], Naiara Perez[1], Pablo Turón[1] and Montse Cuadros[1]

[1]SNLT group at Vicomtech Foundation,
Basque Research and Technology Alliance (BRTA),
Mikeletegi Pasealekua 57, Donostia/San-Sebastián, 20009, Spain
[2]Department of Languages and Computer Systems. University of the Basque Country (UPV-EHU)
Paseo Manuel de Lardizabal, 1, Donostia/San-Sebastián, 20018, Spain

Abstract

This paper describes the participation of the Vicomtech NLP team in the LivingNER 2022 shared task about detecting and normalising mentions of living beings in clinical texts written in Spanish. We participate in each of the 3 LivingNER tasks, combining multiple approaches and strategies. For task 1 (NER) we use a Transformer-based model to perform sequence labelling. For task 2 (Normalisation) we use Semantic Text Search approaches to relate entity mentions to their taxonomy codes. For task 3 we try two different strategies: a trained multi-label classifier and a zero-shot semantic similarity approach. The results for task 1 and task 2 are high, both in our experiments and in the official evaluation results. For task 1 our system obtains an overall of 95.1% of F1-score. For task 2 our system achieves 93.04% F1-score. Task 3 was the most challenging and the one with the least available training data; the scores obtained by all the participating systems have been extremely low. According to the official results, our systems score more than 10 points above the average of the other participating systems for task 1 and 2.

Keywords

Named Entity Recognition, Clinical Text Coding, NCBI Taxonomy, Spanish Clinical Text

1. Introduction

This article describes Vicomtech's participation in the LivingNER 2022 shared task [1]. The challenge proposes recognising and normalising mentions of living beings in Spanish clinical case reports. It is split into three incremental tasks:

- Task 1: the LivingNER-Species NER track (Species mention entity recognition) requires that, given a collection of plain text clinical case report documents, participants return the exact character offsets of all living being mentions, both human and non-human.
- Task 2: the LivingNER-Species Norm track (Species mention normalisation) requires that, given a collection of plain text clinical case report documents, participating systems

IberLEF 2022, September 2022, A Coruña, Spain.

[†]These authors contributed equally.

ezotova@vicomtech.org (E. Zotova); agarciap@vicomtech.org (A. García-Pablos); nperez@vicomtech.org (N. Perez); pturon@vicomtech.org (P. Turón); mcuadros@vicomtech.org (M. Cuadros)
0000-0002-8350-1331 (E. Zotova); 0000-0001-9882-7521 (A. García-Pablos); 0000-0001-8648-0428 (N. Perez); 0000-0002-5563-1120 (P. Turón); 0000-0002-3620-1053 (M. Cuadros)

© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3202/livingner-paper1.pdf

Table 1
Distribution of documents and annotations for LivingNER task 1 and task 2 dataset splits.

	Documents	Annotated Spans			Unique NCBI Codes
		Total	SPECIES	HUMAN	
Train	1,000	16,097	9,090	7,007	813
Validation	500	7,106	3,817	3,289	523

Table 2
Distribution of documents and annotations for LivingNER task 3 dataset splits.

	Documents	Annotated Spans				
		Total	Pet	Animal Injury	Food	Nosocomial
Train	500	7,960	46	256	104	67
Validation	250	3,442	15	13	108	21

return all living being mentions together with their corresponding NCBI Taxonomy [2] concept identifiers.

- Task 3: the LivingNER-Clinical IMPACT track requires that, given a collection of plain text documents, systems perform binary classifications on four different axes and retrieve the list of NCBI Taxonomy IDs that support each binary classification.

We refer the reader to the challenge overview article [1] and the official website (https://temu.bsc.es/livingner/) of the competition for detailed information about LivingNER 2022.

Vicomtech's NLP team has implemented a number of tools to address the different stages of the task incrementally: entity mention detection has been addressed with a Transformer-based sequence labelling system; the problem of entity normalisation has been tackled with Semantic Text Similarity (STS) techniques; finally, supervised and unsupervised techniques have been tested for the multi-axial binary classification problem.

The paper is organised as follows: section 2 briefly describes the official data provided by the LivingNER organisers. It also describes how we have dealt with it to train our systems. Section 3 presents the system used for task 1 (NER). Section 4 describes the approach we have used for task 2 (Norm), while section 5 does the same for task 3 (Clinical IMPACT). Section 6 shows the official results obtained by our systems in the LivingNER competition. Finally, section 7 summarises the conclusions.

2. Dataset description

The LivingNER official dataset is composed of a training set and a validation set for each task. Task 1 annotations consist of spans of entity mentions with their corresponding label: HUMAN or SPECIES (for non-human living beings). For task 1 and task 2, the dataset is composed of the exact same set of labelled documents, in which the task 2 data adds NCBI code information on top of the task 1 labelled entity mentions. Table 1 describes the official datasets provided by

Table 3
Rebalanced training dataset for tasks 1 and 2

	Documents	Annotated Spans			Unique NCBI Codes
		Total	SPECIES	HUMAN	
Train	1,340	21,142	11,807	9,335	993
Validation	160	2,061	1,100	961	258

the competition organisers for these tasks. It is worth mentioning that, out of the 813 unique training codes and the 523 unique validation codes, 390 occur both in the training and validation set. For task 3, the labelled data is composed by a smaller subset of documents (see Table 2).

2.1. Train and Validation split rebalance

As Table 1, the ratio between training and validation data in the official splits was 2:1. In our opinion, the training of our systems could benefit from additional training data, while a smaller validation dataset could still be enough to extract solid conclusions about the validity of the systems prior to the official submission. Thus, we proceeded to rebalance the training and validation data. We reserved 160 documents from the original validation set (about 10% of training set), and combined the remaining validation documents with the original training data. The document and annotation distribution in the resulting splits is shown in Table 3. These new splits were used for our subsequent experiments in task 1 and task 2. All references to train and/or development data should be hereafter interpreted in this manner too.

2.2. NCBI Taxonomy enrichment

LivingNER organisers provided a TSV file containing the NCBI Taxonomy along with the training and validation data. This file contains entries with NCBI codes or IDs and the corresponding descriptions. These are the IDs that need to be associated to each entity mention found in the input texts, as part of task 2.

As another data pre-processing step, we have extended the provided taxonomy entries by incorporating the annotated terms from the training set. Furthermore, we have treated complex codes as atomic single codes. Complex codes occur when an entity mention is assigned more than one NCBI code. Consider the following sentence:

> Convive con **ganado vacuno, porcino** y aves

In this example, the text span “ganado vacuno, porcino” (beef cattle and pigs) is labelled with the complex ID “9913|9825”. Thus, we would add “9913|9825” to the NCBI Taxonomy as a unique ID with its own description. This way, we generated 3,568 new entries with Spanish descriptions.

3. Task 1 system (NER)

LivingNER task 1 requires detecting and classifying mentions of certain entities in the provided clinical documents. In other words, it is a Named Entity Recognition and Classification (NERC) task. There are only two different entity types: HUMAN and SPECIES. The former refers to mentions of doctors, patients, or other people. The latter groups all kinds of non-human living creatures, from micro-biological living creatures such as viruses and bacteria, to wild animals, farm animals or pets.

We have faced the task as a regular sequence labelling task, using IOB-tagging to emit one of B-TAG, I-TAG or O, where TAG is one of the relevant entity types (i.e., HUMAN or SPECIES). The sequence labelling is performed by a Transformer-based model, in particular with BETO [3], which encodes each input token into its contextual word embedding. These word embeddings pass through a classification head that projects the word embedding into the output label space.

Since the LivingNER documents are, in general, too long to fit in one piece into a Transformer model, we have applied the sliding windows technique, as described elsewhere [4]. In few words, we surround each window with a number of context tokens. These tokens are ignored when rebuilding the original document; they simply provide valuable information to resolve the central window by avoiding hard, meaningless segmentation cuts.

The training was configured to run during 100 epochs, with an early-stopping patience of 20 epochs. For this particular experiment the training stopped at epoch 39, obtaining a validation score of 95.14% of micro-averaged F1-score.

4. Task 2 system (Norm)

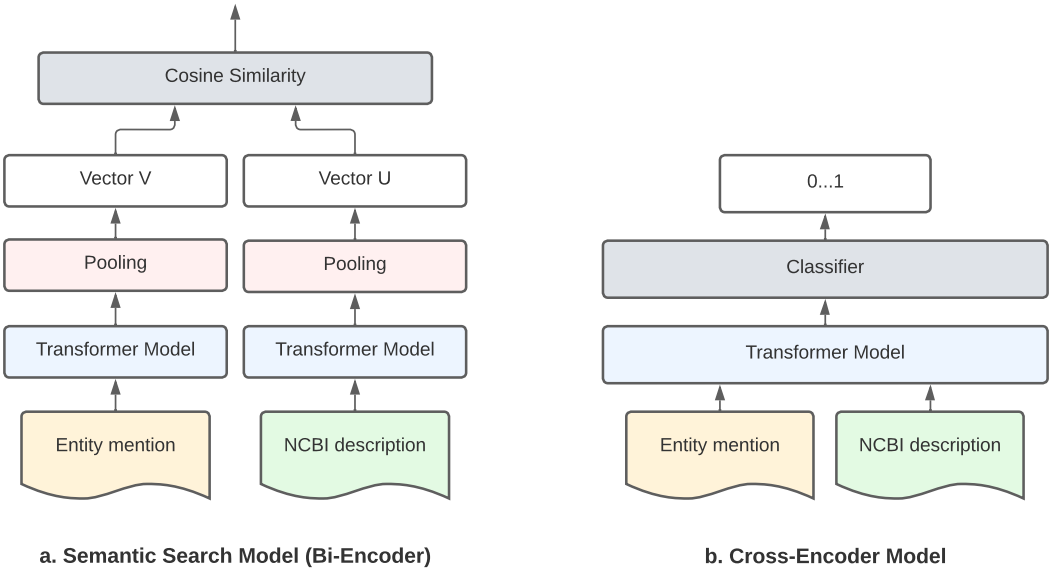
We have approached the LivingNER task 2 (Normalisation) using Semantic Text Similarity (STS) techniques. STS determines how similar two pieces of text are, by measuring their degree of semantic closeness. Semantic search is based on STS, allowing to retrieve relevant text results beyond the mere lexical matching.

4.1. Semantic search

The key concepts of semantic search are the following: query, collection of documents, and degree of relevance between a query and retrieved documents. We have adapted this method to map the LivingNER entities found in task 1 to codes in the NCBI Taxonomy.

In this case, the NCBI descriptions play the role of a collection of documents. The terms detected by the NER system are used as queries to search the closest documents, i.e., the closest NCBI descriptions. This process only applies to entities labelled as SPECIES. We exclude the terms detected as HUMAN, since according to the provided training and validation data, HUMAN entities always receive the same code: “9606”.

Semantic search consists in embedding all entries (sentences, documents, or, as in this case, taxonomy descriptions) into a vector space. At search time, the query, represented in this task by the detected entity mention, is also embedded into the same vector space. This allows a direct comparison of vectors using some metric or distance. Nowadays, the most extended method to



a. Semantic Search Model (Bi-Encoder) **b. Cross-Encoder Model**

Figure 1: Semantic models. Bi-encoder encodes two pieces of text separately and measure its relatedness with cosine similarity function. Cross-encoder model is trained with pairs of text and produces a score.

encode text is to use a pretrained Transformer model [5] to obtain the corresponding embeddings and compute the similarity score using a similarity metric (e.g., the cosine similarity).

4.2. Cross-encoder

In this work, we have extended the basic semantic search method by training a cross-encoder model [6]. Cross-encoders deal with sentence pair scoring and sentence pair classification tasks [7] They have also been proven successful in the clinical domain [8].

In contrast to an unsupervised semantic similarity function, the cross-encoder is trained by encoding both sentences simultaneously and produces a value between 0 and 1 that indicates the similarity or relatedness of the input sentence pair (see Figure 1b). That is, cross-encoders are trained using a set of text pairs labelled as similar/related (i.e., positive) or dissimilar/unrelated (negative).

Here, we have used the training dataset to retrieve positive examples from the NCBI Taxonomy file (extended as indicated in section 2.2). For each entity labelled in the corpus, we have created pairs with the entity's text and the NCBI description corresponding to the entity's NCBI code.

To obtain negative examples, we have used negative sampling methods. We have experimented with two different models:

- Semantic search with CANINE [9]: CANINE is a Transformer-based encoder that operates directly on character sequences, i.e., it does not require a tokeniser and thus has no arbitrary token space to model the texts.

Table 4
Examples for training the cross-encoder created with negative sampling of the query "antivirales" (antivirals). The label 1 indicated that the retrieved NCBI Taxonomy entry matches the code assigned in the training data to "antivirales".

CANINE		LaBSE	
Label	NCBI Description	Label	NCBI Description
1	antiviral	1	antivirales
1	antivirales	1	antiviral
0	antiretrovirales	1	antivíricos
1	Antiviral	1	antivírico
0	Antiaris	1	Antiviral
0	Acinetobacter antiviralis	0	antiparasitarios
0	antiretroviral	0	antiretrovirales
0	Antalis	1	antivírica
0	Antilles racer	0	antiretroviral
0	Antalis antillarum	0	antifúngicos
0	Antillicharis	0	antiparasitario
0	Anticharis	0	antirrábicos
0	Anthias ventralis	0	antiparasitaria
0	Antillicharis antillarum	0	antirretrovirales
0	Taraperla ancilis	0	antirretroviral
0	Agave antillarum	0	antituberculosos
0	Antispilina varii	0	antibacterianos
0	Anteris	0	antifúngico
0	Anticarsia	0	Antiretroviral
0	Antimerina	0	antifúngica

- Semantic search with a pretrained LaBSE model [10]: LaBSE is our preferable model for various reasons; first, it is based on multilingual BERT and is capable of producing language-agnostic sentence embeddings for 109 languages; second, the model combines masked language model and translation language model pretraining with a translation ranking task using bi-directional dual encoders. The multilingual sentence embeddings are crucial in the normalisation task because the corpus is in Spanish and the taxonomy is both in English and Spanish.

We have experimented with retrieving the top 64 and 100 most likely candidates for each labelled entity in the task 2 training set. These retrieved candidates, ranked by the notion of semantic similarity provided by the model embeddings, act as adversarial negative examples, i.e., they are somewhat close in terms of what the model computes, but they are not the correct choice (except, of course, those few candidates that happen to be the correct answer according to the labelled data).

Table 4 shows an example of candidates retrieved by the different models. While CANINE seems more guided by a character-based lexical similarity, LaBSE does a better job at detecting semantic closeness.

As a result, we obtained several datasets where the number of positive examples ranges from

Table 5
Size of training and development corpus for training the cross-encoder.

	Method	Train			Validation		
		Total	Negative	Positive	Total	Negative	Positive
NS1	LaBSE, top 100	295,607	276,502	19,105	49,626	46,312	3,314
NS2	CANINE, top 64	191,479	177,827	13,652	32,925	30,011	2,914
NS3	CANINE, top 100	299,146	284,699	14,447	50,920	47,921	2,999

4 to 7%, after discarding duplicate pairs when joining the positive examples obtained from the training data and the result of the negative sampling. Table 5 describes these datasets.

We have trained three LaBSE cross-encoder models, each from one of the computed datasets. The best checkpoint within 20 epochs of training was chosen according to the macro F1-score calculated on the validation dataset.

4.3. Approaches to NCBI code prediction

Based on the described semantic text search techniques and models, we have experimented with three different approaches to predict the correct set of codes given an entity mention text.

Semantic Search (SS) We encode both the entity words and NCBI Taxonomy with a LaBSE model and retrieve the closest candidate from the NCBI Taxonomy using the cosine similarity function (see Figure 1a). The code of the most similar taxonomy entry is used as the predicted code for each given entity.

Semantic Search and Binary classification (SS-BC) Given a new entity, this approach first computes a list of the top most likely candidates according to semantic similarity. Then, it applies a trained cross-encoder model to classify the candidates as related or not. The output of the model is a score from 0 to 1, representing the level of relatedness of two text pieces provided as input. When this score is higher than a given threshold, the outcome is interpreted as the pairs being related, otherwise as unrelated. We have calculated the F1-score after applying a range of thresholds, and selected the best performing threshold: 0.40.

Semantic Search and Rerank (SS-R) With this approach the prediction of the NCBI codes consists on the following steps:

1. Retrieve 100 candidates from NCBI Taxonomy with the semantic search approach described above.
2. Rerank the retrieved documents with the cross-encoder model described in section 4.2. The cross-encoder model re-scores the 100 candidates retrieved in the first step.
3. Get the candidate with the highest score from the cross-encoder and pick its corresponding NCBI code for the entity.

The process is depicted in Figure 2.

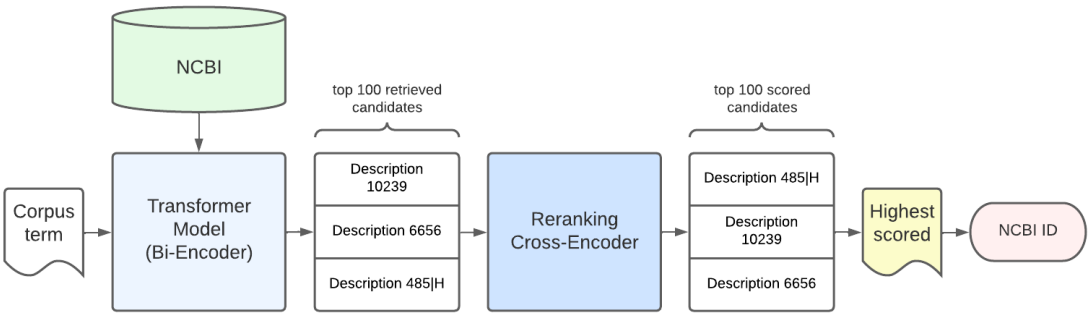


Figure 2: Workflow of the search and rerank process.

Table 6
Performance of the experiments with semantic search (SS), binary classification with cross-encoder model (SS-BC) and semantic search with rerank (SS-R) using different number of candidates.

System	Thresh	macro P	macro R	macro F1	Accuracy
SS@1	n/a	62.62	60.68	61.07	87.46
SS-BC: SS@64 + BC LaBSE on NS1	0.40	52.35	50.77	50.89	79.73
SS-BC: SS@100 + BC LaBSE on NS1	0.40	53.22	51.12	51.37	79.73
SS-BC: SS@64 + BC LaBSE on NS2	0.40	55.90	53.42	53.93	81.64
SS-BC: SS@100 + BC LaBSE on NS3	0.40	57.71	55.51	55.86	82.18
SS-R: SS@100 + R LaBSE on NS1	n/a	**70.08**	**69.28**	**69.05**	**93.00**
SS-R: SS@100 + R LaBSE on NS3	n/a	69.58	68.02	68.22	92.91

4.4. Experiments

We experimented with the three described approaches to predict the correct NCBI codes given an entity text. The performance is calculated over the task 2 validation set, only for entities labelled as SPECIES. All experiments use the Sentence-Transformers [6] and FAISS [11] libraries.

Table 6 shows the results of the experiments with the different approaches and variations: the semantic search (SS) without rerank, the semantic search with binary classification (SS-BC) and the semantic search with rerank (SS-R), using different number of candidates.

According to these results semantic search with rerank outperforms the other approaches, in particular the variant whose cross-encoder model was trained on the examples generated by negative sampling with the LaBSe model (NS1). For this reason, we have selected this system for our submission to the task competition.

5. Task 3 system (Clinical IMPACT)

The LivingNER task 3 builds up on top the previous tasks. For a given clinical document, the system must classify it into certain classes, and gather which NCBI codes support the

classification. The codes, in turn, come from the entities detected in the document. So, for task 3 to be feasible, the outcomes from the two previous tasks are required.

The classes into which a document must be classified in task 3 are:

- Pet: whether the document contains information corresponding to pets and farm animals that live together with the patient.
- Animal injury: whether the document mentions animals (e.g., spiders, wasps, bees) causing a direct injury on humans like bites, stings, etc. Parasites are not included.
- Food: whether the document contains mentions corresponding to food. This includes ingested or food, not other entities intaken by the patient that are not food.
- Nosocomial: detect species mentions corresponding to nosocomial, hospital-acquired, or healthcare-associated infections.

We assume that the entities are already detected and normalised (from the previous two tasks), and we focus on assigning zero or more of the enumerated labels to each of the entities for a given document.

We have tried two different strategies to deal with this challenging scenario. One of them is based on a multi-label classifier approach that takes each entity and the context around it, and tries to emit the relevant categories. The other strategy is based on a zero-shot classification based on semantic similarity between the entity text and the words of the class-labels in Spanish (Pet:"mascota", Food:"comida", etc.).

5.1. Multi-label classification

The multi-label classification approach is based on a Transformer model and a relevant piece of the input document to predict if zero or more of the target classes apply to a given entity mention.

The task organisers provide two formats of data to encode the information relevant to task 3. The final one, the one that has to be submitted, summarises the presence/absence of each class (pet,animal injury, food, nosocomial) and their supporting NCBI codes per document (i.e. each document is represented in a single line). The other format contains the equivalent information for each individual entity annotation, similar to the data from task 1 and task 2. For each document there might be several entities, and for entity mention there is one or more NCBI codes assigned, plus the information about if the entity is evidence of pet/animal-injury/food/nosocomial classes. We make use of this latter format to train a multi-label classifier.

The input to the model is the entity mention in the context in which it appears in a given document. The mention is centred so the available context to the right and to the left is of equivalent length. An entity-mention-mask is calculated as an additional input. This entity-mention-mask contains a 1 in the token-positions that belongs to the entity mention, and a 0 for the context.

The whole context is encoded using a Transformers model, and the resulting contextual-word-embeddings are filtered out according to the entity-mention-mask, and averaged to obtain a single final embedding of fixed size that encodes the instance. This embedding is used as the input for a classification layer that obtain the multi-label output for each of the four possible classes. Figure 3 shows a diagram of the described model.

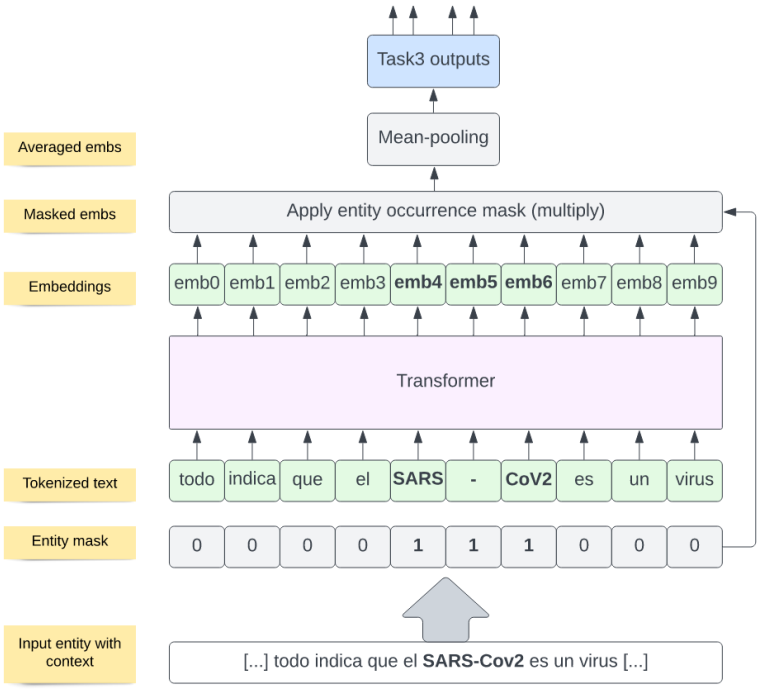


Figure 3: Diagram of the multi-label classifier model used for task 3.

We have trained and validated this model using the provided training and validation data. However, the validation results were not satisfying. Because of that, we also tried with a zero-shot approach based on semantic similarity as an alternative strategy.

5.2. Zero-shot classification

Zero-shot refers to a scenario in which a model trained for a certain task is evaluated on a different task for which is has received no specific training. In the context of LivingNER task 3, we use a pretrained model to embed both the entities and the human readable class names (in Spanish) into the same vector space, so they can be compared.

A similar method has been successfully used in various tasks, such as domain labelling [12], or intent classification [13]. The approach, proposed by [14], uses a pretrained MNLI (Multi-Genre Natural Language Inference) sequence-pair classifier as an out-of-the-box zero-shot text classifier. The idea is to take the text (entity mention) we are interested in labelling as the "premise" and to turn each candidate label into a "hypothesis". If the NLI model predicts that the premise "entails" the hypothesis, we interpret that the corresponding label applies to the provided text.

As the zero-shot labels are categories that should be semantically significant, we turn the target categories into Spanish words or expressions:

- Pet: mascota

Table 7
Performance of the best zero-shot model (ZeroShot) with thresholds on the validation set, and the validation scores obtained by the fine-tuned Multi-label classification model (ML-Class).

System	Threshold	macro F1	macro R	macro P	Accuracy
ML-Class	n/a	30.05	72.62	20.47	-
ZeroShot	0.50	24.15	**42.44**	33.27	51.59
	0.60	**26.56**	35.55	58.31	69.00
	0.70	20.47	25.56	57.15	72.61
	0.80	17.21	18.93	55.78	75.37
	0.90	17.41	19.36	55.82	77.07
	0.95	17.68	19.89	75.91	79.19
	0.97	17.70	19.95	**75.92**	**79.41**

- Animal Injury: lesión de un animal
- Food: comida
- Nosocomial: infección nosocomial

The output of this zero-shot approach is a confidence score per target label. To model the fact that most of the entities do not belong to any label, we set a minimum confidence threshold. If no label obtains a confidence higher than the given threshold, then none of the labels is assigned to that entity. Otherwise, the label with the higher confidence is chosen.

This method has some inherent limitations. On the one hand, it is not well suited to choose labels in a multi-label setting (e.g., if a term is annotated as Pet and Animal Injury at the same time, the model will choose only one label). On the other hand, we are comparing only the entity mention itself, so any other evidence present in the context that might be helpful to make a correct prediction cannot be taken into account.

5.3. Experiments

We have evaluated the trained multi-label classifier on the task 3 validation dataset, and we have also experimented with different models for the zero-shot approach. The zero-shot model that obtained the best F1-score was XLM-RoBERTa-large model fine-tuned over several NLI datasets (https://huggingface.co/vicgalle/xlm-roberta-large-xnli-anli).

Table 7 shows the results of the experiments with different thresholds for the zero-shot approach, together with the scores obtained by the multi-label classification approach. For the zero-shot approach, the best performing threshold in the validation set was 0.6.

6. Results

In this section we describe the official results obtained by our submitted systems in the LivingNER competition. The results were computed by the LivingNER organisers using the labelled test and background set submitted by each competing team. The test and background set is a collection of 13,467 clinical case reports. Among the 13,467 clinical case reports, 485 are the actual test set

Table 8
Official results of our participation in LivingNER task 1 (NER) and task 2 (Norm)

		HUMAN			SPECIES			Total		
		P	R	F1	P	R	F1	P	R	F1
Task 1	BETO	0.978	0.979	0.978	0.944	0.918	0.931	0.958	0.944	0.951
	MEAN	0.931	0.875	0.885	0.811	0.758	0.778	0.876	0.808	0.824
	STDEV	0.116	0.239	0.223	0.249	0.256	0.251	0.154	0.247	0.237
Task 2	LaBSE SSR	0.978	0.979	0.978	0.907	0.882	0.895	0.938	0.923	0.930
	MEAN	0.959	0.962	0.960	0.760	0.692	0.723	0.849	0.807	0.827
	STDEV	0.019	0.014	0.015	0.289	0.258	0.270	0.158	0.148	0.151

Table 9
Official results of our participation in LivingNER task 3 (Clinical IMPACT) with the Multi-Label classifier (ML-Class) and the Zero-shot classifier (ZeroShot), taking into account the code predictions.

	Pets			Animal injuries			Food			Nosocomial		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ML-Class	0.000	0.000	0.000	0.000	0.000	0.000	0.003	0.154	0.005	0.000	0.000	0.000
ZeroShot	0.000	0.000	0.000	0.001	0.125	0.001	0.009	0.115	0.016	0.000	0.000	0.000
MEAN	0.009	0.102	0.017	0.000	0.021	0.000	0.009	0.154	0.016	0.000	0.000	0.000
STD	0.015	0.162	0.027	0.000	0.051	0.000	0.010	0.152	0.018	0.000	0.000	0.000

used for the evaluation. The rest is the background added to prevent manual corrections on the automatically labelled data.

At the time of writing this article, the results from all the other participants have not been disclosed by the organisers, and the only information we have to compare our systems is the average metrics aggregated from all the participants, together with the standard deviation.

Table 8 shows the results obtained by our only submission to the LivingNER task 1 (NER) and task 2 (Norm). The results for task 1 are high, with precision and recall several points above 90%, and more than 10 points above the mean score for all the participants.

The results for task 2 are also high. In this task, the challenge lied in assigning correct codes to the entities labelled as SPECIES, since entities labelled as HUMAN received automatically the same code. In the case of SPECIES, our system obtains a score of 89.5%, which seems a very competitive score compared to the average of the other participating systems, which is about 17 points lower.

Table 9 shows the results obtained by our two different submissions to LivingNER task 3 (Clinical IMPACT). Table 10 shows the results for the same task, but relaxing it to consider only the classification of four axes, that is, ignoring the supporting codes.

In both cases, the results are extremely low, with precision and recall scores of zero or near zero in most of the cases. According to the mean scores from all the participant systems, these low scores are the trend for this task, which indicates that either the task was particularly difficult, or that the amount of provided data was not enough to solve the proposed problem.

Table 10
Official results of our participation at LivingNER task 3 (Clinical IMPACT) with the Multi-Label classifier (ML-Class) and the Zero-shot classifier (ZeroShot), not taking into account the code predictions.

	Pets			Animal injuries			Food			Nosocomial		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ML-Class	0.024	0.833	0.047	0.006	0.500	0.012	0.015	0.846	0.030	0.000	0.000	0.000
ZeroShot	0.033	0.250	0.058	0.003	0.625	0.006	0.024	0.308	0.044	0.002	0.750	0.003
MEAN	0.020	0.292	0.037	0.009	0.333	0.018	0.018	0.391	0.034	0.001	0.208	0.003
STD	0.017	0.311	0.030	0.011	0.246	0.021	0.018	0.404	0.034	0.002	0.332	0.005

7. Conclusions

In this paper we have described our participation in the LivingNER 2022 competition about detecting and normalising entities in clinical texts. The competition is split into three different tasks with an incremental level of difficulty. For task 1 (NER), which requires finding mentions to entities in the provided clinical texts, with have trained a sequence labelling model based on Transformers. For task 2 (Norm), which requires assigning specific NCBI codes to each detected entity, we have implemented a system based on Semantic Text Similarity. For task 3 (Clinical IMPACT), which consists in classifying each document under certain categories and indicate which codes are evidence for said classification, we have tried a supervised classifier and a zero-shot semantic comparison. Our results for task 1 and task 2 are above the 90% of F1-score in most of the cases, and more than 10 points above the mean score obtained by other participating systems. However, task 3 results are extremely low, both for our systems and for the other participating systems, according to the mean scores obtained. It is unclear if task 3 was too challenging or the provided data was too scarce. As future work, it would be interesting to make more experiments with the task 3 scenario to try to improve the results.

References

[1] A. Miranda-Escalada, E. Farré-Maduell, S. Lima-López, D. Estrada, L. Gascó, M. Krallinger, Mention detection, normalization & classification of species, pathogens, humans and food in clinical documents: Overview of LivingNER shared task and resources, Procesamiento del Lenguaje Natural (2022).

[2] C. L. Schoch, S. Ciufo, M. Domrachev, C. L. Hotton, S. Kannan, R. Khovanskaya, D. Leipe, R. Mcveigh, K. O'Neill, B. Robbertse, et al., Ncbi taxonomy: a comprehensive update on curation, resources and tools, Database 2020 (2020).

[3] J. Cañete, G. Chaperon, R. Fuentes, J. Pérez, Spanish Pre-Trained BERT Model and Evaluation Data, in: Proceedings of the Practical ML for Developing Countries Workshop at the Eighth International Conference on Learning Representations (ICLR 2020), 2020, pp. 1–9.

[4] A. García-Pablos, N. Perez, M. Cuadros, Vicomtech at CANTEMIST 2020, in: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020) co-located with 36th Conference of the Spanish Society for Natural Language Processing (SEPLN 2020), 2020, pp. 489–498.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017.

[6] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992.

[7] R. Nogueira, K. Cho, Passage re-ranking with bert, ArXiv abs/1901.04085 (2019).

[8] A. Rahimi, T. Baldwin, K. Verspoor, WikiUMLS: Aligning UMLS to Wikipedia via cross-lingual neural ranking, in: Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 5957–5962.

[9] J. H. Clark, D. Garrette, I. Turc, J. Wieting, Canine: Pre-training an Efficient Tokenization-Free Encoder for Language Representation, Transactions of the Association for Computational Linguistics 10 (2022) 73–91.

[10] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, W. Wang, Language-agnostic BERT Sentence Embedding, 2020. arXiv:2007.01852.

[11] J. Johnson, M. Douze, H. Jégou, Billion-scale similarity search with GPUs, IEEE Transactions on Big Data 7 (2019) 535–547.

[12] O. Sainz, G. Rigau, Ask2Transformers: Zero-shot domain labelling with pretrained language models, in: Proceedings of the 11th Global Wordnet Conference, Global Wordnet Association, University of South Africa (UNISA), 2021, pp. 44–52.

[13] C. Xia, C. Zhang, X. Yan, Y. Chang, P. Yu, Zero-shot user intent detection via capsule neural networks, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 3090–3099.

[14] W. Yin, J. Hay, D. Roth, Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach, 2019. doi:10.48550/ARXIV.1909.00161.