uhKD4 at eHealth-KD Challenge 2021: Deep Learning Approaches for Knowledge Discovery from Spanish Biomedical Documents

Dayany Alfaro-González, Dalianys Pérez-Perera, Gilberto González-Rodríguez, Antonio Jesús Otaño-Barrera, and Rocío Cruz-Linares[0000−0002−0069−8950]

Faculty of Math and Computer Science, University of Havana, La Habana, Cuba

Abstract. This paper describes the system presented by team uhKD4 in the IberLEF eHealth Knowledge Discovery Challenge 2021. The challenge proposes two tasks devoted to extract the semantic meaning of sentences mainly health-related in the Spanish language: Task A (entity recognition) and Task B (relation extraction). The sequential attainment of both tasks represents the main evaluation scenario of the challenge. The system is built upon two independent deep-learning-based architectures, one for each task of the challenge. Task A is addressed as a sequence labelling problem with a model that uses Long Short-Term Memory layers to encode context information and linear chain Conditional Random Fields as tag decoders. Task B is approached as a multi-class classification problem using a Convolutional Neural Network that consists mainly of convolutional layers to recognize n-grams, the pooling layers to determine the most relevant features and a logistic regression layer at the end to perform classification. The system obtained the fourth position in the main evaluation scenario of the competition. In the individual evaluation of the tasks the model for Task A showed average results while the Task B model reached the third position.

Keywords: eHealth · Knowledge Discovery · Natural Language Processing · Information Extraction · Named Entity Recognition · Relation Extraction · Deep Learning.

1 Introduction

This paper presents a description of the solution submitted by team uhKD4 at the IberLEF eHealth Knowledge Discovery Challenge 2021. The challenge proposes two tasks devoted to extract the semantic meaning of sentences mainly health-related in the Spanish language: Task A (entity recognition) aims to identify all the entities in a document and their types and Task B (relation extraction)

IberLEF 2021, September 2021, Málaga, Spain.

seeks to recognize all relevant semantic relationships between the entities recognized. The sequential attainment of both tasks represents the main evaluation scenario of the challenge[7].

The system proposed consists in two independent components, one for each task. In order to solve the named entity recognition (NER) problem associated to Task A we present a model that uses Long Short-Term Memory (LSTM) layers to encode context information, motivated by the fact that it has demonstrated remarkable achievements in modeling sequential data [4]. On top of that are added a dense layer and a Conditional Random Field (CRF) [3] layer, which has been widely used as a tag decoder taking the context-dependent representations and producing a sequence of tags corresponding to the input sequence [4]. The relation extraction (RE) problem framed in Task B is approached using a Convolutional Neural Network (CNN) that consists mainly of convolutional layers to recognize n-grams, the pooling layers to determine the most relevant features and a fully connected neural network with a softmax at the end to perform classification [6].

The rest of this paper is organized as follows. Section 2 describes in detail the architectures used by the system. The official results achieved in each scenario of the challenge are shown in Section 3. In Section 4 are shared some insights derived from experimentation. Finally, in Section 5 are stated the conclusions and future work recommendations.

2 System Description

Our system is built upon two independent deep-learning-based architectures. Accordingly, two different models are defined and each task is carried out separately. Task A is approached as a sequence labelling problem in which each token from an input sequence is assigned a label that represents the combination of the BILUOV entity tagging scheme with each one of the possible types of an entity. The BILUOV tags correspond to: ***B****egin*, to represent the start of an entity; ***I****nner*, to represent its continuation; ***L****ast*, to represent its end; ***U****nit*, to represent single word entities; ***O****ther*, to represent words that are not a part of any entity; and *o****V****erlapping*, to represent words that belong to multiple entities [1]. For example, in the sentence *"El cáncer de la cavidad nasal y de los senos paranasales no es común"* each word should be labeled as stated between parenthesis: *El* (O) *cáncer* (V-Concept) *de* (I-Concept) *la* (I-Concept) *cavidad* (I-Concept) *nasal* (L-Concept) *y* (O) *de* (I-Concept) *los* (I-Concept) *senos* (I-Concept) *paranasales* (L-Concept) *no* (O) *es* (O) *común* (U-Concept). Thus, the output of the model considers 21 different labels: the O label and the combination of the remaining tags (BILUV) and the entity types (Concept, Action, Predicate and Reference). The proposed approach to Task B is to solve a multi-class classification problem, in which given a sentence and a highlighted pair of entities, one of the predefined relations is assigned to occur from the first entity toward the second one. A new artificial relation class *none* is defined to symbolize the non-occurrence of any relation between a pair of entities.

2.1 Preprocessing

The initial step to extract useful information from the input of raw text is the tokenization of each sentence, since both of the tasks require the analysis of the sequence of words in the sentence. A fixed length for the sentences is defined as a parameter for the models and each sequence of tokens is trimmed or padded accordingly to fit the designated length. Below are exposed the particular features that were considered to obtain the input representation for each model.

Common

- **Word embedding:** Pre-trained word embedding word2vec [5] that have dimensionality of 300 and was trained on the the Spanish Billion Words Corpus with the variant of skip-gram model with negative-sampling. The weights are kept unchanged during the training phase.
- **POS-tag embedding:** Embedding to encode the information expressed by the Part-of-speech tag of the token.

Task A

- **Character representation:** Every token is trimmed or padded in order to ensure that they all have the same predefined number of characters. By means of an embedding layer, each character of a word is translated to a vector, that represents one of all the ASCII letters, digits, and punctuation symbols and then are fed into a RNN-based model, that uses a Bidirectional Long Short-Term Memory (BiLSTM) to obtain a character-level representation of the token.

Task B

- **BILUOV and Entity Type embedding:** Embedding intended to encode the information that gives the corresponding label of each word according to the combination of the BILUOV tag system and the possible types of entity.
- **Position embeddings:** Embeddings to encode the relative distance between each word and the two target entities in the sentence. In the case of a multi-word entity is considered the distance to the first word of such entity.

2.2 Named Entity Recognition Model

Figure 1 shows the architecture of the defined model. As stated in the previous subsection the input of the model is a sequence of tokens, each one represented as the concatenation of the vectors from word and POS-tag embeddings and the character-level features. After the input is handled, the sequence of word vectors is processed in both directions by a BiLSTM layer and the features extracted from the forward and backward passes are concatenated together. The resulting sequence is intended to increase the amount of information available to the

network, improving the context available to the algorithm (e.g. knowing what words immediately follow and precede a word in a sentence). Afterward, the sequence is processed by a simple LSTM layer to extract the most important features. Finally, a dense layer with a linear activation function followed by a linear-chain CRF are used to output the most probable sequence of labels corresponding to the tokens. The CRF layer uses sentence-level tag information to add some constraints to the final predicted labels to ensure they are valid. These constraints can be learned automatically from the dataset during the training process.

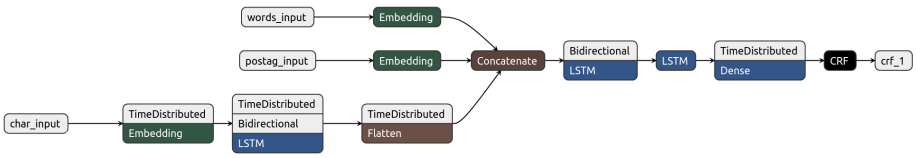

Fig. 1. Task A model architecture.

Since the goal is to classify in only four types of entities, a subsequent phase of decoding the output of the CRF layer is needed. The required transformation is realized in a way that is similar to the process described by team UH-MatCom at the previous edition of the challenge [1]. The process is accomplished in two steps. First, rules are used to discover the possible entities that use overlapped words and are not formed by continuous words. After, the remaining entities are assumed to be a continuous sequence of tokens and are detected in an iterative manner.

2.3 Relation Extraction Model

The architecture defined for Task B is shown in Figure 2. The relation extraction system is provided only with raw sentences marked with the positions of the two entities of interest and the corresponding type of each one. Thus, exploiting the elements that can be derived from that input, each relation mention is represented by a matrix $X = [w_1, w_2, ..., w_n]$, where n is the defined length for the sentences and w_i is the result of concatenating for the i-th token the embeddings described before.

The matrix X is processed by the convolutional layer in order to extract high-level features. A filter with window size s can be denoted as $F = [f_1, f_2, ..., f_s]$. Applying the convolution operation on the two matrices X and F is gotten a score sequence $T = [t_1, t_2, ..., t_{n-s+1}]$:

$$t_i = g(\sum_{j=0}^{s-1} f_{j+1}^T w_{j+i} + b) \tag{1}$$

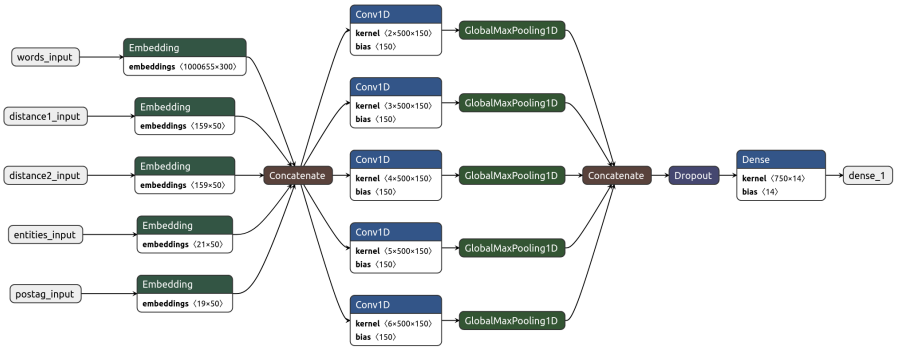

Fig. 2. Task B model architecture.

where g is some non-linear function and b is a bias term. This process is replicated for various filters with distinct window sizes to explore the contribution of different n-grams. Then, a pooling layer is applied to aggregate the scores for each filter to assure the invariance to the absolute positions but retain the relative positions among the n-grams and the entities. Specifically, a global max pooling layer is used to aggressively summarize the most important or relevant features from each score sequence. A dropout is applied to the resulting feature vector for regularization, and then is fed into a fully connected layer of standard neural networks that is followed by a softmax layer in the end in order to carry out classification [6].

2.4 Hyperparameters Setup

Tables 1 and 2 show the selected set of hyperparameters for the NER and RE models respectively. In both tables are exposed the configurations respecting the input handling at the top, whereas the middle section covers the rest of the network and at the bottom are located the hyperparameters for training.

The hyperparameter tuning process was carried out manually, taking as a starting point some settings that have shown a positive impact in past works involving similar architectures. The provided development collection was used as the validation dataset. The number of epochs was selected according to the performance shown in training curves.

2.5 Training

For the implementation of the systems was used Python programming language and the framework Keras(v2.2.4) with TensorFlow(v1.13.1) as backend. In the NER model was used the keras_contrib(v0.0.2) implementation for the CRF layer. Tokenization and POS-tags were obtained using the model `es_core_news_md` of the Python library spaCy (v3.0.6).

The training collection provided for the challenge was the only data used to train both models. The process was carried out in a machine with a 4 core AMD A10-8700P CPU at 1.80 GHz with an installed memory of 16 GB. For the NER model the training time was close to 8 hours and for the RE model it took little more than 2 hours.

Table 1. Hyperparameters setup for NER model

Hyperparameter	**Value**
POS-tag embedding	50
Character embedding	50
Character BiLSTM recurrent units	150
Max sentence length	40
Max word length	20
BiLSTM recurrent units	300
BiLSTM recurrent dropout	0.5
LSTM recurrent units	300
LSTM recurrent dropout	0.5
Optimizer	RMSprop
Learning rate	0.001
Loss function	Negative Log-Likelihood
Mini-Batch size	64
Epochs	9

Table 2. Hyperparameters setup for RE model

Hyperparameter	**Value**
POS-tag embedding	50
BILUOV and Entity Type embedding	50
Position Entity 1 embedding	50
Position Entity 2 embedding	50
Max sentence length	80
Convolution number of filters	150
Convolution window sizes	(2,3,4,5,6)
Non-linear function g	tanh
Dropout rate	0.5
Optimizer	Adam
Learning rate	0.001
Loss function	Categorical Crossentropy
Mini-Batch size	32
Epochs	15

3 Results

Table 3 presents the official results for the main scenario of evaluation in the challenge, where our team obtained the fourth position, achieving a F1 score of 0.423. There is a significant difference respecting the F1 scores of our system and the ones ranked higher. Thus, although we achieve competitive results, there is still room for improvement.

Table 3. Scenario 1 (Main Evaluation)

Team	F1	Precision	Recall
Vicomtech	0.531	0.541	0.535
PUCRJ-PUCPR-UFMG	0.528	0.568	0.503
IXA	0.499	0.465	0.539
uhKD4	**0.423**	**0.485**	**0.374**
UH-MMM	0.339	0.292	0.404
Codestrange	0.232	0.337	0.177
baseline	0.232	0.337	0.177
JAD	0.109	0.234	0.071

In the second task, regarding entity extraction, our system shows the least promising results of all scenarios, ranking fifth with F1 score of 0.527, as shown in Table 4. Whilst, on the contrary, a value of 0.318 for F1 score is achieved and the third position is reached for the relation extraction task, which results are presented in Table 5.

Table 4. Scenario 2 (Task A)

Team	F1	Precision	Recall
PUCRJ-PUCPR-UFMG	0.706	0.715	0.697
Vicomtech	0.684	0.700	0.747
IXA	0.653	0.614	0.698
UH-MMM	0.608	0.546	0.685
uhKD4	**0.527**	**0.518**	**0.537**
Yunnan-Deep	0.334	0.520	0.246
baseline	0.306	0.350	0.225
JAD	0.263	0.316	0.071
Yunnan-1	0.173	0.271	0.127
Codestrange	0.080	0.415	0.044

Table 5. Scenario 3 (Task B)

Team	F1	Precision	Recall
IXA	0.430	0.454	0.409
Vicomtech	0.372	0.542	0.283
uhKD4	**0.318**	**0.556**	**0.222**
PUCRJ-PUCPR-UFMG	0.263	0.367	0.205
UH-MMM	0.054	0.077	0.041
Codestrange	0.033	0.438	0.017
baseline	0.033	0.438	0.017
JAD	0.007	0.375	0.004

4 Discussion

We would like to remark the relevance of the used features for both models. In particular, the NER model using only the pretrained word embedding showed poor results while the addition of the POS-tag and character information provided a significant boost in performance.

Regarding RE task, a mayor issue to overcome is the data scarcity problem, the amount of non-relation entity pairs is often superior to the ones that represent a relation, which leads to a widely unbalanced dataset and have a negative impact on the performance of models. To mitigate this problem we enriched the input representation with BILUOV tags and entity type information, in order to capture patterns in which the entities appear in a sentence that may be helpful to discriminate between positive and negative instances. The technique of adding the tag system information has been explored before in an architecture that is similar to ours and good results were achieved [8]. Experimentation proved that the incorporation of those features was highly influential in performance, as we expected.

Also, related to the architecture of the RE model, it is worth mentioning that we experimented using max pooling layers or the global ones and better results were achieved in the second case.

5 Conclusions

In this paper was described the system proposed by team uhKD4 at the IberLEF eHealth Knowledge Discovery Challenge 2021. Two independent deep-learning-based models were defined to solve each task of the competition. Task A is solved as a sequence labelling problem, by a model that uses a word2vec pretrained embedding along with syntactic features as the input representation, which is afterwards processed by LSTM and CRF layers. Task B is approached as a multi-class classification. In this case, besides the pretrained word embedding and syntactic features, it is also used information from the BILUOV tags and the relative distance to the highlighted entities. Then a CNN with filters of multiple window sizes and a logistic regression layer at the end performs classification.

The system obtained the fourth position in the main evaluation scenario of the competition. In the individual tasks the NER model showed average results while the RE model reached the third position.

As future work recommendations we propose to consider the use of domain specific features and external sources of knowledge. Also, to explore the use of contextual embeddings, such as Bidirectional Encoder Representations from Transformers (BERT) [2].

References

1. Consuegra-Ayala, J.P., Palomar, M.: UH-MatCom at eHealth-KD Challenge 2020: Deep-Learning and Ensemble Models for Knowledge Discovery in Spanish Documents (2020)
2. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2019)
3. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning. p. 282–289. ICML '01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)
4. Li, J., Sun, A., Han, J., Li, C.: A survey on deep learning for named entity recognition (2020)
5. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality (2013)
6. Nguyen, T., Grishman, R.: Relation extraction: Perspective from convolutional neural networks. pp. 39–48 (01 2015). https://doi.org/10.3115/v1/W15-1506
7. Piad-Morffis, A., Gutiérrez, Y., Estevez-Velarde, S., Almeida-Cruz, Y., Muñoz, R., Montoyo, A.: Overview of the eHealth Knowledge Discovery Challenge at IberLEF 2021. Procesamiento del Lenguaje Natural **67**(0) (2021)
8. Ye, W., Li, B., Xie, R., Sheng, Z., Chen, L., Zhang, S.: Exploiting entity bio tag embeddings and multi-task learning for relation extraction with imbalanced data (2019)