JAD at eHealth-KD Challenge 2021: Simple Neural Network with BERT for Joint Classification of Key-Phrases and Relations

José Gabriel Navarro Comabella[1][0000-0002-6278-2744], Jorge Daniel Valle Diaz [1][0000-0003-4781-2930] and Alberto Helguera Fleitas [1][0000-0002-3043-4534]

[1] School of Math and Computer Science, University of Havana, 10200 Havana, Cuba
tigrejg98@gmail.com;{jorge.valle, alberto.helguera}@estudiantes.matcom.uh.cu

Abstract . This article presents the design choices and training strategy behind the model presented by the JAD team for at eHealth-KD Challenge 2021. The model consist of identifying key-phrases and relations among them using a predefined system. It was a simple model that summarizes some parts of a general approach to NLP problem. The system is easy to train and test using cloud services like Google Colab. It did not perform very well at the competition. The paper includes possible improvements.

Keywords: eHealth, Knowledge Discovery, Natural Language Processing, Machine Learning, Entity Recognition, Relation Extraction, NLP, Simple, BERT, Deep Learning

1 Introduction

This article describes the design choices and training strategy behind the model presented by the JAD team for eHealth-KD Challenge 2021[1]. An annotation scheme for key-phrases and relations was given, with labelled examples for training and evaluation of the model used. The challenge (Main Task) was formed by several subtasks:

- Subtask A (Entity recognition): Given a list of eHealth documents written in Spanish, the goal of this subtask is to identify all the entities per document and their types. These entities are all the relevant terms (single word or multiple words) that represent semantically important elements in a sentence.
- Subtask B (Relation extraction): Subtask B continues from the output of Subtask A, by linking the entities detected and labelled in the input document. The purpose of this subtask is to recognize all relevant semantic relationships between the entities recognized.

Our team presented a system with a fairly simple architecture: a pre-trained multilingual BERT[2] which output representation is used by several dense layers. Our model

IberLEF 2021, September 2021, Málaga, Spain.

is intended to summarize a general approach to Natural Language Processing (NLP) problem. It is easy to train and test using cloud services like Google Colab.

The rest of the paper is organized as follows. Section 2 explains in detail the proposed system. The results of the model in the several scenarios evaluated during the eHealth-KD 2021 event are presented in Section 3. Section 4 analyses briefly matters of interest related to the development and performance of the models. Finally, the conclusions of the work are shown in Section 5.

2 System Description

Figure 1 shows a general overview of the proposed system. Our system is based on a pre-trained multi-lingual BERT[2] layer which pre-processes the input sentence and then feeds the generated embeddings to the deep learning model. This model consists of two dense layers followed by a dropout and a dense layer for output. A binary output that can be translated to the desired output format is produced. The loss function used is binary cross entropy and the metric is binary accuracy.

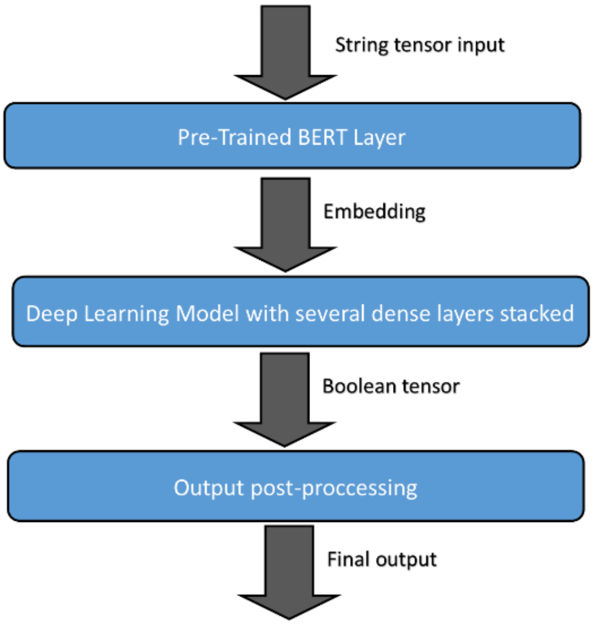

Fig. 1. Model Diagram

2.1 Output format

The output consists of a simple array/tensor of binary values with predefined length. This array may be partitioned in two sub-arrays, the first consisting in the key-phrases annotation and the second consisting in the relations annotations.

Figure 2 shows the key-phrases array, that can be re-interpreted to a 2D array in which for each valid index pair i,j:

- i: Represents the key-phrase label position in a custom fixed-length array of custom labels
- j: Represents the word position in the sentence with a maximum position of the 100th word

If an element is true, it is possible to find the word and the label that matches it.

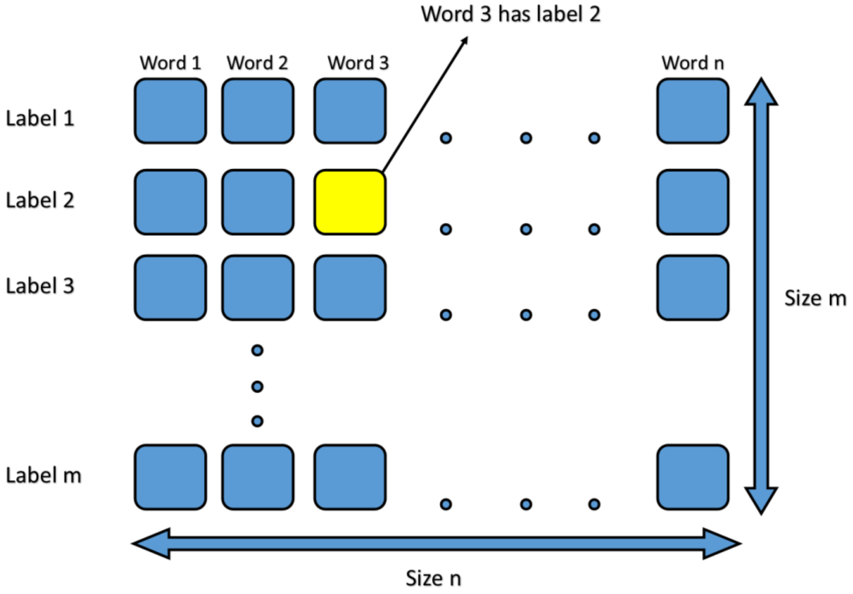

Fig. 2. Key-phrases array

Figure 3 shows the relations array, that could be parsed to a 3D array in which for each valid index pair i,j,k:

- i: Represents the relation label position in a fixed-length array of custom labels
- j: Represents the word position in the sentence with a maximum position of the 100th origin word
- k: Represents the word position in the sentence with a maximum position of the 100th destination word

If an element is true, it is possible to find the origin word, destiny word, and the label that matches it.

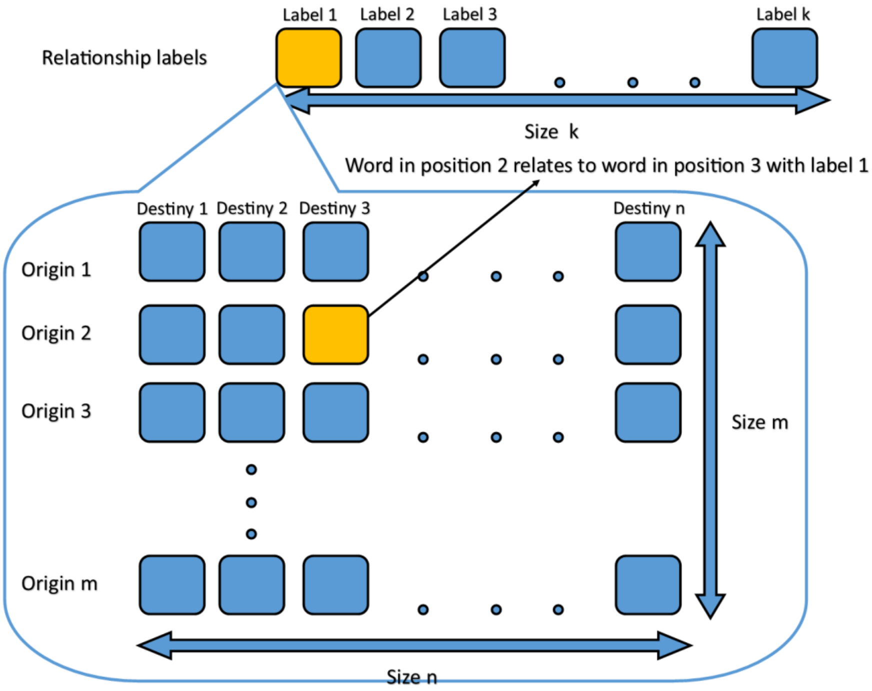

Fig. 3. Relations array

Output processing. This output format must be parsed to the competition output format and vice versa. There are certain problems that were addressed:

- The model output just considers individual words as key-phrase, and the competition output may require several words: This was addressed including a new type of relation called *samebox* which is reflexive and links every word that should be in the same key-phrase. Also several words key-phrase's relations were copied to every single word key-phrase.
- Inconsistencies in the model output: This was addressed by a permissive parsing to the competition output that solved many inconsistencies by itself.

2.2 Training

Our system was trained using a Google Colab notebook with GPU in Python with Keras and Tensor Flow with a binary cross entropy loss function. The training task with all its hyperparameters is available in Google Colab for reference at https://colab.research.google.com/drive/1L0AG1fD9dHzVlv8i-cOc1OruCiedggKG?usp=sharing

Figure 4 shows the accuracy achieved by our model after each training epoch. Accuracy can not be observed well enough but due to the elevated amount of false values in the target output is reasonable that after not many epochs is difficult to interpret improvement due to small improvements in the terms of less than 1%.

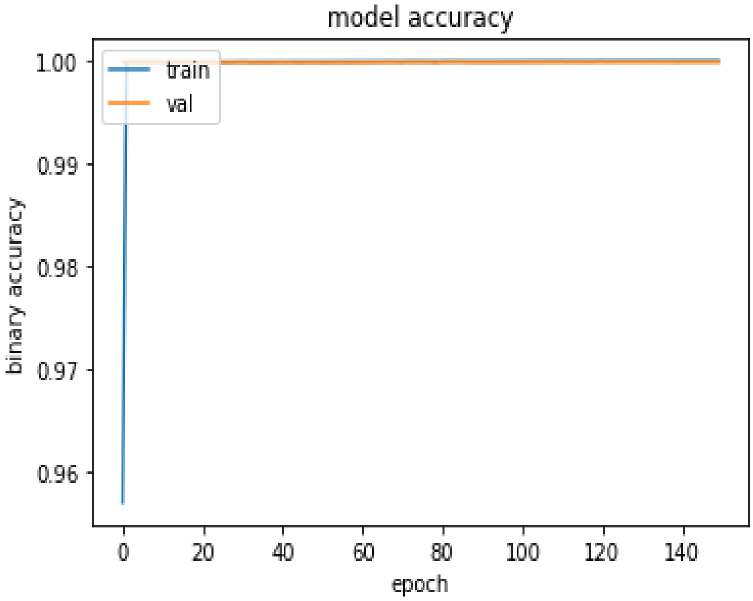

Fig. 4. Model binary accuracy

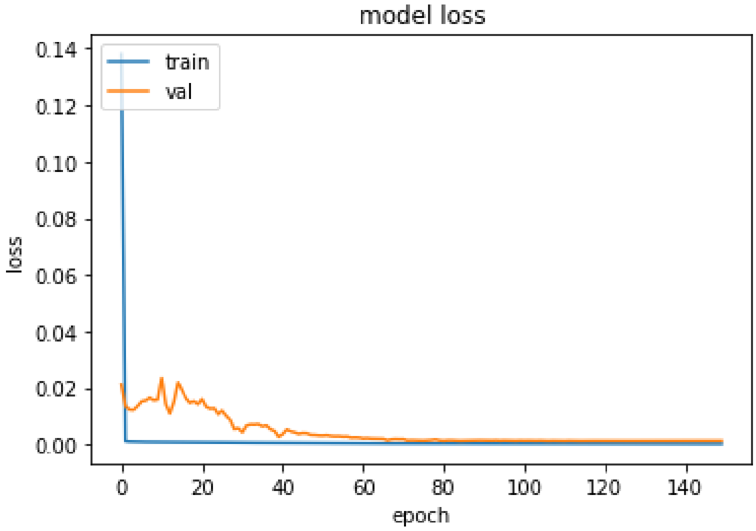

Fig. 5. Model loss

Figure 5 shows the loss achieved by our model after each training epoch. Loss is a little easier to observe in these graphics. It diminishes with time, but dramatically faster in the training set, and slower in the evaluation set. Both sets get closer to 0 with many epochs, but after a while, around 130-145 epochs there is no improvement in evaluation set.

3 Results

Tables 1, 2, and 3 summarize the metrics of each team system in the different tasks of the competition, and ranks the teams according to F1.

Table 1. (Main Task)

	Team	**F1**	**Precision**	**Recall**
🥇	Vicomtech	0.53106	0.54075	0.53464
🥈	PUCRJ-PUCPR-UFMG	0.52835	0.56849	0.50276
🥉	IXA	0.49886	0.46457	0.53863
	uhKD4	0.42264	0.48529	0.37431
	UH-MMM	0.33865	0.29163	0.40374
	CodestrangeTeam	0.23201	0.33703	0.17689
	baseline	0.23201	0.33703	0.17689
	JAD	***0.10949***	***0.23441***	***0.07143***

Table 2. (Task A)

	Team	**F1**	**Precision**	**Recall**
🥇	PUCRJ-PUCPR-UFMG	0.70601	0.71491	0.69733
🥈	Vicomtech	0.68413	0.69987	0.74706
🥉	IXA	0.65333	0.61372	0.6984
	UH-MMM	0.60769	0.54604	0.68503
	uhKD4	0.52728	0.51751	0.53743
	Yunnan-Deep	0.33406	0.52036	0.24599
	baseline	0.30602	0.35034	0.27166
	JAD	***0.2625***	***0.31579***	***0.2246***
	Yunnan-1	0.17322	0.27107	0.12727
	CodestrangeTeam	0.08019	0.415	0.04439

Table 3. (Task B)

	Team	**F1**	**Precision**	**Recall**
🥇	IXA	0.4304	0.45357	0.40948
🥈	Vicomtech	0.37191	0.54186	0.28311
🥉	uhKD4	0.31771	0.55623	0.22236
	PUCRJ-PUCPR-UFMG	0.26324	0.36659	0.20535
	UH-MMM	0.05384	0.07727	0.04131
	CodestrangeTeam	0.03275	0.4375	0.01701
	baseline	0.03275	0.4375	0.01701
	JAD	***0.00722***	***0.375***	***0.00365***

The evaluation in both tasks was carried out using the annotated corpus proposed in the challenge. The results were measured with a standard F1 measure as described in detail in the challenge overview [1]. Also, precision and recall measures were recorded and presented.

From tables 1 to 3 our team always ranked 8th according to F1. Results in Task A were superior to results in Task B. None of our team results performed better than baseline.

4 Discussion

The system achieved poor results in the challenge. The system was non-performant finding relations, and a little better labelling key-phrases. In both cases it was worse than baseline. These results could be due to several reasons:

One of the reasons could be simplicity of the model. If we add recurrent or convolutional layers the results may improve. Such improvement in the long run in the evaluation set could also indicate that the evaluation set was too similar to the training set. The way that the outputs were modeled might be improved to include less negative values. If we divide the model in two separate models also the results could improve.

The accuracy and loss achieved in training was not too good but in the final dataset the model performed poorly. In fact these metrics were a lot better in the training set than in the evaluation set, but the most relevant is the evaluation set because it contains data that our model has not trained on. This last one metrics may look well but in reality, these are not that good. Binary accuracy got to 0.997, but we should remember our output format is large in parameters size so this could imply not such a great performance.

5 Conclusions

This paper describes the system presented by team JAD, in the eHealth-KD Challenge 2021. A deep-learning model was trained and ensembled to automatically extract relevant entities and relations, from plain text documents. The results achieved by the system in the challenge were not outstanding, ranking last in the main task, being better at classifying entities, but still worse than baseline.

The main goal of our team was not winning the competition but to build knowledge and a general and simple model easy to understand, implement, train and run. This goal was completed mostly. The power of a simple model was overvalued. It would be interesting to add some recurrent layers, re-train BERT in a more specific dataset, and reducing output size or changing output format.

6 Acknowledgements

We thank all the organizers, staff and reviewers of the eHealth-KD Challenge 2021. Special thanks for the professors of the AI Department of MATCOM-UH.

References

1. A. Piad-Morffis, Y. Gutiérrez, S. Estevez-Velarde, Y. Almeida-Cruz, R. Muñoz, A. Montoyo, Overview of the eHealth Knowledge Discovery Challenge at IberLEF 2021, Procesamiento del Lenguaje Natural 67 (2021).
2. J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).