TASS 2018: Workshop on Semantic Analysis at SEPLN, septiembre 2018, págs. 71-76

LABDA at TASS-2018 Task 3: Convolutional Neural Networks for Relation Classification in Spanish eHealth documents

LABDA en TASS-2018 Task 3: Redes Neuronales Convolucionales para Clasificación de Relaciones en documentos electrónicos de salud en español

Víctor Suárez-Paniagua[1], **Isabel Segura-Bedmar**[2], **Paloma Martínez**[3]
Computer Science Department
Carlos III University of Madrid
Leganés 28911, Madrid, Spain
[1]vspaniag,[2]isegura,[3]pmf@inf.uc3m.es

Resumen: Este trabajo presenta la participación del equipo LABDA en la subtarea de clasificación de relaciones entre dos entidades identificadas en documentos electronicos de salud (eHealth) escritos en español. Usamos una Red Neuronal Convolucional con el *word embedding* y el *position embedding* de cada palabra para clasificar el tipo de la relación entre dos entidades de la oración. Anteriormente, este método de aprendizaje automático ya ha mostrado buen rendimiento para capturar las características relevantes en documentos electronicos de salud los cuales describen relaciones. Nuestra arquitectura obtuvo una F1 de 44.44 % en el escenario 3 de la tarea, llamada como Setting semantic relationships. Solo cinco equipos presentaron resultados para la subtarea. Nuestro sistema alcanzó el segundo F1 más alto, siendo muy similar al resultado más alto (micro F1=44.8 %) y más alto que el resto de los equipos. Una de las principales ventajas de nuestra aproximación es que no requiere ningún recurso de conocimiento externo como características.

Palabras clave: Extraccion de relaciones, aprendizaje profundo, redes neuronales convolucionales, textos biomédicos

Abstract: This work presents the participation of the LABDA team at the subtask of classification of relationships between two identified entities in electronic health (eHealth) documents written in Spanish. We used a Convolutional Neural Network (CNN) with the word embedding and the position embedding of each word to classify the type of the relation between two entities in the sentence. Previously, this machine learning method has already showed good performance for capturing the relevant features in electronic health documents which describe relationships. Our architecture obtained an F1 of 44.44 % in the scenario 3 of the shared task, named as Setting semantic relationships. Only five teams submitted results for this subtask. Our system achieved the second highest F1, being very similiar to the top score (micro F1=44.8 %) and higher than the remainig teams. One of the main advantage of our approach is that it does not require any external knowledge resource as features.

Keywords: Relation Classification, Deep Learning, Convolutional Neural Network, biomedical texts

1 Introduction

Nowadays, there is a high increase in the publication of scientific articles every year, which demonstrates that we are living in an emerging knowledge era. This explosion of information makes it nearly impossible for doctors and biomedical researchers to keep up to date with the literature in their fields. The development of automatic systems to extract and analyse information from electronic health (eHealth) documents can significantly reduce the workload of doctors.

The TASS workshop proposes shared tasks on sentiment analysis in Spanish each year. Concretely, the goal of TASS-2018 Task 3 (Martínez-Cámara et al., 2018) is to create a competition where Natural Language Processing (NLP) experts can train their sys-

ISSN 1613-0073

tems for extracting the relevant information in Spanish eHealth documents and evaluate them in a objective and fair way.

Recently, Deep Learning has had a big impact on NLP tasks becoming the state-of-the-art technique. Convolutional Neural Network (CNN) is a Deep Learning architecture which has shown good performance in Computer Vision task such as image classification (Krizhevsky, Sutskever, y Hinton, 2012) and face recognition (Lawrence et al., 1997).

The system described in (Kim, 2014) was the first work to use a CNN for a NLP task. It created a vector representation for each sentence by extracting the relevant information with different filters in order to classify them into predefined categories obtaining good results. In addition, CNN was used with good performance for relation classification between nominals in the work of (Zeng et al., 2014). Furthermore, this architecture has been also used in the biomedical domain for the extraction of drug-drug interactions in (Suárez-Paniagua, Segura-Bedmar, y Martínez, 2017a). This system did not require any external biomedical knowledge in order to provide very close results to those obtained using lots of hand-crafted features. We also employed the same approach of (Suárez-Paniagua, Segura-Bedmar, y Martínez, 2017b), which was used for extracting relationships between keyphrases in the Semeval-2017 Task 10: ScienceIE (Augenstein et al., 2017), which proposed very similar subtasks than those defined in TASS-2018 Task 3.

In this work, we describe the participation of the LABDA at the subtask C in the classification of relationships between two identified entities in Spanish documents about health. In this subtask, the test dataset includes the text, the boundaries and the types of their entities to generate the prediction.

2 Dataset

The task provides an annotated corpus from MedlinePlus documents which is divided into the training set for the learning step, development set for the validation and test set for the evaluation of the systems.

The relationship between entities defined as concepts are: *is-a*, *part-of*, *property-of* and *same-as*. There are also relationships defined as roles: *subject* and *target*. The training set contains 559 sentences with 3,276 entities, 1,012 relations and 1,385 roles, the development set contains another 285 sentences. The dataset contains 3,276 entities and 1,012 relations and 1,385 roles in the train set, the development set contains 285 sentences. A detailed description of the method used to collect and process documents can be found in (Martínez-Cámara et al., 2018).

Unlike the other two previous subtasks, the documents include annotated entities with boundaries and types. In this way, it is possible to measure and compare the different approaches only focusing on the goal of the subtask C.

2.1 Pre-processing phase

As some of the relationships types are asymmetrical, for each pair of entities marked in the sentence, we generate two instances. Thus, a sentence with n entities will have $(n - 1) \times n$ instances. Each instance is labelled with one of the six classes *is-a*, *part-of*, *property-of*, *same-as*, *subject* and *target*. In addition, a *None* class is also considered for the non-relationship between the entities. Due to the fact that there are some overlapped entities, we consider each sentence as a graph where the vertices are the entities and the edges are the non-overlapped entities with itself in order to obtain recursively all the possible paths without overlapping, thus we have different instances for each overlapped entities. Table 2 shows the resulting number of instances for each class on the train, validation and test sets.

Label	Train	Validation	Test
is-a	238	299	41
part-of	222	171	36
property-of	600	366	84
same-as	42	19	8
subject	1018	636	206
target	1510	988	308
None	27112	20631	5265

Tabla 2: Number of instances for each relationship type in each dataset: train, validation and test.

After that, we tokenize and clean the sentences following a similar approach as that described in (Kim, 2014), converting the numbers to a common name, words to lower-case, replacing special Spanish accents to Unicode, e.g $\tilde{n}$ to n, and separating special characters with white spaces by regular

Relationship between entities	Instances after entity blinding	Label
(ataque de asma → produce)	*'un entity1 se entity2 cuando los entity0 entity0 .'*	*None*
(ataque de asma ← produce)	*'un entity2 se entity1 cuando los entity0 entity0 .'*	*target*
(ataque de asma → síntomas)	*'un entity1 se entity0 cuando los entity2 entity0 .'*	*None*
(ataque de asma ← síntomas)	*'un entity2 se entity0 cuando los entity1 entity0 .'*	*None*
(ataque de asma → empeoran)	*'un entity1 se entity0 cuando los entity0 entity2 .'*	*None*
(ataque de asma ← empeoran)	*'un entity2 se entity0 cuando los entity0 entity1 .'*	*None*
(produce → síntomas)	*'un entity0 se entity1 cuando los entity2 entity0 .'*	*None*
(produce ← síntomas)	*'un entity0 se entity2 cuando los entity1 entity0 .'*	*None*
(produce → empeoran)	*'un entity0 se entity1 cuando los entity0 entity2 .'*	*subject*
(produce ← empeoran)	*'un entity0 se entity2 cuando los entity0 entity1 .'*	*None*
(síntomas → empeoran)	*'un entity0 se entity0 cuando los entity1 entity2 .'*	*None*
(síntomas ← empeoran)	*'un entity0 se entity0 cuando los entity2 entity1 .'*	*target*
(asma → produce)	*'un ataque de entity1 se entity2 cuando los entity0 entity0 .'*	*None*
(asma ← produce)	*'un ataque de entity2 se entity1 cuando los entity0 entity0 .'*	*None*
(asma → síntomas)	*'un ataque de entity1 se entity0 cuando los entity2 entity0 .'*	*None*
(asma ← síntomas)	*'un ataque de entity2 se entity0 cuando los entity1 entity0 .'*	*None*
(asma → empeoran)	*'un ataque de entity1 se entity0 cuando los entity0 entity2 .'*	*None*
(asma ← empeoran)	*'un ataque de entity2 se entity0 cuando los entity0 entity1 .'*	*None*
(produce → síntomas)	*'un ataque de entity0 se entity1 cuando los entity2 entity0 .'*	*None*
(produce ← síntomas)	*'un ataque de entity0 se entity2 cuando los entity1 entity0 .'*	*None*
(produce → empeoran)	*'un ataque de entity0 se entity1 cuando los entity0 entity2 .'*	*subject*
(produce ← empeoran)	*'un ataque de entity0 se entity2 cuando los entity0 entity1 .'*	*None*
(síntomas → empeoran)	*'un ataque de entity0 se entity0 cuando los entity1 entity2 .'*	*None*
(síntomas ← empeoran)	*'un ataque de entity0 se entity0 cuando los entity2 entity1 .'*	*target*

Tabla 1: Instances with two different entities relationship after the pre-processing phase with entity blinding of the sentence *'Un ataque de asma se produce cuando los síntomas empeoran.'*.

expressions.

Furthermore, the two target entities of each instance are replaced by the labels "*entity1*", "*entity2*", and by "*entity0*"for the remaining entities. This method is known as entity blinding, and supports the generalization of the model. For instance, the sentence in Figure 1: *'Un ataque de asma se produce cuando los síntomas empeoran.'* with the entities *ataque de asma*, *asma*, *produce*, *síntomas* and *empeoran* should be transformed to the relation instances showed in Table 1.

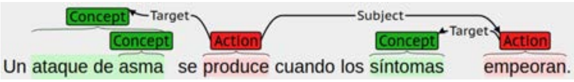

Figura 1: Relationships and entities in the sentence *'Un ataque de asma se produce cuando los síntomas empeoran.'*.

We observed that there are some instances that involve relationships between an entity and its overlapped entity, for this reason, we remove them from the dataset because we can not deal with these relations in the entity blinding process. Moreover, there are relationships with more than one label, in this case, we take just one label because our system is not able to cope with a multi-class problem.

3 CNN model

In this section, we present the CNN architecture which is used for the task of relation extraction in electronic health documents. Figure 2 shows the entire process of the CNN starting from a sentence with marked entities to return the prediction.

3.1 Word table layer

After the pre-processing phase, we created an input matrix suitable for the CNN architecture. The input matrix should represent all training instances for the CNN model; therefore, they should have the same length. We determined the maximum length of the sentence in all the instances (denoted by n), and then extended those sentences with lengths shorter than n by padding with an auxiliary token "*0*".

Moreover, each word has to be represented by a vector. To do this, we randomly initialized a vector for each different word which allows us to replace each word by its word embedding vector: $\mathbf{W}_e \in \mathbb{R}^{|V| \times m_e}$ where V is the vocabulary size and m_e is the word embedding dimension. Finally, we obtained a vector $\mathbf{x} = [x_1; x_2; ...; x_n]$ for each instance where each word of the sentence is represented by its corresponding word vector from the word embedding matrix. We denote p_1 and p_2 as the positions in the sentence of the two

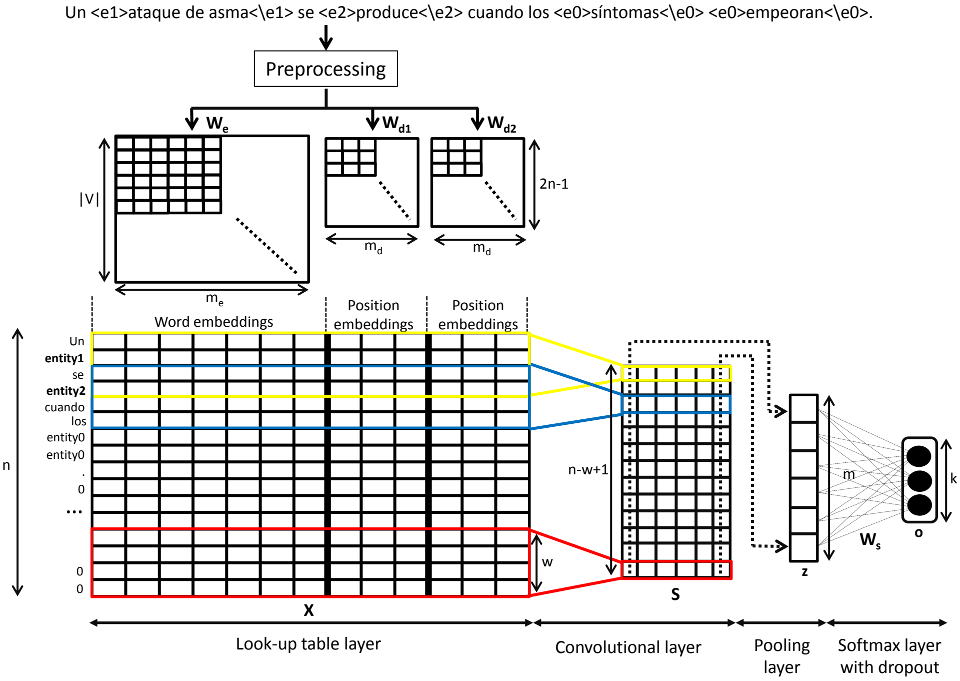

Figura 2: CNN model for the Setting semantic relationships subtask of TASS-2018-Task 3.

entities to be classified.

The following step involves calculating the relative position of each word to the two candidate entities as $i - p_1$ and $i - p_2$, where i is the word position in the sentence (padded word included), in the same way as (Zeng et al., 2014). In order to avoid negative values, we transformed the range $(-n + 1, n - 1)$ to the range $(1, 2n - 1)$. Then, we mapped these distances into a real value vector using two position embeddings $\mathbf{W}_{d1} \in \mathbb{R}^{(2n-1)\times m_d}$ and $\mathbf{W}_{d2} \in \mathbb{R}^{(2n-1)\times m_d}$. Finally, we created an input matrix $\mathbf{X} \in \mathbb{R}^{n\times(m_e+2m_d)}$ which is represented by the concatenation of the word embeddings and the two position embeddings for each word in the instance.

3.2 Convolutional layer

Once we obtained the input matrix, we applied a filter matrix $\mathbf{f} = [f_1; f_2; ...; f_w] \in \mathbb{R}^{w\times(m_e+2m_d)}$ to a context window of size w in the convolutional layer to create higher level features. For each filter, we obtained a score sequence $\mathbf{s} = [s_1; s_2; ...; s_{n-w+1}] \in \mathbb{R}^{(n-w+1)\times 1}$ for the whole sentence as

$$s_i = g(\sum_{j=1}^{w} f_j x_{i+j-1}^T + b)$$

where b is a bias term and g is a non-linear function (such as tangent or sigmoid). Note that in Figure 2, we represent the total number of filters, denoted by m, with the same size w in a matrix $\mathbf{S} \in \mathbb{R}^{(n-w+1)\times m}$. However, the same process can be applied to filters with different sizes by creating additional matrices that would be concatenated in the following layer.

3.3 Pooling layer

In this layer, the goal is to extract the most relevant features of each filter using an aggregating function. We used the max function, which produces a single value in each filter as $z_f = max\{\mathbf{s}\} = max\{s_1; s_2; ...; s_{n-w+1}\}$. Thus, we created a vector $\mathbf{z} = [z_1, z_2, ..., z_m]$, whose dimension is the total number of filters m representing the relation instance. If there are filters with different sizes, their output values should be concatenated in this layer.

3.4 Softmax layer

Prior to performing the classification, we performed a dropout to prevent overfitting. We obtained a reduced vector $\mathbf{z}_d$, randomly setting the elements of $\mathbf{z}$ to zero with a probability p following a Bernoulli distribution. After that, we fed this vector into a fully connected softmax layer with weights $\mathbf{W}_s \in \mathbb{R}^{m\times k}$ to compute the output prediction values for the classification as $\mathbf{o} = \mathbf{z}_d\mathbf{W}_s + d$ where d is a bias term; we have $k = 6$ classes in the

Label	Correct	Missing	Spurious	Precision	Recall	F1
is-a	8	61	8	50 %	11.59 %	18.82 %
part-of	5	27	5	50 %	15.63 %	23.81 %
property-of	9	53	12	42.86 %	14.52 %	21.69 %
same-as	1	4	0	100 %	20 %	33.33 %
subject	50	87	37	57.47 %	36.5 %	44.64 %
target	113	99	72	61.08 %	53.3 %	56.93 %
Scenario 3	186	331	134	58.12 %	35.98 %	44.44 %

Tabla 3: Results over the test set using a CNN with position embedding.

dataset and the "*None*çlass. At test time, the vector **z** of a new instance is directly classified by the softmax layer without a dropout.

3.5 Learning

For the training phase, we need to learn the CNN parameter set $\theta = (\mathbf{W}_e, \mathbf{W}_{d1}, \mathbf{W}_{d2}, \mathbf{W}_s, d, \mathbf{F}_m, b)$, where $\mathbf{F}_m$ are all of the m filters **f**. For this purpose, we used the conditional probability of a relation r obtained by the softmax operation as

$$p(r|\mathbf{x}, \theta) = \frac{\exp(\mathbf{o}_r)}{\sum_{l=1}^{k} \exp(\mathbf{o}_l)}$$

to minimize the cross entropy function for all instances $(\mathbf{x}_i, y_i)$ in the training set T as follows

$$J(\theta) = \sum_{i=1}^{T} \log p(y_i|\mathbf{x}_i, \theta)$$

In addition, we minimize the objective function by using stochastic gradient descent over shuffled mini-batches and the Adam update rule (Kingma y Ba, 2014) to learn the parameters.

4 Results and Discussion

The CNN model was training with the training set and we obtained the best values of each parameters fine-tuning them on the validation set (see Table 4).

The results were measured with precision (P), recall (R) and F1, defined as:

$$P = \frac{C}{C+S} \quad R = \frac{C}{C+M} \quad F1 = 2\frac{P \times R}{P+R}$$

where Correct (C) are the relations that matched to the test set and the prediction, Missing (M) are the relations that are in the test set but not in the prediction, and Spurious (S) are the relations that are in the prediction but not in the test set.

Parameter	Value
Maximal length in the dataset, n	38
Word embeddings dimension, M_e	300
Position embeddings dimension, M_d	10
Filter window sizes, w	3, 4, 5
Filters for each window size, m	200
Dropout rate, p	0.5
Non-linear function, g	ReLU
Mini-batch size	50
Learning rate	0.001

Tabla 4: The CNN model parameters and their values used for the results.

Table 3 shows the results of the CNN configuration with position embeddings. We observe that the number of Missing is very high. This may be due to the fact that the dataset is very unbalanced and these instances are classified as *None* by the system. In fact, we see that the classes that are more representative have better Recall. To solve this problem we propose to use sampling techniques to increase the number of instances of the less representative classes.

Only five teams submitted results for this subtask. Our system achieved the second highest F1, being very similiar to the top score (micro F1=44.8 %), but very much higher than the other teams, which are bellow than 11 % of F1. One of the main advantage of our approach is that it does not require any external knowledge resource.

5 Conclusions and Future work

In this paper, we propose a CNN model for the subtask C (Setting semantic relationships) of the TASS-2018 Task 3. The official results for this model show that the CNN is a very promising system because neither expert domain knowledge nor external features are needed. The configuration of the architecture is very simple with a basic preprocessing adapted for Spanish documents.

The results show that the system produces very many false negatives. We think that this may be due to the unbalanced nature of the dataset. To solve this problem, we propose to use oversampling techniques to increase the number of instances of the less representative classes. Our system also seems to have difficulties in order to distinguish the directionality of the relationships. For these reasons, we will use more complex settings of the architecture for tackling the directionality problem.

Moreover, we plan to use external features as part of the embeddings such as the entity labels given by the second subtask, the Part-of-Speech (PoS) tags and the dependency types of each word for the Spanish documents in order to increase the information of each sentence. We want to explore in detail each feature contribution and the fine-tune all the parameters. Furthermore, we will use some rules to distinguish the relations and the roles with the entity labels and train two different classifier, thus, they would be more accurate. In addition, we will use another neural network architectures like the Recurrent Neural Network and possible combinations with the CNN.

Funding

This work was supported by the Research Program of the Ministry of Economy and Competitiveness - Government of Spain, (DeepEMR project TIN2017-87548-C2-1-R).

Bibliografía

Augenstein, I., M. Das, S. Riedel, L. Vikraman, y A. McCallum. 2017. Semeval 2017 task 10: Scienceie - extracting keyphrases and relations from scientific publications. En *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, páginas 546–555, Vancouver, Canada, August. Association for Computational Linguistics.

Kim, Y. 2014. Convolutional neural networks for sentence classification. En *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, páginas 1746–1751.

Kingma, D. P. y J. Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Krizhevsky, A., I. Sutskever, y G. E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. En *Advances in Neural Information Processing Systems 25*, páginas 1097–1105. Curran Associates, Inc.

Lawrence, S., C. L. Giles, A. C. Tsoi, y A. D. Back. 1997. Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8(1):98–113, Jan.

Martínez-Cámara, E., Y. Almeida-Cruz, M. C. Díaz-Galiano, S. Estévez-Velarde, M. A. García-Cumbreras, M. García-Vega, Y. Gutiérrez, A. Montejo-Ráez, A. Montoyo, R. Muñoz, A. Piad-Morffis, y J. Villena-Román. 2018. Overview of TASS 2018: Opinions, health and emotions. En E. Martínez-Cámara Y. Almeida Cruz M. C. Díaz-Galiano S. Estévez Velarde M. A. García-Cumbreras M. García-Vega Y. Gutiérrez Vázquez A. Montejo Ráez A. Montoyo Guijarro R. Muñoz Guillena A. Piad Morffis, y J. Villena-Román, editores, *Proceedings of TASS 2018: Workshop on Semantic Analysis at SEPLN (TASS 2018)*, volumen 2172 de *CEUR Workshop Proceedings*, Sevilla, Spain, September. CEUR-WS.

Suárez-Paniagua, V., I. Segura-Bedmar, y P. Martínez. 2017a. Exploring convolutional neural networks for drug-drug interaction extraction. *Database*, 2017:bax019.

Suárez-Paniagua, V., I. Segura-Bedmar, y P. Martínez. 2017b. LABDA at semeval-2017 task 10: Relation classification between keyphrases via convolutional neural network. En *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, páginas 969–972.

Zeng, D., K. Liu, S. Lai, G. Zhou, y J. Zhao. 2014. Relation classification via convolutional deep neural network. En *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014), Technical Papers*, páginas 2335–2344, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.