Suárez-Paniagua, V., Rivera, R.M., Segura-Bedmar, I., Martínez, P. (2019). A two-stage deep learning approach for extracting entities and relationships from medical texts. *Journal of Biomedical Informatics*, 99, 103285

DOI: 10.1016/j.jbi.2019.103285

# A two-stage deep learning approach for extracting entities and relationships from medical texts

Víctor Suárez-Paniagua[a], Renzo M. Rivera Zavala[b], Isabel Segura-Bedmar[c], Paloma Martínez[d]

*Computer Science Department,*
*Carlos III University of Madrid,*
*Leganés 28911, Madrid, Spain*

[a] *vspaniag@inf.uc3m.es*
[b] *renzomauricio.rivera@alumnos.uc3m.es*
[c] *isegura@inf.uc3m.es*
[d] *pmf@inf.uc3m.es*

**Abstract**

This work presents a two-stage deep learning system for Named Entity Recognition (NER) and Relation Extraction (RE) from medical texts. These tasks are a crucial step to many natural language understanding applications in the biomedical domain. Automatic medical coding of electronic medical records, automated summarizing of patient records, automatic cohort identification for clinical studies, text simplification of health documents for patients, early detection of adverse drug reactions or automatic identification of risk factors are only a few examples of the many possible opportunities that the text analysis can offer in the clinical domain. In this work, our efforts are primarily directed towards the improvement of the pharmacovigilance process by the automatic detection of drug-drug interactions (DDI) from texts. Moreover, we deal with the semantic analysis of texts containing health information for patients. Our two-stage approach is based on Deep Learning architectures. Concretely, NER is performed combining a bidirectional Long Short-Term Memory (Bi-LSTM) and a Conditional Random Field (CRF), while RE applies a Convolutional Neural Network (CNN). Since our approach uses very few language resources, only the pre-trained word embeddings, and does not exploit any domain resources (such as dictionaries or ontologies), this can be easily expandable to support other languages and clinical applications that require the exploitation of semantic information (concepts and relationships) from texts.

*Preprint submitted to Journal of Biomedical Informatics* *August 29, 2019*

During the last years, the task of DDI extraction has received great attention by the BioNLP community. However, the problem has been traditionally evaluated as two separate subtasks: drug name recognition and extraction of DDIs. To the best of our knowledge, this is the first work that provides an evaluation of the whole pipeline. Moreover, our system obtains state-of-the-art results on the eHealth-KD challenge, which was part of the Workshop on Semantic Analysis at SEPLN (TASS-2018).

*Keywords:*
Name Entity Recognition, Relation Extraction, Deep Learning, health documents

## 1. Introduction

Natural Language Processing (NLP) and Information Extraction (IE) can bring tremendous benefits in the biomedical and clinical domains. The automated semantic analysis of information offers an effective way to acquire knowledge from unstructured texts. Coding of electronic medical records, summarization of patient records, text simplification of health documents for patients, cohort identification for clinical studies, early identification of adverse drug reactions or risk factors are only a few examples of the many possible applications in the clinical domain that can benefit from applying the NLP technology [1].

During the last decade, the NLP community have made tremendous advances in IE. Probably, biomedical and clinical domains have been ones of the most explored fields due to the numerous shared tasks organized in the last ten years. Starting from the pioneer BioCreative [2] until the most recent NLP clinical Challenges (n2c2)[1], all these tasks have contributed significantly to advance the knowledge of NLP and IE methods for analysing medical texts. The DDIExtraction shared tasks [3, 4, 5] were one of the first efforts to provide a framework for the reliable and fair evaluation and comparison of systems for the detection and classification of drug names and extraction of drug-drug interactions (DDI), a particular type of adverse drug reaction, from medical texts. Detecting this type of information is crucial to improve the pharmacovigilance systems, whose primary mission is the prevention of secondary effects, adverse effects or other problem related to drugs.

[1] https://n2c2.dbmi.hms.harvard.edu/

The Pharmacovigilance field has also gained increasing attention within the BioNLP community. The special issue titled *Mining the Pharmacovigilance Literature* [6] collects some of the main works about using NLP for pharmacovigilance. In the last years, several shared tasks have been organized to foster research on the automatic detection of information related to adverse drug reactions (ADR). TAC 2017 ADR [7] proposed the evaluation of systems for adverse drug reactions extraction from drug labels, while the second Social Media Mining for Health Research and Applications Workshop dealt with the detection of adverse drug reactions from tweets [8]. The interest in this kind of shared tasks grows every year in the BioNLP community. At the moment that we write this paper, August 2018, a track about ADR and Medication Extraction in clinical narratives [2], organized by the Harvard Medical School Department of Biomedical Informatics (DBMI) and the Volgenau School of Engineering of George Mason University, and a new edition of TAC dedicated to DDI extraction are being conducted [3].

All these competitions have focused on English, while very few efforts have been made to support the research activity for extracting relevant information from medical texts written in other languages than English. To the best of our knowledge, eHealth-KD challenge [9], which is part of the Workshop on Semantic Analysis at SEPLN (TASS-2018) [4], has been the first initiative to promote the development of information extraction techniques to automatically extract knowledge from eHealth documents written in the Spanish language. The documents were taken from MedLinePlus [5], an informative website directed to patients, which offers information about health topics such as medicines and diseases. The shared task proposed the identification and classification of keyphrases, as well as the detection of all relevant semantic relationships between the entities recognized.

In this paper, we present a two-stage system for Named Entity Recognition (NER) and Relation Extraction (RE) from medical texts. These tasks are a crucial step to many natural language understanding applications in the clinical domain. Our efforts are primarily directed towards the improvement of the pharmacovigilance process by the automatic detection of DDI from texts. Moreover, we also deal with the semantic analysis of texts containing

[2]https://n2c2.dbmi.hms.harvard.edu/

[3]https://bionlp.nlm.nih.gov/tac2018druginteractions/

[4]http://www.sepln.org/workshops/tass/2018/

[5]https://medlineplus.gov/spanish/

health information for patients. The proposed two-stage system involves the entire process of classifying relations from raw data using Deep Learning architectures. Concretely, NER is performed combining a bidirectional Long Short-Term Memory (Bi-LSTM) and a Conditional Random Field (CRF) and RE applies a Convolutional Neural Network (CNN). We provide extensive experimentation of our approach on different datasets, the DDI corpus [10] and the dataset used in the eHealth-KD challenge.

During the last decade, many NLP research groups have dedicated numerous efforts to address the problem of DDI extraction, which is evaluated as two separate subtasks: drug name recognition and relation extraction task (extraction of DDIs). Unlike the previous works in DDI extraction, we provide the results obtained by a full IE pipeline involving the two subtasks. Moreover, our system obtains state-of-the-art results on the eHealth-KD challenge. Furthermore, this approach exploits very few language resources such as pre-trained word embeddings and does not use any domain resources. Therefore, our approach is easily expandable to support other languages and clinical applications that require the exploitation of semantic information from texts.

The organization of this paper is as follows. In the next section, we discuss previous works for the NER and RE tasks, and a review of the IE systems. Section 3 describes our two-stage pipeline. In Section 4, we present and discuss the experimental results. Finally, conclusions and potential future work items are identified in Section 5.

## 2. Related work

Automatic knowledge and information extraction are critical issues in biomedical literature. Biomedical knowledge could be useful to improve biomedical research, clinical medicine, biomedical applications and so forth. Named Entity Recognition (NER) and Relation Extraction (RE) are the most important subtasks in information extraction.

### *2.1. Biomedical Named Entity Recognition*

Biomedical Named Entity Recognition (Bio-NER) is the task of detecting biomedical entities mentions in medical texts and classifying them in predefined categories. First approaches in Bio-NER used dictionary and rule-based methods, but they suffer from significant limitations such as low recall, requiring expert domain knowledge, continuous maintenance and low portabil-

ity to other entity types, the inability of dealing with spelling errors, among others. Machine learning (ML) based methods overcome these limitations using mathematical methods and statistical techniques to learn from data and generate predictions or decisions based on data. Conditional Random Field (CRF) is one of the most successful algorithms for NER task. Some of the most representative Bio-NER works based on CRF are ABNER [11], BANNER [12], Chemspot [13], Gimli [14]. However, most works based on machine learning methods focus their efforts in the manual design of hand-crafted features, which are obtained from knowledge resources or by using NLP tools.

It is important to emphasize that the performance of machine learning algorithms highly depends on the selection and representation of the most informative features for the task. Basic features used in NER are linguistic, orthographic, morphological, context and lexical characteristics. Semantic features from terminological resources or existing Bio-NER tools are also widely included. Syntactic features always depend on the performance of NLP tools such as tokenizers, chunkers, PoS taggers or syntactic parsers. The proper selection of the most suitable features for NER always requires experts with strong domain knowledge and are expensive to acquire. Moreover, the feature set for a given NER task cannot be directly applied to other entity types, domain or language.

Deep Learning methods for NER can automatically learn patterns capturing relevant syntactic and semantic information from corpora. Later, these patterns are used as features for the the identification and classification of Named Entities (NE). This fact allows the independence of a specific language or domain. Moreover, these methods do not require a high degree of maintenance. Other advantages of deep learning methods for NER are:

i) they can make predictions about new terms not seen before,

ii) they have a higher tolerance to misspellings,

iii) they can deal with problems of ambiguity.

Recently, Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) are commonly used in NER tasks achieving state-of-art performance. Now, we describe some state-of-the-art systems for BioNER based on Deep Learning methods.

One of the most used RNN model for NER is Long Short-Term Memory (LSTM) due to its property for storing ("remembering") patterns over arbitrary time intervals, and therefore suitable for process and predict time series given sequences of labels and relate parts of a sequence. A hybrid bidi-

rectional LSTM and CNN architecture was applied in [15, 16]. The model proposed by [15] was evaluated on the CoNLL-2003 and on the OntoNotes corpus obtaining an F1 of 91.62 % and 86.28 %, respectively. The system described in by [16] was evaluated on the BioCreative II Gene Mention task corpus (BC2) with an F1 of 80.58%. In addition, it was also evaluated on the BioNLP 2009 event extraction (BioNLP09) task (87.06%) and on the NCBI disease corpus (NCBI) (84.26 %.).

Lample et al. [17] compared a hybrid Bi-LSTM with a CRF layer and a pure stack LSTM (S-LSTM) model. Both architectures relied on character and pre-trained word embeddings. The models were evaluated on the CoNLL-2013 English corpus, CoNLL-2012 German corpus, CoNLL-2012 Dutch corpus, and CoNLL-2012 Spanish corpus. The Bi-LSTM model yielded an F1 of 90.94%, 78.76%, 81.74% and 85.75% for each of the previous datasets. The S-LSTM model provided an F1 of 90.33%, 75.66%, 79.88% and 83.93% for each of the previous datasets.

Other more recent works [18, 19] used a hybrid model combining pre-trained word embedding models, a bidirectional LSTM network with a CRF network (Bi-LSTM-CRF). Unlike LSTM-CNN architecture, a Bi-LSTM network is used to calculate character embeddings. The model proposed by [18] was evaluated on the DDI corpus obtaining an F1 of 79.26%. The model proposes by [19] was evaluated on the JNLPBA and BioCreative II Gene Mention (GM) corpus providing an F1 of 75.87% and 89.46%, respectively.

Ma et al.[20] introduce a novel architecture combining the after mentioned architectures (LSTM-CNN-CRF) using a first CNN layer for extracting character-level representations of words with character embeddings as inputs. Character-level representation vector is concatenated with the word embedding representation vector as the input for a second Bi-LSTM layer. The output vectors of the second Bi-LSTM are the input for the CRF layer for decode the best sequence of labels. The model was evaluated for two different tasks: NER (CoNLL 2003 corpus) and PoS tagging (Penn Treebank WSJ corpus), obtaining a f-score of 91.21% for NER task and a 97.55% accuracy for POS tagging task.

The inputs of deep learning neural networks are numerical vectors that represent the embeddings of the words, their character or their PoS, among other lexical information. Several works [21, 22, 23] have shown that character-level word embeddings can significantly improve learning for specific domains. Moreover, they are useful for morphologically rich languages and can contribute to the recognition of unknown terms.

*2.2. Biomedical Relation Extraction*

We describe the most recent works for relation extraction in the biomedical and clinical domain, with a particular focus on those based on Deep Learning methods.

Most essential works on Relation Extraction in the last decade were based on machine learning algorithms using a large number of hand-crafted features. Mainly, the top system of the DDIExtraction shared task [24] was a linear SVM classifier using a hybrid kernel with features based on syntactic tags, dependency graph, negation cue, bag-of-words of the entities and surrounding words around the target entities in order to generate a relation class prediction. Building these systems requires high domain knowledge and linguistic analysis in order to choose the most suitable feature set for the task. This method obtains a 65.1% in F1 for the classification of relationships between drugs. Deep learning techniques solve this requirement because they learn the relevant features for each instance according to its classification.

The system described in [25] was the first work using a CNN for the classification of DDI sentences. This system uses a model of word embeddings trained from a collection from MEDLINE documents. It created a vector representation for each sentence by extracting the relevant information with different filters in order to classify them into predefined categories outperforming the previous works. It consists of four layers: look-up table layer, convolutional layer, max-pooling layer, and a Softmax layer. They obtained 69.75% in F1 using position embedding and negative instance filtering to discard some sentences with non-relationship target entities.Also for the DDIExtraction task, [26] proposed adding multiple word embeddings from different sources like PubMed, PMC, MedLine and Wikipedia as word embeddings of a CNN (MCNN) to improve the results until 70.21% of F1. [27] built parallel CNNs that take the input from the dependency parse tree and the sequential order of the sentences (CNN+DCNN) achieving an F1 of 70.81%. In [28], a CNN model represents the syntactic information, called SCNN. Concretely, the word embeddings are extended by including the position and the PoS of each word. The last layer combines the convolutional features and traditional features (such as the drug names, their surrounding words, the dependency types, and the biomedical semantic types) forming the input for the Softmax function, which is the classifier. The work reaches an F1 of 68.6% using two sequential models, for detection and classification of drug relationships for the DDIExtraction task. The work described in [29] applied a recurrent network with Bidirectional Long Short Term Memory

Network and obtains and F1 of 71.48% for the DDI classification combining the max-pooling and the attentive pooling (joint AB-LSTM). The authors of obtained the state-of-the-art for this task using 10 convolutional layers applied sub-sequentially [30] taking multiple channels of the word embeddings. The system reaches 86,27% in F1, which is highly superior to the previous systems.

Recently, the International Workshop on Semantic Evaluation (SemEval) organized some evaluations of computational semantic analysis systems for Relation Extraction. Concretely, the goal of SemEval 2017 Task 10: ScienceIE is the automatic extraction of keyphrases and their relationships from scientific publications [31]. The top system for the relation extraction subtask was a CNN with max-pooling that used the word, position, type of entity and POS tags embeddings of the words between the target entities in the sentence in order to generate the class prediction [32] and obtaining 63.8% in F1. Additionally, SemEval-2018 task 7 [**?** ] is focused on the extraction of semantic relationships in scientific papers and defined two subtasks for detection and classification. The architecture of [33] ranked first in the classification subtasks using an ensemble of CNN and LSTM with the word, POS and relative position embeddings of the words in the sentence which achieves 49.3% in F1. For the eHealth-KD challenge, the top system [34] implemented a CNN using the embeddings of the words, their POS tags and relative distances to the target entities resulting in 44.8% in F1 for the relation extraction task. The system, which ranked second [35], used a CNN with the word embedding and position embedding of each word obtaining 44.44% in F1 for the relation classification task of the eHealth-KD challenge.

### *2.3. A two-stage Information Extraction systems from medical texts*

Concerning the research that proposes pipelines that combine NER and RE processes, most works are focused on protein-protein interaction (PPI) extraction. The work reported in [36] describes a three steps method for PPI extraction. The first step uses a multilabel CNN to recognize protein entities, then, a Syntax CNN to extract relational protein pairs and finally, the PPI triples (protein, interaction word, protein) are obtained using a dictionary method complemented by a syntactic pattern method to cope with the missed interaction words. This approach achieves an F1 of 40.18% on an extension of the Aimed corpus [37] annotated with proteins, binary interaction labels and interaction words. This system is not as a complete pipeline because the

annotations from the corpus are used to filter the results for the following phase.

Bunescu et al. [37] presents an evaluation of different information extraction methods for identifying human protein names in Medline abstracts and then recognize PPI using a set of 1000 manually-annotated Medline abstracts. The authors test Dictionary-based, SVM, k-NN, among others, are for the NER task reporting a 70% precision and about 90% recall in the best case using a MaxEnt (Maximum Entropy learning) method that exploits a generalized protein-name dictionary. Thus, different relation extractors are evaluated with this method. The ELCS (longest common subsequences) method taking as input the result of the MaxEnt NER method achieves near 75% recall and approximately 25% precision. In general, machine learning methods outperform methods based on hand-written rules.

The Turku Event Extraction System (TEES) [38] is a pipeline system based on a graph-generation approach that detects events via a set of features built via dependency parsing. It reports an F1 of 60% for the complete pipeline detecting entities and relations on GENIA corpus used in BioNLP Shared Task 2013. This system also participated at DDIExtraction 2013 task in the SemEval conference, but the results are presented on the two steps indepently.

More recently, another approach working on domains other than PPI is the end-to-end LSTM recurrent neural model that captures both word sequence and dependency tree structures [39]. The novelty of this system concerning the previous ones is that entities and relations are jointly modeled in neural architectures by richer linguistic structures. This system has been used in SemEval-2017 ScienceIE task obtaining an F1 of 38% with a pipeline including entity segmentation+entity classification+relation classification. Nevertheless, the winner in the ScienceIE 2017 task with an F1 of 43% is a system based on [40] that incorporates character-level encoding and gazetteers obtaining from external knowledge-based sources (such as Freebase) among other extensions [39].

Finally, a hybrid approach based on Bi-LSTM and CRF [41] was proposed for the TASS-2018 Task 3. eHealth Knowledge Discovery task for NER and RE on Spanish eHealth documents inspired in the SemEval-2017 task. The system obtained an F1 of 46.4% in the complete pipeline including entity detection+entity classification+relation classification. However, the system only addresses entity detection and classification subtasks.

Although deep learning based methods for NER and RE achieve satisfac-

tory results, there is still much room for improvement. On the other hand, full IE systems are scarce, and their results are quite low. That is why in this work we propose a two-stage system based on deep learning methods using different semantic, syntactic, morphological and orthographic embedding features.

## 3. Methods

This section presents the two modules that compose our two-stage IE system (see Figure 1). Our system deals with three different subtasks: A) named entity recognition, B) entity classification and C) relation extraction. Therefore, there are three possible evaluation scenarios:

- Scenario 1: Only plain text is given (subtasks A, B, C).
- Scenario 2: Plain text with manually annotated entity boundaries are given (subtasks B, C).
- Scenario 3: Plain text with manually annotated entities and their types are given (subtask C).

Our approaches for NER and RE, which are described below, are purely based on deep learning.

### *3.1. Named Entity Recognition*

In this section, we describe the module for NER. Our approach is based on a deep network with two Bi-LSTM layers and a last layer for CRF (see Figure 2). The input for the first Bi-LSTM layer are character embeddings. In the second layer, the output of the first layer is concatenated with word embeddings and sense-disambiguate embeddings. Finally, the last layer uses a CRF to obtain the most suitable labels for each token.

Firstly, the system preprocesses the texts in order to create the input for training the neural network. Figure 3 summarizes all preprocessing steps. First, sentences are split and tokenized by using Spacy [42], an open source library for advanced natural language processing with support for 26 languages. BRAT format has become a de facto standard for corpora annotation. Currently, most NER corpora are released using it. BRAT format is a standoff format where each line represents an annotation, such as an entity, a relation or an event. For example, an entity annotation is represented by

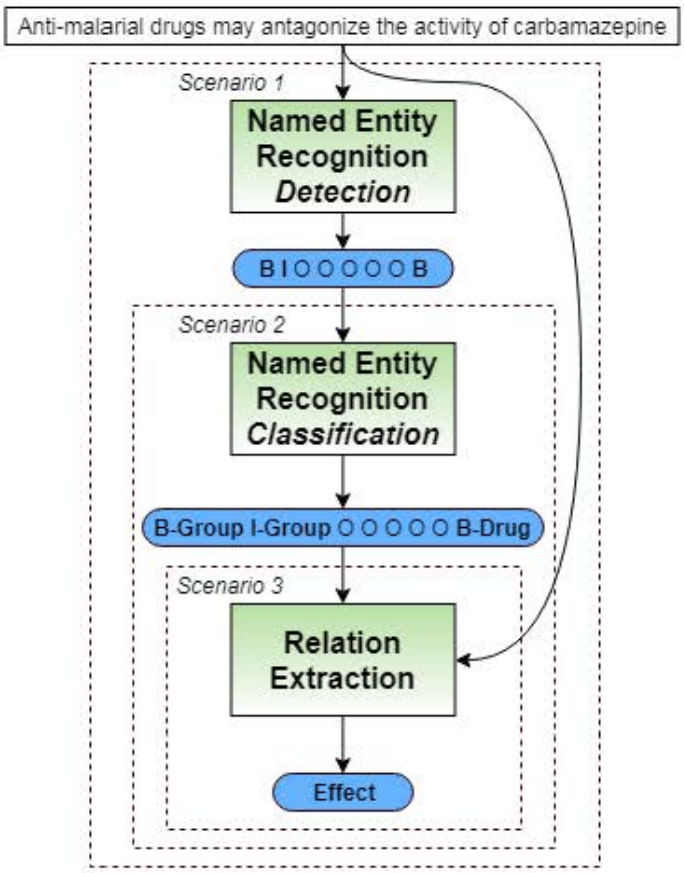


Figure 1: Pipeline of the proposed two-stage system.

a unique ID and defined by its type (for example, Concept, Action, Drug, Brand, Group or Drug-n). This annotation format includes the start and end offsets of the entity mentionand the text of its mention. Additionally, a single TAB character separates the information in each line. We annotate each token in a sententence according the information from the BRAT format using the BMEWO-V extended tag encoding, which allows us to capture information about the sequence of tokens in the sentence.

The BMEWO-V encoding distinguishes the B tag to indicate the start of an entity, the M tag to indicate the continuity of an entity, the E tag to indicate the end of an entity, the W tag for indicate a single entity, and the O tag to represent other tokens that do not belong to any entity. The special tag V represents the overlapping entities. BMEWO-V is similar to other previous

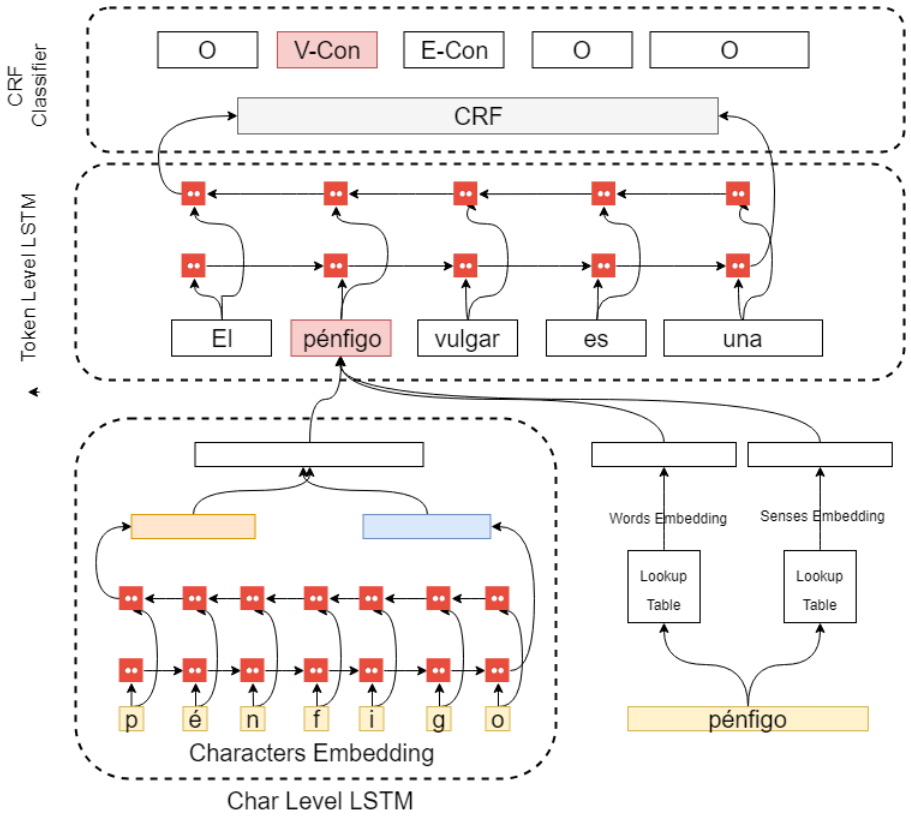


Figure 2: Overview of NER architecture.

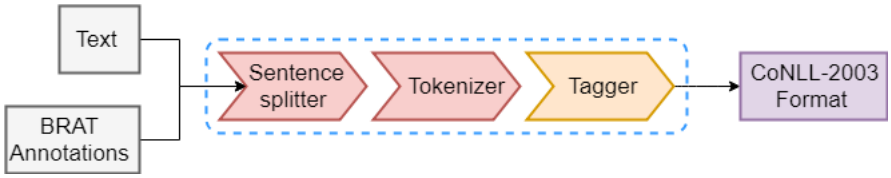


Figure 3: Preprocessing steps required for our NER model.

encodings [43], but it also allows the representation of discontinuous entities, overlapping or nested entities. As a result, we obtain our sentences annotated in CoNLL-2003 format.

Now, we describe the architecture of our deep network for NER task.

*3.1.1. Architecture of our deep neural network for NER*

Recurrent neural networks (RNN) are powerful algorithms and useful for sequential data processing, achieving ground-breaking results in many NLP tasks (e.g., machine translation, NER). Long-Short Term Memory (LSTM) [44] are RNNs variant aimed to deal with gradient vanishing problem. Bidirectional LSTMs (BiLSTM) [45, 46] are capable of learning long-term dependencies and maintaining contextual features from both past and future states while avoiding the vanishing/exploding gradients problem [47, 48]. They consist of two separate bidirectional hidden layers that feed forward to the same output layer.

*First Bi-LSTM layer using character embeddings*

Although word embedding models capture syntactic and semantic information, they do not exploit other linguistic information such as morphological information, orthographic transcription or part-of-speech (POS) tags. According to [23], the use of character embeddings improves learning for specific domains and is useful for morphologically rich languages. For this reason, we decided to consider the character embedding representation in our system to obtain morphological and orthographic information from tokens. We used a 25 features vector to represent each character. In this way, tokens in sentences are represented by their corresponding character embeddings, which are the input for our Bi-LSTM network.

*Second Bi-LSTM layer using word and Sense embeddings*

The output of the first layer is concatenated with the word embeddings and with the sense-disambiguation embeddings of the tokens in a given input sentence. This concatenation of features is the input for the second Bi-LSTM layer.

Currently, there are many pre-trained word embedding models freely available for the NLP community to use. These models are trained on extensive collections of texts such as Google News, MedLine or Wikipedia. In this work, we exploit two different word embeddings models (see Table 1): i) Spanish Billion Words [49], which was trained on different text corpora written in Spanish (such as Ancora Corpus [50] and Wikipedia) , and ii) a pre-trained word embedding GloVe model [51] trained on 2014 Wikipedia and Gigaword 5 edition corpus written in English.

One of the most critical limitations of word embeddings models is that a single word vector represents all possible meanings for a word. In other words,

word embedding models can not distinguish polysemous words correctly. For this reason, in addition to the two word embeddings models, we also exploit a sense embedding model trained with the Sense2Vec tool [52], which provides multiple word vectors for each word based on the meaning of the word. The meaning of the word in this model is defined by the lexical class or Part-of-Speech (PoS) of a word, its context and its relation to the adjacent and related words within a phrase, sentence or paragraph. Sense2Vec analyses the context of a word and then assigns its more adequate vector. The sense embedding model used in this work [52] (see Table 1) was trained using a collection of comments published on Reddit (corresponding to the year 2015), which mostly consists of texts written in English. Unfortunately, there is no a model of pre-trained sense embeddings for Spanish.

| Model | Language | Vocabulary | Algorithm |
|---|---|---|---|
| Spanish Billion Words [49] | Spanish | 1 million | Word2Vec |
| Glove.6B [51] | English | 2 million | Glove |
| Reddit [52] | English | 1 million | Sense2Vec |

Table 1: Pre-trained models used in our work.

The goal of the second layer is to obtain a sequence of probabilities corresponding to each label of the BMEWO-V encoding format. In this way, for each input token, this layer returns six probabilities (one for each tag in BMEWO-V). The final tag should be with the highest probability.

*Last layer based on Conditional Random Fields (CRF)*

To improve the accuracy of predictions, we also used a Conditional Random Field (CRF) [53] model, which takes as input the label probability for each independent token from the previous layer and obtains the most probable sequence of predicted labels based in the correlations between labels and their context. Handling independent labels for each word shows sequence restrictions. For example, the "I-DRUG" tag cannot appear before a "B-DRUG" tag or after a "B-DRUG" tag. CRF is used jointly with Bi-LSTMs to avoid the label independence assumptions of LSTMs and to impose sequence labeling constraints as show in [17]. For a sequence CRF model interactions between two successive labels are considered, training and decoding can be solved efficiently by adopting the Viterbi algorithm.

Finally, once tokens have been annotated with their corresponding labels in the BMEWO-V encoding format, the entity mentions are transformed into

the BRAT format. V tags, which identify nested or overlapping entities, are generated as new annotations within the scope of other mentions.

### *3.2. Relation Extraction*

This subsection describes the approach for the relation extraction task. Figure 4 shows the system overview used for this subtask.

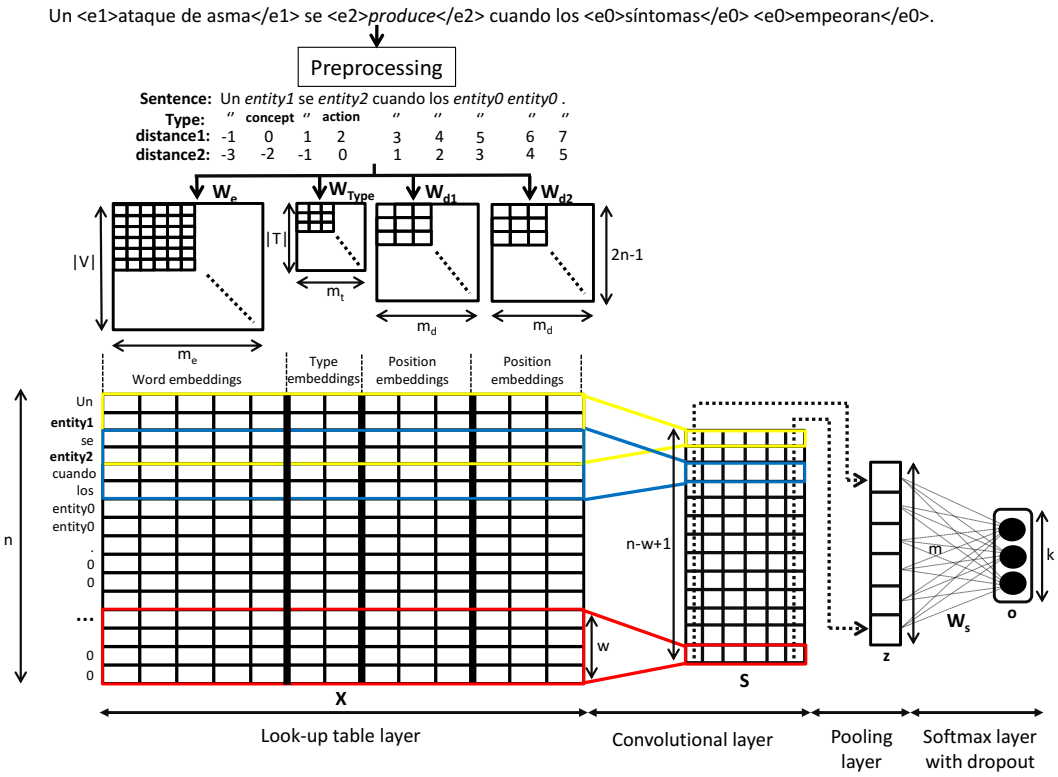


Figure 4: CNN model for relation extraction. English translation: 'An asthma attack occurs when symptoms get worse.'.

As the first step in the process, we must generate all possible relation instances. An instance for a relationship consists on a pair of annotated entities that happen in the same sentence. If the relationship is symmetrical, we must consider pairs as unordered pairs, while if the relationship is asymmetrical, we must consider ordered pairs. Figure 5 shows a sentence taken from the eHealth-KD dataset (described in Section 4), annotated with several entities and their relationships. In this example, the relationships Target and

Subject are asymmetrical. Table 2 shows all possible relation instances and their relation type generated from this sentence. We also consider a *None* type for representing the non-relationship between the entities.

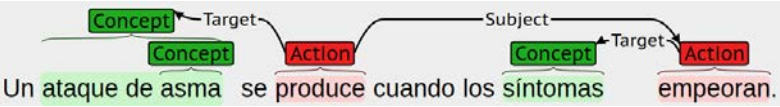


Figure 5: Example of sentence taken from the eHealth-KD dataset. English translation: 'An asthma attack occurs when symptoms get worse.'.

| Relation instances | Relation type |
|---|---|
| **(ataque de asma → produce)** | *None* |
| **(ataque de asma ← produce)** | *target* |
| **(ataque de asma → síntomas)** | *None* |
| **(ataque de asma ← síntomas)** | *None* |
| **(ataque de asma → empeoran)** | *None* |
| **(ataque de asma ← empeoran)** | *None* |
| **(asma → produce)** | *None* |
| **(asma ← produce)** | *None* |
| **(asma → síntomas)** | *None* |
| **(asma ← síntomas)** | *None* |
| **(asma → empeoran)** | *None* |
| **(asma ← empeoran)** | *None* |
| **(produce → síntomas)** | *None* |
| **(produce ← síntomas)** | *None* |
| **(produce → empeoran)** | *subject* |
| **(produce ← empeoran)** | *None* |
| **(síntomas → empeoran)** | *None* |
| **(síntomas ← empeoran)** | *target* |

Table 2: Possible relation instances for the sentence shown in Figure5.

After that, the systems tokenizes and cleans the generated instances following a similar approach as described in [54], converting the numbers to a common name (NUMBER), words to lower-case, replacing Spanish accents to Unicode, for example $\tilde{n}$ to $n$, and separating special characters with white spaces by regular expressions.

Furthermore, the two target entities of each instance are replaced by the labels "*entity1*" and "*entity2*", while the remaining entities by "*entity0*". This method is known as entity blinding, and supports the generalization of the model. For instance, the sentence for the first relation instance **(ataque de asma → produce)** should be transformed to *'un entity1 se entity2 cuando los entity0 entity0 .'*

Medical texts contain some nested entities, which are entities that overlap other entities. For this reason, we create relation instances for each entity mention contained in a nested entity. Moreover, we remove all possible relation instances that involve a relationship between entities in the same nested entity. To do this, we consider each sentence as a graph where the vertices are the entities and the edges are the non-overlapped entities with itself. This graph allows us to obtain all the possible paths without overlapping recursively.

Some entities can contain gaps in their mentions. For example, the noun phrase *"ganglionic or peripheral adrenergic blocking drugs"* contains two different drug entities: *ganglionic adrenergic blocking drugs* and *peripheral adrenergic blocking drugs*. The first one is a discontinuous entity because it contains a gap in its mention. In these cases, we remove the overlapping part of the entities. In the previous example, we discard the words *adrenergic blocking drugs* and keep the *ganglionic* and *peripheral* as the interacting drugs.

*3.2.1. Architecture of our deep network for RE*

Below, we describe each of the layers in our CNN model relation extraction task. Firstly, a lookup operation transforms each word in the input sentence into a real value vector according to its embeddings. Then a Convolutional layer followed by a max-pooling operation represents into a vector the relationship of the instance. Finally, a classification layer creates a prediction for a predefined class.

*Word table layer*

After the pre-processing phase, we create an input matrix suitable for the CNN architecture, which represents all the training instances. We need that all sentences have the same length. We determined the length $n$ as the longest sentence, and the sentences are extended with an auxiliary token "*0*" if their lengths are shorter than this threshold .

Moreover, each word has to be represented by a vector. To do this, we randomly initialized a vector for each different word. Thus, each word is replaced 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 relation instance where each word of the sentence is represented by its corresponding word vector from the word embedding matrix. Furthermore, we extract the types of the two interacting entities from the Name Entity Classification task and converting them to a real value vector with a type embedding matrix $\mathbf{W}_{Type} \in \mathbb{R}^{|T| \times m_t}$.

We denote $p_1$ and $p_2$ as the positions in the sentence of the two target entities that make up the relation instance. The following step involves calculating the relative position of each word to the two target 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 [55]. 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, the type embedding and the two position embeddings for each word in the instance.

*Convolutional layer*

Once we obtain 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 4, 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.

*Pooling layer*

In this layer, the goal is to extract the most relevant features of each filter using an aggregating function. We use 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.

*Softmax layer*

Before performing the classification, we perform a dropout to prevent overfitting. We obtain 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 feed 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 and $k$ is the number of classes in the dataset. At test time, the vector $\mathbf{z}$ of a new instance is directly classified by the Softmax layer without a dropout.

*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 $\mathbf{f}$. 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 [56] to learn the parameters.

### *3.3. Datasets*

In this section, we describe the two biomedical datasets with different languages, English and Spanish, used for the experiments.

*3.3.1. The DDI corpus*

The DDI corpus [57] is a valuable annotated corpus which provides gold standard data for training and evaluating machine-learning algorithms to extract pharmacological substances and DDIs from texts. This corpus measured the performance of the systems in the two editions of DDIExtraction [3, 4]. The DDI corpus contains 233 selected abstracts about DDIs from MedLine as well as 792 documents describing DDIs from the DrugBank database [58]. The corpus contains four types of pharmacological substances (drug, group, brand, and non-human drug) and four different types of drug interactions (mechanism, effect, advice and int). [57] describes these types with more detail. The corpus was annotated manually with a total of 18,502 pharmacological substances and 5028 DDIs. Tables 3 and 4 show some statistics of the DDI corpus.

| Entity type | Training | Test |
|---|---|---|
| *Drug* | 11,646 | 351 |
| *Brand* | 1,866 | 59 |
| *Group* | 4,225 | 155 |
| *Non-human drug (Drug-n)* | 765 | 121 |
| Total | 18,502 | 686 |

Table 3: Entity types in the DDI corpus.

| DDI types | Training | Test |
|---|---|---|
| *Mechanism* | 1,319 | 302 |
| *Effect* | 1,687 | 360 |
| *Advice* | 826 | 221 |
| *Int* | 188 | 96 |
| Total | 4,020 | 979 |

Table 4: Relation (DDI) types in the DDI corpus.

*3.3.2. The eHealth-KD challenge dataset*

The eHealth-KD challenge [9], which is part of the Workshop on Semantic Analysis at SEPLN (TASS-2018), provided to participating teams an

annotated collection of MedlinePlus documents. MedlinePlus [6] is an informative website directed to patients, which offers information about health topics such as medicines and diseases. The documents were annotated with keyphrases (entities) and semantic relations. Two different entity types are proposed to classify the keyphrases: *Concept* and *Action*. Likewise, six types of relationships are defined: *is-a*, *part-of*, *property-of* and *same-as*, which are relationships between concepts, and *subject* and *target*, which can represent relationships between actions and concepts or between actions themselves. Tables 5 and 6 show the statistics for the entity and relation types in the eHealth-KD dataset, respectively.

The dataset was split into three subsets: training set (559 sentences), validation set (285 sentences) and test set (300 sentences). At the same time, the test set has three different subsets for measuring the performance in each scenario. A detailed description of this dataset can be found in [59].

| Entity type | Train | Validation | Scenario 1 Test | Scenario 2 Test | Scenario 3 Test |
|---|---|---|---|---|---|
| *Concept* | 2,427 | 849 | 432 | 439 | 434 |
| *Action* | 1,525 | 434 | 163 | 154 | 183 |
| Total | 3,952 | 1,283 | 595 | 593 | 617 |

Table 5: Entity types in the eHealth-KD challenge datasets.

| Relation type | Train | Validation | Scenario 1 Test | Scenario 2 Test | Scenario 3 Test |
|---|---|---|---|---|---|
| *is-a* | 434 | 370 | 74 | 92 | 69 |
| *part-of* | 149 | 145 | 31 | 33 | 32 |
| *property-of* | 399 | 244 | 58 | 58 | 62 |
| *same-as* | 30 | 13 | 2 | 1 | 5 |
| *subject* | 693 | 339 | 147 | 117 | 137 |
| *target* | 991 | 504 | 180 | 195 | 212 |
| Total | 2,696 | 1,615 | 492 | 496 | 517 |

Table 6: Relation types in the eHealth-KD challenge datasets.

[6] https://medlineplus.gov/spanish/

## 4. Results and Discussion

In this section, we present the experiment results provided by each module of our information extraction system independently. Finally, we discuss the results provided by the full pipeline, which takes each module takes as input the output of the previous subsystems. We use the standard evaluation metrics of entity and relation extraction: precision (P), recall (R) and F1.

### *4.1. NER results*

We evaluate our NER model in two different tasks:

i) the detection and classification of pharmacological substances in the DDI corpus,

ii) the detection and classification of keyphrases in the eHealth-KD challenge dataset.

The parameters of the sets and the hyper parameters for our Bi-LSTM CRF model are summarized in Table 7.

| Parameter | DDI | eHealth-KD |
|---|---|---|
| Sense-disambiguation embedding dimension | 100 | 128 |
| Word embeddings dimension | 100 | 300 |
| Character embedding dimension | 50 | 50 |
| Hidden layers dimension (for each LSTM) | 100 | 100 |
| Learning method | SGD | SGD |
| Dropout rate | 0.5 | 0.5 |
| Learning rate | 0.005 | 0.005 |
| Epochs | 100 | 100 |

Table 7: Parameters for Bi-LSTM CRF model.

Table 9 shows the results of our model for the entity detection task. The results for the classification task are shown in Tables 11 and 10.

| **Dataset** | **P** | **R** | **F1** |
|---|---|---|---|
| DDI | 87.24% | 87.15% | 87.19% |
| eHealth-KD | 86.2% | 88.2% | 87.2% |

Table 8: Results for the entity detection task.

Comparing to previous works in drug name detection [4] on the DDI corpus, our system also achieves, only 0.6% worse than the top system [65],

| Dataset | P | R | F1 |
|---|---|---|---|
| **Our system** | 78.80% | **81.80%** | **80.30%** |
| Zeng [18] | 83.60% | 77.80% | 79.20% |
| LIU [60] | **84.70%** | 72.8% | 78.3% |
| WBI [61] | 73.40% | 69.80% | 71.50% |
| LASIGE [62] | 69.60% | 62.10% | 65.60% |
| UTurku [63] | 73.70% | 57.90% | 64.80% |
| UC3M [64] | 51.70% | 54.20% | 52.90% |

Table 9: Comparison of our NER system with other systems for DrugNER task.

which describes a rich linguistic and semantic feature set to train a CRF classifier. An essential advantage of our approach over this previous work is that our system does not exploit any domain-specific features, and thereby, it could be easily adapted to any named entity category.

Besides, our approach achieves an F1 of 87.2% for the entity detection task on the eHealth-KD dataset, which was the best result for entity detection in the eHealth-KD challenge.

Moreover, the NER module provides better results on the eHealth-KD dataset than on the DDI corpus. A possible reason may be that the DDI corpus contains several discontinuous named entities which are hard to disambiguate with the V tag, while entities in the eHealth-KD dataset do not contain any gap in their mentions.

Regarding the results for the entity classification task on the DDI corpus, the *Group* type obtains the best performance, followed by the *Drug* and *Brand* types. Our F1 for the group type (86.06%) is almost 9% higher than that of the top system in DDIExtraction 2013 [4, 65]. Groups of drugs are usually named by multi-word expressions, such as "Nondepolarizing Neuromuscular Blocker" or "HMG CoA Reductase Inhibitor". Therefore, our deep neural network based on character, word and sense embeddings seems to obtain better results in the classification of multi-word expressions than the rich linguistic and semantic feature set used by the CRF model in [65].

Our results are similar to those obtained for the *Drug* type. However, a previous work [63], which exploits information from terminological resources such as DrugBank or MetaMap obtains better results for the *Brand* type (representing trademarked drugs) than our approach. As in previous works, our approach obtains a deficient performance for non-human drug entities.

This is caused by the small number of instances in the training dataset.

Our system obtained the best performance for the entity classification task in the eHealth-KD challenge, with an accuracy of 95.9%. Table 11 shows the results for each entity type. We can observe that the *Concept* class is easier to classify than the *Action* class because there is an unbalance of annotations between them. Concretely, the instances of the *Action* type being almost half of the *Concept* type. On the other hand, there is ambiguity in Action-type entities that can sometimes represent concepts. For example in the sentence 'Los empleados dedicados al cuidado de la salud están expuestos a muchos riesgos laborales.' ('Employees dedicated to health care are exposed to many occupational hazards.') the entity 'cuidado' is of the Action type but in the sentence 'Practicar deportes puede ser divertido, pero si no se tiene cuidado también puede ser peligroso.' ('Playing sports can be fun, but if you're not careful, it can also be dangerous.') the entity 'cuidado' is of the Concept type. Tables 12 and 13 compares our results with the rest of participating systems for subtasks A and B for the eHealth-KD challenge.

| **Drug type** | **P** | **R** | **F1** |
|---|---|---|---|
| *Drug* | 80.73% | 89.22% | 84.76% |
| *Brand* | 75.00% | 90.00% | 81.82% |
| *Group* | 84.06% | 94.31% | 88.89% |
| *Non-human drug* | 58.59% | 35.21% | 43.99% |
| Overall | 76.11% | 78.10% | 76.21% |

Table 10: Entity classification results on the DDI corpus.

| **Entity Type** | **P** | **R** | **F1** |
|---|---|---|---|
| *Concept* | 85.24% | 86.77% | 86.00% |
| *Action* | 80.00% | 83.22% | 81.58% |
| Overall | 84% | 86% | 85% |

Table 11: Results for the entity classification on the eHealth-KD dataset.

### *4.2. RE results*

Table 14 summarizes the parameters for our CNN model where the hyperparameter of the network $M_e$, $M_d$, $M_t$, $w$ and $m$ were fine-tuning with a grid search on each dataset.

| **Entity Type** | **P** | **R** | **F1** |
|---|---|---|---|
| **Our system** | **86.20%** | **88.20%** | **87.20%** |
| SINAI [66] | 77.00% | 81.00% | 79.00% |
| UPF_UPC [67] | 86.00% | 75.00% | 80.00% |
| LABDA [35] | 31.00% | 32.00% | 32.00% |

Table 12: Results of the participaiting systems in the eHealth-KD challenge subtask A (Entity detection).

| **System** | **Accuracy** | **Correct** | **Incorrect** |
|---|---|---|---|
| **Our system** | **95.90%** | **496** | **21** |
| SINAI [66] | 92.10% | 546 | 47 |
| UPF_UPC [67] | 95.40% | 418 | 20 |
| TALP [34] | 93.10% | 552 | 41 |
| LABDA [35] | 59.40% | 218 | 149 |

Table 13: Results of the participaiting systems in the eHealth-KD challenge subtask B (Entity classification).

| Parameter | DDI | eHealth-KD |
|---|---|---|
| Maximal length in the dataset, $n$ | 128 | 40 |
| Word embeddings dimension, $M_e$ | 300 | 300 |
| Position embeddings dimension, $M_d$ | 5 | 10 |
| Type embeddings dimension, $M_t$ | 10 | 10 |
| Filter window sizes, $w$ | 2, 4, 6 | 3, 4, 5 |
| Filters for each window size, $m$ | 200 | 200 |
| Dropout rate, $p$ | 0.5 | 0.5 |
| Non-linear function, $g$ | ReLU | ReLU |
| $l2$-regularization | 3 | 0.1 |
| Mini-batch size | 50 | 50 |
| Learning rate | 0.001 | 0.001 |

Table 14: Parameters for CNN model.

In the eHealth-KD dataset, some sentences describe relationships between nested entities, that is, between an entity and its overlapped entity. We are not able to blind all possible entity mentions forming a nested entity. For this reason, we do not consider these relation instances. Moreover, there are relationships with more than one type. In this case, we only consider the

first type because our system can not cope with a multi-class problem.

As can be seen in Tables 15 and 16, the relation extraction task seems to be a more complex task than the previous tasks. These tables show the results on the Scenario 3 or RE subtask, that is when the input for our system are the annotated entities and their corresponding types.

| **DDI type** | **P** | **R** | **F1** |
|---|---|---|---|
| *mechanism* | 74.23% | 63.91% | 68.68% |
| *effect* | 65.57% | 66.67% | 66.12% |
| *advise* | 75.12% | 68.33% | 71.56% |
| *int* | 86.11% | 32.29% | 46.97% |
| Overall | 71.26% | 62.82% | 66.78% |

Table 15: Relation classification results on the DDI corpus.

| **Relation type** | **P** | **R** | **F1** |
|---|---|---|---|
| *is-a* | 44% | 15.94% | 23.4% |
| *part-of* | 37.5% | 9.38% | 15% |
| *property-of* | 57.45% | 43.55% | 49.54% |
| *same-as* | 50% | 20% | 28.57% |
| *subject* | 57.69% | 43.8% | 49.79% |
| *target* | 67.58% | 69.81% | 68.68% |
| Overall | 61.73% | 48.36% | 54.23% |

Table 16: Results for the relation classification on the eHealth-KD dataset.

Tables 17 and 18 compare the results of our approach with other previous systems evaluated on the DDI corpus and the eHealth-KD dataset, respectively. As is shown in Table 17, our approach provides modest results compared with other previous systems for DDI extraction. This may be due to our architecture is simpler than those used in those works. On the other hand, our method ranks first in the eHealth-KD Challenge overcoming the winner system in the relation Extraction task.

Focusing on the DDI corpus, our RE module obtains the best performance for the *advice* type. The entities in this category are typically described by very similar patterns such as "DRUG should not be used in combination with DRUG" or "Caution should be observed when DRUG is administered with DRUG", which can be quickly learned by the model because they are

| Systems | P | R | F1 |
|---|---|---|---|
| ***Our*** | 71.26% | 62.82% | 66.78% |
| *Multichannel + 10 CNN layers [30]* | **86.18%** | **87.2%** | **86.27%** |
| *Joint AB-LSTM [29]* | 73.41% | 69.66% | 71.48% |
| *CNN+DCNN [27]* | 78.24% | 64.66% | 70.81% |
| *MCCNN [26]* | 75.99% | 65.25% | 70.21% |
| *CNN with MEDLINE word embedding [25]* | 75.72% | 64.66% | 69.75% |
| *SCNN [28]* | 72.5% | 65.1% | 68.6% |
| *FBK-irst [24]* | 64.6% | 65.6% | 65.1% |

Table 17: Results of the relation classification systems on the DDI corpus.

| Systems | P | R | F1 |
|---|---|---|---|
| ***Our system*** | **61.73%** | **48.36%** | **54.23%** |
| *TALP [34]* | - | - | 44.8% |
| *LaBDA [35]* | 58.12% | 35.98% | 44.44% |

Table 18: Results of the relation classification systems on the eHealth-KD dataset.

prevalent in the DDI corpus. The *mechanism* type is the second one with the best performance (68.68%), even though its number of instances is lower than the effect type (see Table 4). Finally, the *int* type is the most challenging type to classify because the training instances for this type are much more scarce (5.6%) than those of the remainder of the types (41.1% for *effect*, 32.3% for *mechanism* and 20.9% for *advice*).

Regarding the results on the eHealth-KD dataset, the system overcomes the top system in the RE subtask improving the results from 44.8% to 54.23%. Besides, these results are directly related to the number of training instances for each relation types. In this way, the system obtains the best F1 for the target relation, followed by the subject type which is the most representative class in this dataset.

### *4.3. Two-stage pipeline results*

We follow the same evaluation metrics defined in eHealth-KD Task in order to obtain a comparative and measure each scenario independently for the two-stage performance. The results for the Scenario 1 are calculated with

the aggregated metrics of the subtask A, B and C as follows:

$$P = \frac{correct(A) + \frac{1}{2}partial(A) + correct(B) + correct(C)}{correct(A) + partial(A) + spurious(A) + correct(B) + incorrect(B) + correct(C) + spurious(C)} \quad (1)$$

$$R = \frac{correct(A) + \frac{1}{2}partial(A) + correct(B) + correct(C)}{correct(A) + partial(A) + missing(A) + correct(B) + incorrect(B) + correct(C) + missing(C)} \quad (2)$$

where *correct* are the labels that matched to the test set and the prediction (true positives), *missing* are the labels that are in the test set but not in the prediction (false negatives), *spurious* are the labels that are in the prediction but not in the test set (false positives), *partial* are the detected entities whose boundaries do not exactly match and *incorrect* are the entities wrongly classified. In order to calculate the same metrics for the Scenarios 2 and 3, the instances for the previous tasks are canceled in the equations. The F-measure or F1 is calculated from the precision and recall of each corresponding scenario. Furthermore, the average of all three scenarios gives the final score. Differently to eHealth-KD which has a test set for each scenario, in DDI corpus we consider the same test for all the scenarios.

Table 19 shows the results for each scenario using the eHealth-KD dataset. The final score for all the scenarios is 67.62%, which is 21.2 points in a percentage higher than the top system in this task. The main reason for this improvement is that the winner system did not use a RE system and has lower statistics for Task C in the scenarios. We obtain a state-of-the-art technique which shows a high precision in the different scenarios. Furthermore, we outperform all the previous F1 measure in the eHealth-KD challenge.

Table 20 presents all scenarios taking the DDI corpus test set given an average of 61.92% F1. Despite the Scenario 3 in DDI corpus obtains better results than on the eHealth-KD dataset, the final score is lower because it is directly affected by the performance of the NER module which affects the remaining scenarios. We can see that the number of spurious and incorrect are very high for task A and B which caused a low Precision in Scenarios 1 and 2 and we will take into consideration for future work.

## 5. Conclusions

This paper presents a two-stage IE system from medical texts. Our system deals with three different tasks: A) entity detection, B) entity classification and C) relation extraction. The system is composed of two different

| | Scenario 1 | Scenario 2 | Scenario 3 |
|---|---|---|---|
| Correct A | 505 | - | - |
| Partial A | 40 | - | - |
| Missing A | 50 | - | - |
| Spurious A | 64 | - | - |
| Correct B | 511 | 553 | - |
| Incorrect B | 34 | 40 | - |
| Correct C | 168 | 205 | 250 |
| Missing C | 324 | 291 | 267 |
| Spurious C | 240 | 183 | 155 |
| Recall | 73.78% | 69.61% | 48.36% |
| Precision | 77.08% | 77.27% | 61.73% |
| F-measure | 75.39% | 73.24% | 54.23% |

Table 19: eHealth-KD dataset results for the different scenarios.

| | Scenario 1 | Scenario 2 | Scenario 3 |
|---|---|---|---|
| Correct A | 2236 | - | - |
| Partial A | 67 | - | - |
| Missing A | 149 | - | - |
| Spurious A | 1488 | - | - |
| Correct B | 2149 | 2149 | - |
| Incorrect B | 1503 | 1503 | - |
| Correct C | 559 | 559 | 615 |
| Missing C | 253 | 253 | 364 |
| Spurious C | 951 | 951 | 248 |
| Recall | 71.97% | 60.66% | 62.82% |
| Precision | 55.6% | 52.46% | 71.26% |
| F-measure | 62.73% | 56.26% | 66.78% |

Table 20: DDI corpus results for the different scenarios.

modules: one for detecting and classifying entities, and a second one for extracting relationships between them. Both modules are based on Deep Learning architectures. The NER module is based on a Bi-LSTM network, exploiting character, word and sense embeddings models as input, and a final CRF-layer. The RE module is a CNN model using the word, entity type and position embeddings as input, and a Softmax classifier in its last layer. We

perform a detailed experimentation on two different datasets:

i) the DDI corpus composed of scientific texts in English and annotated with pharmacological substances as well as their possible drug interactions,

ii) the eHealth-KD dataset, a collection of articles about health for Spanish speaking patients, annotated with concepts, actions and general semantic relations such as part-of, property-of, among others (see Table 6).

Firstly, we evaluate the performance of the system for each module separately. Thus, the NER module does not require any annotation, while the RE module takes as input the texts annotated with entities. Finally, the whole system is tested taking as input only plain text to assess the two-stage system in a real scenario.

Our final goal is to exploit this two-stage IE system in different clinical applications such as the automatic cohort identification for epidemiological and clinical studies and the summarization of clinical records, which are the main objectives defined in the research project DeepEMR, supported by Government of Spain. We have already applied successfully deep learning methods for the classification of clinical records, like a CNN model that identifies anaphylaxis cases (severe allergic reactions) described in clinical records [68]. This identification can significantly facilitate the conduct of epidemiological and clinical studies in the allergy field, as well as the reduction of their costs. As next steps, the extraction of concepts and their semantic relations will also provide valuable information for the performance of these clinical studies. Moreover, knowing the fundamental concepts as well as their relationships in a text is a crucial step in order to create a comprehensive summary of the clinical records for a given patient.

The performance of the presented system achieves the state-of-the-art results on the eHealth-KD dataset for all subtasks (NER and RE). Besides, our two-stage pipeline is the state-of-art system for the eHealth-KD challenge. Focusing on the DDI corpus, the NER module outperforms previous state-of-art results for drug name recognition [65]. However, our RE module is far from the state-of-art system for DDI extraction because it is a basic CNN compared to the ten-layer CNN system of [30].

Our work achieves the state-of-art results for the eHealth-KD challenge and, to the best of our knowledge, is the first attempt to evaluate a two-stage IE system for extracting DDI from texts. Another significant contribution of our work is that our approach is a task-independent method because it can deal with different entity and relation types. Moreover, the same approach can be applied to different languages (such as Spanish and English), only

needing different pre-trained word embeddings for each language.

Our results of the full IE system shows that there is much room for improvement. As future work, we plan to study different combinations of deep learning architectures for NER and RE modules. In addition, we want to explore deeper layers systems that seem to improve the results for both tasks in DDI corpus. We propose to try different word embeddings pre-trained in the clinical domain that takes the semantic knowledge of the sentences in order to improve the results in these subtasks. Furthermore, we plan to add more syntactic information of the sentence, such as Part-of-Speech tags, Chunk labels, dependency types, through the embeddings. Particularly for the RE task, we propose to augment the class instances with distant supervision techniques or Generative Adversarial Neural Networks to deal with the imbalanced datasets in IE tasks.

## 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).

[1] H. Dalianis, Applications of Clinical Text Mining, Springer International Publishing, Cham, 2018, pp. 109–148. URL: https://doi.org/10.1007/978-3-319-78503-5_10. doi:10.1007/978-3-319-78503-5_10.

[2] L. Hirschman, A. Yeh, C. Blaschke, A. Valencia, Overview of biocreative: critical assessment of information extraction for biology, BMC Bioinformatics 6 (2005) S1. doi:10.1186/1471-2105-6-S1-S1.

[3] I. Segura-Bedmar, P. Martınez, D. Sánchez-Cisneros, The 1st ddiextraction-2011 challenge task: Extraction of drug-drug interactions from biomedical texts, in: Proceedings of DDIExtraction-2011 challenge task, 2011.

[4] I. Segura Bedmar, P. Martínez, M. Herrero Zazo, Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013), in: Proceedings of the 7th International Workshop on. Semantic Evaluation (SemEval 2017), Association for Computational Linguistics, 2013, pp. 341–350.

[5] I. Segura-Bedmar, P. Martínez, M. Herrero-Zazo, Lessons learnt from the ddiextraction-2013 shared task, Journal of biomedical informatics 51 (2014) 152–164.

[6] I. Segura-Bedmar, P. Martínez, Pharmacovigilance through the development of text mining and natural language processing techniques, Journal of biomedical informatics 58 (2015) 288–291.

[7] K. Roberts, D. Demner-Fushman, J. M. Tonning, Overview of the TAC 2017 adverse reaction extraction from drug labels track, in: Proceedings of the 2017 Text Analysis Conference, TAC 2017, Gaithersburg, Maryland, USA, November 13-14, 2017, 2017.

[8] A. Sarker, G. Gonzalez (Eds.), Proceedings of the 2nd Social Media Mining for Health Research and Applications Workshop co-located with the American Medical Informatics Association Annual Symposium (AMIA 2017), Washington D.C., United States, November 4, 2017, volume 1996 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017.

[9] E. Martínez-Cámara, 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, J. Villena-Román, Overview of TASS 2018: Opinions, health and emotions, in: 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, J. Villena-Román (Eds.), Proceedings of TASS 2018: Workshop on Semantic Analysis at SEPLN (TASS 2018), volume 2172 of *CEUR Workshop Proceedings*, CEUR-WS, Sevilla, Spain, 2018.

[10] M. Herrero-Zazo, I. Segura-Bedmar, P. Martínez, T. Declerck, The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions, Journal of Biomedical Informatics 46 (2013) 914–920.

[11] B. Settles, ABNER: An open source tool for automatically tagging genes, proteins and other entity names in text, Bioinformatics 21 (2005) 3191–3192. URL: `http://www.cs.wisc.edu/~bsettles/abner/`. doi:`10.1093/bioinformatics/bti475`.

[12] R. LEAMAN, G. GONZALEZ, Banner: an Executable Survey of Advances in Biomedical Named Entity Recognition, Biocomputing 2008 (2007) 652–663. URL: http://www.worldscientific.com/doi/abs/10.1142/9789812776136{_}0062. doi:10.1142/9789812776136_0062.

[13] T. Rocktäschel, M. Weidlich, U. Leser, Chemspot: A hybrid system for chemical named entity recognition, Bioinformatics 28 (2012) 1633–1640. URL: https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/bts183. doi:10.1093/bioinformatics/bts183.

[14] D. Campos, S. Matos, J. L. Oliveira, Gimli: open source and high-performance biomedical name recognition, BMC Bioinformatics 14 (2013). URL: http://www.biomedcentral.com/1471-2105/14/54http://bioinformatics.ua.pt/gimli.

[15] J. P. C. Chiu, E. Nichols, Named Entity Recognition with Bidirectional LSTM-CNNs (2015). URL: http://arxiv.org/abs/1511.08308. doi:10.3115/1119176.1119204. arXiv:1511.08308.

[16] N. Limsopatham, N. Collier, Learning Orthographic Features in Bi-directional LSTM for Biomedical Named Entity Recognition (????). URL: http://www.nactem.ac.uk/biotxtm2016/papers/Limsopatham.pdf.

[17] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, C. Dyer, Neural Architectures for Named Entity Recognition (2016). URL: http://arxiv.org/abs/1603.01360. doi:10.18653/v1/N16-1030. arXiv:1603.01360.

[18] D. Zeng, C. Sun, L. Lin, B. Liu, Enlarging drug dictionary with semi-supervised learning for Drug Entity Recognition, Proceedings - 2016 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2016 (2017) 1929–1931. doi:10.1109/BIBM.2016.7822818.

[19] M. Gridach, Character-level neural network for biomedical named entity recognition, Journal of Biomedical Informatics 70 (2017) 85–91. URL: http://dx.doi.org/10.1016/j.jbi.2017.05.002. doi:10.1016/j.jbi.2017.05.002.

[20] X. Ma, E. Hovy, End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF (????) 1064–1074. URL: http://www.aclweb.org/anthology/P16-1101.

[21] C. Lyu, B. Chen, Y. Ren, D. Ji, Long short-term memory rnn for biomedical named entity recognition, BMC bioinformatics 18 (2017) 462.

[22] L. Luo, Z. Yang, P. Yang, Y. Zhang, L. Wang, H. Lin, J. Wang, An attention-based bilstm-crf approach to document-level chemical named entity recognition, Bioinformatics 34 (2017) 1381–1388.

[23] W. Ling, T. Luís, L. Marujo, R. F. Astudillo, S. Amir, C. Dyer, A. W. Black, I. Trancoso, Finding function in form: Compositional character models for open vocabulary word representation, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, p. 1520–1530.

[24] M. F. M. Chowdhury, A. Lavelli, Fbk-irst: A multi-phase kernel based approach for drug-drug interaction detection and classification that exploits linguistic information, 7th International Workshop on Semantic Evaluation (SemEval 2013) 351 (2013) 53.

[25] S. Liu, B. Tang, Q. Chen, X. Wang, Drug-drug interaction extraction via convolutional neural networks, Computational and Mathematical Methods in Medicine Vol. 2016 (2016) 8 pages.

[26] C. Quan, L. Hua, X. Sun, W. Bai, Multichannel convolutional neural network for biological relation extraction, BioMed research international 2016 (2016).

[27] S. Liu, K. Chen, Q. Chen, B. Tang, Dependency-based convolutional neural network for drug-drug interaction extraction, in: Bioinformatics and Biomedicine (BIBM), 2016 IEEE International Conference on, IEEE, 2016, pp. 1074–1080.

[28] Z. Zhao, Z. Yang, L. Luo, H. Lin, J. Wang, Drug drug interaction extraction from biomedical literature using syntax convolutional neural network, Bioinformatics (2016).

[29] S. K. Sahu, A. Anand, Drug-drug interaction extraction from biomedical text using long short term memory network, CoRR abs/1701.08303 (2017). URL: http://arxiv.org/abs/1701.08303.

[30] I. N. Dewi, S. Dong, J. Hu, Drug-drug interaction relation extraction with deep convolutional neural networks, in: 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2017, pp. 1795–1802. doi:10.1109/BIBM.2017.8217933.

[31] I. Augenstein, M. Das, S. Riedel, L. Vikraman, A. McCallum, Semeval 2017 task 10: Scienceie - extracting keyphrases and relations from scientific publications, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 546–555. URL: http://www.aclweb.org/anthology/S17-2091.

[32] J. Y. Lee, F. Dernoncourt, P. Szolovits, MIT at semeval-2017 task 10: Relation extraction with convolutional neural networks, CoRR abs/1704.01523 (2017). URL: http://arxiv.org/abs/1704.01523. arXiv:1704.01523.

[33] J. Rotsztejn, N. Hollenstein, C. Zhang, Eth-ds3lab at semeval-2018 task 7: Effectively combining recurrent and convolutional neural networks for relation classification and extraction, CoRR abs/1804.02042 (2018). URL: http://arxiv.org/abs/1804.02042. arXiv:1804.02042.

[34] S. Medina, J. Turmo, Joint classification of key-phrases and relations in electronic health documents, in: Proceedings of TASS 2018: Workshop on Sentiment Analysis at SEPLN co-located with 34th SEPLN Conference (SEPLN 2018), Sevilla, Spain, September 18th, 2018., 2018, pp. 83–88. URL: http://ceur-ws.org/Vol-2172/p9-talp_tass2018.pdf.

[35] S. Medina, J. Turmo, Labda at tass-2018 task 3: Convolutional neural networks for relation classification in spanish ehealth documents, in: Proceedings of TASS 2018: Workshop on Sentiment Analysis at SEPLN co-located with 34th SEPLN Conference (SEPLN 2018), Sevilla, Spain, September 18th, 2018., 2018, pp. 71–76. URL: http://ceur-ws.org/Vol-2172/p7-labda_tass2018.pdf.

[36] Z. Zhao, Z. Yang, C. Sun, L. Wang, H. Lin, A hybrid protein-protein interaction triple extraction method for biomedical literature, in: Bioinformatics and Biomedicine (BIBM), 2017 IEEE International Conference on, IEEE, 2017, pp. 1515–1521.

[37] R. Bunescu, R. Ge, R. J. Kate, E. M. Marcotte, R. J. Mooney, A. K. Ramani, Y. W. Wong, Comparative experiments on learning information extractors for proteins and their interactions, Artificial intelligence in medicine 33 (2005) 139–155.

[38] J. Björne, T. Salakoski, Tees 2.2: biomedical event extraction for diverse corpora, BMC bioinformatics 16 (2015) S4.

[39] W. Ammar, M. Peters, C. Bhagavatula, R. Power, The ai2 system at semeval-2017 task 10 (scienceie): semi-supervised end-to-end entity and relation extraction, in: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Association for Computational Linguistics, 2017, pp. 592–596. URL: http://www.aclweb.org/anthology/S17-2097. doi:10.18653/v1/S17-2097.

[40] M. Miwa, M. Bansal, End-to-end relation extraction using lstms on sequences and tree structures, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, 2016, pp. 1105–1116. doi:10.18653/v1/P16-1105.

[41] R. M. Rivera Zavala, P. Martínez, I. Segura-Bedmar, A Hybrid Bi-LSTM-CRF model for Knowledge Recognition from eHealth documents, Technical Report, 2018. URL: http://ceur-ws.org/Vol-2172/p6_hybrid_bi_lstm_tass2018.pdf.

[42] Explosion AI, spaCy - Industrial-strength Natural Language Processing in Python, ???? URL: https://spacy.io/.

[43] A. Borthwick, J. Sterling, E. Agichtein, R. Grishman, Exploiting diverse knowledge sources via maximum entropy in named entity recognition, in: Sixth Workshop on Very Large Corpora, 1998. URL: http://www.aclweb.org/anthology/W98-1118.

[44] S. Hochreiter, J. Schmidhuber, Long short-term memory., Neural computation 9 (1997) 1735–80. URL: http://www.ncbi.nlm.nih.gov/pubmed/9377276.

[45] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM networks, in: Proceedings of the International Joint Conference on Neural Networks, volume 4, IEEE, 2005, pp. 2047–2052. URL: http://ieeexplore.ieee.org/document/1556215/. doi:10.1109/IJCNN.2005.1556215.

[46] C. Dyer, M. Ballesteros, W. Ling, A. Matthews, N. A. Smith, Transition-Based Dependency Parsing with Stack Long Short-Term Memory, Technical Report, 2015. URL: http://arxiv.org/abs/1505.08075. doi:10.3115/v1/P15-1033. arXiv:1505.08075.

[47] Y. Bengio, P. Simard, P. Frasconi, Learning Long-Term Dependencies with Gradient Descent is Difficult, IEEE Transactions on Neural Networks 5 (1994) 157–166. URL: http://ieeexplore.ieee.org/document/279181/. doi:10.1109/72.279181. arXiv:arXiv:1211.5063v2.

[48] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training Recurrent Neural Networks (2012). URL: http://arxiv.org/abs/1211.5063. doi:10.1109/72.279181. arXiv:1211.5063.

[49] C. Cardellino, Spanish Billion Words Corpus and Embeddings, 2016. URL: http://crscardellino.me/SBWCE/.

[50] M. Taulé, M. A. Martí, M. Recasens, Ancora: Multilevel annotated corpora for catalan and spanish., in: LREC 2008, 2008, pp. 96–101.

[51] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543.

[52] A. Trask, P. Michalak, J. Liu, sense2vec - A Fast and Accurate Method for Word Sense Disambiguation In Neural Word Embeddings (2015). URL: http://arxiv.org/abs/1511.06388. arXiv:1511.06388.

[53] J. Lafferty, A. McCallum, F. C. N. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning 8 (2001) 282–289. URL: http://repository.upenn.edu/cis{_}papers/159/{%}5Cnhttp://dl.acm.org/citation.cfm?id=655813. doi:10.1038/nprot.2006.61. arXiv:arXiv:1011.4088v1.

[54] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1746–1751.

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

[56] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, CoRR abs/1412.6980 (2014).

[57] M. Herrero-Zazo, I. Segura-Bedmar, P. Martínez, T. Declerck, The DDIcorpus: An annotated corpus with pharmacological substances and drug-drug interactions, Journal of Biomedical Informatics 46 (2013) 914 – 920.

[58] D. S. Wishart, C. Knox, A. C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang, J. Woolsey, Drugbank: a comprehensive resource for in silico drug discovery and exploration, Nucleic acids research 34 (2006) D668–D672.

[59] A. Piad-Morffis, Y. Gutiérrez, R. Muñoz, A corpus to support ehealth knowledge discovery technologies, Journal of biomedical informatics 94 (2019) 103172.

[60] S. Liu, B. Tang, Q. Chen, X. Wang, Drug name recognition: Approaches and resources, Information (Switzerland) 6 (2015) 790–810. doi:10.3390/info6040790.

[61] I. Segura-Bedmar, P. Martinez, M. Herrero-Zazo, Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013), Technical Report SemEval, 2013. URL: http://www.aclweb.org/anthology/S13-2056{%}3E. doi:10.1.1.310.783.

[62] T. Grego, F. M. Couto, LASIGE : using Conditional Random Fields and ChEBI ontology, Technical Report SemEval, 2013. URL: http://aclweb.org/anthology/S13-2109.

[63] J. Björne, S. Kaewphan, T. Salakoski, UTurku : Drug Named Entity Recognition and Drug-Drug Interaction Extraction Using SVM Classification and Domain Knowledge, Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013) 2 (2013) 651–659. URL: http://www.aclweb.org/anthology/S13-2108.

[64] D. Sanchez-Cisneros, P. Martínez, I. Segura-Bedmar, Combining dictionaries and ontologies for drug name recognition in biomedical texts, in: Proceedings of the 7th international workshop on Data and text mining in biomedical informatics - DTMBIO '13, ACM Press, New York, New York, USA, 2013, pp. 27–30. URL: http://dl.acm.org/citation.cfm?doid=2512089.2512100. doi:10.1145/2512089.2512100.

[65] T. Rocktäschel, T. Huber, M. Weidlich, U. Leser, Wbi-ner: The impact of domain-specific features on the performance of identifying and classifying mentions of drugs, in: Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), volume 2, 2013, pp. 356–363.

[66] P. López, M. C. Díaz-Galiano, M. Teresa Martín-Valdivia, L. Alfonso Ureña-López, Clasificando acciones y conceptos con UMLS en MedLine, Technical Report, 2018. URL: https://medlineplus.gov/.

[67] J. V. Palatresi, H. R. Hontoria, TASS2018: Medical knowledge discovery by combining terminology extraction techniques with machine learning classification, Technical Report, 2018. URL: https://medlineplus.gov/xml.html.

[68] C.-R. C. Segura-Bedmar, Isabel, M. A. Tejedor, M. Moro-Moro, Predicting of anaphylaxis in big data EMR by exploring machine learning approaches, Journal of Biomedical Informatics Accepted (2018).