Intelligent Environments 2020
C.A. Iglesias et al. (Eds.)

doi:10.3233/AISE200061

Enhancing the spaCy Named Entity Recognizer for Crowdsensing

Julio FERNÁNDEZ-PEDAUYE [a,1], Carlos PERIÑÁN-PASCUAL [b]
Francisco ARCAS-TÚNEZ [a] and José M. CECILIA [b]
[a] *Universidad Católica de Murcia (UCAM)*
[b] *Universitat Politècnica de València*

Abstract. Social sensing leverages user-contributed data from social media by considering participants as "social sensors", i.e. agents that provide information about their environment through social-media services such as Twitter, Facebook or Instagram. Social sensors may serve as a complementary source to physical sensors as (1) they can explain why or how specific events occurred, and (2) they can be deemed to be an alternative source in case that physical sensors malfunction or a sensor network cannot be afforded. However, one of the main challenges for social sensors is to know *where* a particular event has occurred. Social-media services rely on user preferences to geolocate their opinions, which is not really a widespread practice and, therefore, it limits the success of these techniques as early warning systems. In this paper, we analyze the spaCy named entity recognizer (NER), an open-source tool widely used by the community, to identify named entities in Spanish microtexts taken from social networks. The spaCy NER is based on Artificial Neural Networks, and our preliminary results show that further training should be undertaken to increase its accuracy. Indeed, it is well known that supervised methods are domain dependent, so their performance tends to decrease when dealing with target documents that come from a domain different from that of the training dataset. For this purpose, a training tool has been designed to automatically generate datasets suitable for spaCy NER's training with Twitter-based microtexts in Spanish. Using the dataset generated by this tool, the spaCy NER tool increases its accuracy to 0.7 F-score, defeating by a wide margin the use of other classic datasets such as AnCora, WIKINER or CONLL for training.

Keywords. named entity recognition, spaCy, social sensing, social-media analytics, geolocalization

1. Introduction

Mobile crowdsensing (MCS) is a recent research trend based on data collection from a large number of sensing devices [1]. In comparison with traditional physical or hard sensors, MCS is inexpensive, since there is no need to network deployment, and its spatio-temporal coverage is outstanding. Two different approaches of MCS have been distinguished, i.e. (1) mobile sensing, which leverages raw data generated from the hardware sensors that are embedded in mobile devices (e.g. accelerometer, GPS, camera or mi-

[1]Corresponding Author: Julio Fernández-Pedauye Campus de los Jerónimos S/N, 30107 Murcia, Spain; E-mail: jfernandez14@alu.ucam.edu.

crophone, among others), and (2) social sensing (or social networking), which leverages user-contributed data from social media. The latter considers participants as "social sensors", i.e. agents that provide information about their environment through social-media services after the interaction with other agents. Doran et al. [2] highlighted that social sensors may serve as a complementary or an alternative source to physical sensors. Although physical sensors can identify what happened, social sensors can provide an explanation of why or how specific events may happen. Moreover, social sensors can be deemed to be an alternative source in case that physical sensors malfunction or a sensor network cannot be afforded.

There are several challenges to overcome before unleashing all the potential of social sensors [3,4]. One of these challenges has to do with recognizing and extracting exact named entities (a.k.a. Named Entity Recognition, or NER) like Persons, Locations and Organizations. This is actually very useful to mining information from text for answering a given query [5,6]. Several NER approaches have been used in different Natural Language Processing (NLP) frameworks, such as Chainer[2], CoreNLP[3], TensorFlow [4] or Spacy [5]. However, NER procedures are not straightforward to be conducted on texts obtained from social networks, e.g., Twitter [7]. They are usually optimized to deal with long texts that are grammatically correct and their performance degrades due to issues like variability in spelling and presence of grammatically incomplete sentences, which are generally short and noisy. Furthermore, social networks are worldwide and each culture can have different ways of communicating on them, using different languages and even writing in mixed languages with combinations of a wide variety of slang [8]. This dramatically reduces the ability of the NER modules of these frameworks to detect named entities in this context.

The great majority of NER modules of NLP frameworks are based on artificial neural networks (ANNs), which are computing systems vaguely inspired by biological neural networks. ANNs consist of many simple computing units (artificial neurons) organized in sequential layers [9]. Deep Learning (DL) algorithms for developing NER procedures are being particularly successful in this direction (we refer the reader to [10] for an up-to-date review). The success of ANN-based systems are mainly found in the training dataset. There are many NER datasets in the literature. For example, CoNLL [11] was created from newswire articles in four different languages (Spanish, Dutch, English, and German) and focused on four types of named entities - PER (person), LOC (location), ORG (organization) and MISC (miscellaneous). Moreover, AnCora [12] is a multilingual corpus annotated at different linguistic levels consisting of 500,000 words in Catalan (AnCora-Ca) and in Spanish (AnCora-Es). More recently, NER tasks have also been performed on social-media data, e.g. Twitter [13]. Named entities on Twitter are also more variable (e.g. person, company, facility, band, sports team, movie, TV show, etc.), as they are based on user behavior on Twitter. Indeed, a specific workshop is held every year since 2015 for processing noisy user-generated text, such as that found in social media, online reviews, crowdsourced data, web forums, clinical records, and language-learner essays. However, to the best of our knowledge, all tasks related to noisy text are in English and we have not found any Spanish dataset that performs well for

[2]https://chainer.org/
[3]https://stanfordnlp.github.io/CoreNLP/
[4]https://www.tensorflow.org/
[5]https://spacy.io/

training NER modules based on ANNs. In this article, we develop a strategy to train the SpaCy NER module to increase its accuracy when dealing with noisy-generated text obtained from Twitter. First, a dataset is automatically generated from Spanish Wikipedia. Then, the SpaCy NER module is trained with this dataset before an in-depth evaluation is carried out to demonstrate that our solution provides a good framework for Spanish NER with noisy-generated text.

The remainder of this paper is structured as follows. Section 2 introduces the SpaCy NER module, together with a description of the dataset generator and the training procedure. Section 3 presents the evaluation of our model. Section 4 provides some conclusions and directions for future work.

2. Method

2.1. spaCy Named Entity Recognizer

spaCy is a free open-source library for NLP, written in Python. It includes several features such as NER, part-of-speech (POS) tagging, and dependency parsing, just to name a few. spaCy provides a statistical NER system, which labels contiguous spans of tokens. The spanish NER model identifies several named entities, including locations, organizations and people. Moreover, programmers can include additional classes to the NER module and update the model with new examples. The statistical model is based on a transition system called BILUO. The action of predicting the transition is structured as follows:

$$EMBED \rightarrow ENCODE \rightarrow ATTEND \rightarrow PREDICT.$$

First, spaCy represents words, looking at their context to recalculate these representations. Then, it creates a summary vector of a sentence based on this contextual representation, compiling everything into a single piece of information in order to predict the next transition. In this process, the first step is the embedding, where four characteristics of each word are used, i.e. it standardizes the string, prefix, suffix, and word shape characteristics (digits replaced by d, lower case characters with w etc.), in a technique known as ”hashing trick” or ”bloom embeddings”.

Once the out-of-context word embeddings are obtained, spaCy includes the neighboring words. In the encoding, a convolutional neural network (CNN) is used to concatenate and reduce the dimensionality of up to four words around each word in each case. In the attend stage, the extraction of characteristics is made by taking into account whether there has been a previously labelled entity and functions of arbitrary characteristics are produced. Finally, the prediction stage uses a multilayer perceptron (MLP) to predict the next action to be taken.

2.2. Dataset generation for training

The spaCy NER module is trained with the AnCora [12] and WikiNER [14] corpora, which are based on Spanish. As mentioned above, the results of NER procedures for social-network microtexts are not very good. This is particularly true for Spanish, where there are less resources such as datasets, NER modules, etc. To improve the accuracy of the spaCy NER model, we have developed a training corpus of 1,647,142 sentences

(467,126 after balancing) with annotated named entities, including people, places and organizations.

Wikipedia is a free-content encyclopedia with more than 1.5 million articles written by volunteers, texts that may need consensus among contributors and under the possibility of constant review and update. This makes Wikipedia a potential candidate to extract a corpus with thousands of updated phrases every day. Therefore, to achieve this training dataset automatically, an application has been developed to perform the following tasks:

1. **Read the Spanish Wikipedia data**: The application loads Wikipedia sources in XML format. Since the dump files can be several GBs, the scraping procedure must be performed carefully by loading the information in different chunks to avoid memory overflow.
2. **Storing the information**: The information is stored in a relational database, where there is a relationship between articles and the article type. This is important for eventually tagging the information with the targeted named entities.
3. **Cleaning up the information**: Once all the information has been structured in the database, the application proceeds to segment each article into sentences and label the entities found. The correct structuring in a database allows the parallelization and optimization of the process which is translated in a performance improvement.
4. **Annotation**: The annotation is carried out as follows. First, if there is a word that is found in the title of the article, the application assumes it is a noun. In this way, the named entity is tagged with the type of article, which is extracted from the XML Infobox tag. Second, based on the hypertext annotations for the internal reference links that Wikipedia uses to relate articles, we are able to unambiguously extract named entities with the Infobox tag of the article referenced. Finally, a previously trained tagger is used to get the POS tag of each word, where only nouns are selected as candidates. Potential candidates are compared with the titles of articles previously stored in the database; if an article is detected, the named entity is tagged with the associated Infobox tag of the article.
5. **Removing noise**: To avoid introducing noise, phrases that consist of only one entity or phrases below 5 words are discarded.
6. **Clustering**: Once we have the sentences tagged, a clustering of tags is carried out to classify them on the basis of three main tags: person, place and organization.

2.3. Training procedure

The training of the model is performed using the spaCy training tool. In this case, we only train the named-entity detection model, but it would be possible to train others. The corpus is first organized, so that the same amount of samples is maintained for each type of entity. In this way, the model will not be more likely to find one type of entity than another. Moreover, the corpus is then randomly shuffled and, to validate the training, it is separated into 70-10-20 % for training, validation and evaluation respectively. Five groups are created using the K-Fold Cross Validation technique to check the stability of the results. Mini-batches are created incrementally and are passed to the model for training. The number of batches will result in the number of iterations for each training session.

In addition to partitioning training into small batches, the dropout hyperparameter must be adjusted, a percentage used in a regularization technique to avoid over-adjusting the model to the training data. This is an efficient and low-cost computational technique in which some nodes (layer outputs) are "dropped" during training; after several tests, we concluded that 0.1 (10%) is the most appropriate value for this setting. It is worth highlighting that two different models have been developed, one more oriented to formal texts and the other more oriented to non-formal texts. The model trained for formal texts such as news corresponds to the described above, without any additional change. For the model trained for non-formal texts such as tweets, however, several tests have been carried out in the evaluation, concluding with a combination of phrases with all the text converted to lowercase and phrases with the text as it would be formally written. In this way, the model is able to deal with the language variety found in social networks.

3. Evaluation

Table 1 summarizes the datasets that were used for evaluation purposes. It should be noted that some of these datasets were used for both training and evaluation. In these cases, and as explained in the Evaluation section, we didn't use the whole dataset for both tasks, but only 80% of the dataset for training and 20% for evaluation, so that over-training results could be avoided. We consider AnCora, CONLL 2002 and WiKiNER as the main "gold standards", as occurs in many other NER methods that were evaluated with these datasets. On the other hand, 'Own Corpus' refers to the automatically tagged corpus described in the Method section, and 'Manually tagged Tweets' corresponds to the model that was evaluated with 500 manually annotated tweets in Spanish.

Corpus	Labeling Mode	Data extraction	Number of sentences
AnCora[6]	Manual	Spanish news wire and a balanced Castilian Spanish corpus (3LB)	17.376
CONLL 2002[7]	Manual	Spanish news wire articles	10.224
WikiNER [8]	Automated	Spanish Wikipedia articles	114.058
Own Corpus	Automated	Spanish Wikipedia articles	467.126 (1.647.142 unbalanced)
Manually tagged Tweets	Manual	Spanish Tweets	500

Table 1. Dataset description.

Table 2 shows the F-score of the spaCy NER module trained with different datasets to test performance. The F-score considers both the precision p, which is the number of correct positive results divided by the number of all positive results returned by the classifier, and the recall r, which is the number of correct positive results divided by the number of all samples that should have been identified as positive. The F-score is the harmonic mean of p and r, where the best score is 1 and the worst is 0. Results indicate that by taking 20% of the data set as evaluation the best results are obtained by Own Corpus, i.e. F-score of 0.93. For AnCora, WIKINER and CONLL 2002, the results decrease to an F-score of 0.83.

For the gold standards AnCora and CONLL 2002, results are rather similar among the different training procedures, i.e. 0.65-0.67. It is noteworthy to highlight that the evaluation has not been carried out with the same dataset that was employed for training. The last row is the Manually tagged Tweets dataset, where the best results are reported for the training with Own Corpus (i.e. F-score of 0.7).

Evaluation Set	Training dataset				
	AnCora + WikiNER (spaCy)	Own Corpus	AnCora	WikiNER	CONLL 2002
20% Corpus	0.87	0.93	0.84	0.90	0.83
AnCora	n.a.	0.65	n.a.	0.63	0.67
CONLL 2002	0.61	0.60	0.66	0.60	n.a.
Manually tagged Tweets	0.33	0.70	0.38	0.37	0.36

Table 2. F-score of the spaCy NER module trained with different combinations of training datasets and evaluated with the evaluation datasets; n.a. means that the score was not available, because the training dataset is the same as the evaluation dataset.

4. Conclusions and Future work

The detection of named entities, such as location, persons or organizations, is a fundamental task for the development of effective social-sensing tools. However, social sensors rely on social media information, and this information is generally too noisy and grammatically incorrect, limiting the ability of the NER procedures to recognize correct named entities in this context. In this paper, we have carried out an extensive evaluation of the well-known and open-source spaCy NER module. To increase its accuracy, an application has been proposed to automatically develop a training dataset from Spanish Wikipedia. Our results reveal that training the spaCy NER module with the dataset obtained with our application, the F-score improves to 0.7. We are fully aware that this is a preliminary work that evaluates a well-known NER tool with the aim to increase its accuracy. However, with the recent advances in deep-neural networks, we definitely think this is the way to go for opening new paths in developing effective NER tools.

In the future, we will work on disambiguation procedures that could actually help to provide useful information from social sensors. In addition, integrating rule-based methods into our approach could contribute to deriving the fine-grained semantics of named entities.

Acknowledgments

This work was partially supported by the Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia under Project 20813/PI/18, and by Spanish Ministry of Science, Innovation and Universities under grant RTI2018-096384-B-I00 and RTC-2017-6389-5.

References

[1] B. Guo, Z. Wang, Z. Yu, Y. Wang, N.Y. Yen, R. Huang and X. Zhou, Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm, *ACM computing surveys (CSUR)* **48**(1) (2015), 1–31.

[2] D. Doran, K. Severin, S. Gokhale and A. Dagnino, Social media enabled human sensing for smart cities, *AI Communications* **29**(1) (2016), 57–75.

[3] Z. Xu, L. Mei, K.-K.R. Choo, Z. Lv, C. Hu, X. Luo and Y. Liu, Mobile crowd sensing of human-like intelligence using social sensors: A survey, *Neurocomputing* **279** (2018), 3–10.

[4] D.E. Boubiche, M. Imran, A. Maqsood and M. Shoaib, Mobile crowd sensing–Taxonomy, applications, challenges, and solutions, *Computers in Human Behavior* **101** (2019), 352–370.

[5] A. Mansouri, L.S. Affendey and A. Mamat, Named entity recognition approaches, *International Journal of Computer Science and Network Security* **8**(2) (2008), 339–344.
[6] J. Bakerman, K. Pazdernik, A. Wilson, G. Fairchild and R. Bahran, Twitter geolocation: A hybrid approach, *ACM Transactions on Knowledge Discovery from Data (TKDD)* **12**(3) (2018), 1–17.
[7] J.J. Jung, Online named entity recognition method for microtexts in social networking services: A case study of twitter, *Expert Systems with Applications* **39**(9) (2012), 8066–8070.
[8] L.C. Yang, B. Selvaretnam, P.K. Hoong, I.K. Tan, E.K. Howg and L.H. Kar, Exploration of road traffic tweets for congestion monitoring, *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* **8**(2) (2016), 141–145.
[9] J. Schmidhuber, Deep learning, *Scholarpedia* **10**(11) (2015), 32832.
[10] V. Yadav and S. Bethard, A survey on recent advances in named entity recognition from deep learning models, *arXiv preprint arXiv:1910.11470* (2019).
[11] E.F. Sang and F. De Meulder, Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition, *arXiv preprint cs/0306050* (2003).
[12] M. Recasens and M.A. Martí, AnCora-CO: Coreferentially annotated corpora for Spanish and Catalan, *Language resources and evaluation* **44**(4) (2010), 315–345.
[13] T. Baldwin, M.-C. de Marneffe, B. Han, Y.-B. Kim, A. Ritter and W. Xu, Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition, in: *Proceedings of the Workshop on Noisy User-generated Text*, 2015, pp. 126–135.
[14] J. Nothman, N. Ringland, W. Radford, T. Murphy and J.R. Curran, Learning multilingual named entity recognition from Wikipedia, *Artificial Intelligence* **194** (2013), 151–175.