HORUS-NER: A Multimodal Named Entity Recognition Framework for Noisy Data

Diego Esteves[1,3], José Marcelino[1], Piyush Chawla[2], Asja Fischer[4], and Jens Lehmann[3]

[1] Farfetch.com, London, UK
{diego.esteves, jose.marcelino}@farfetch.com
[2] The Ohio State University, Columbus, 43210, USA
[3] SDA Research Group, Germany
[4] Ruhr-Universität Bochum, Faculty of Mathematics
44801 Bochum, Germany

Abstract. Recent work based on Deep Learning presents state-of-the-art (SOTA) performance in the named entity recognition (NER) task. However, such models still have the performance drastically reduced in noisy data (e.g., `social media, search engines`), when compared to the formal domain (e.g., `newswire`). Thus, designing and exploring new methods and architectures is highly necessary to overcome current challenges. In this paper, we shift the focus of existing solutions to an entirely different perspective. We investigate the potential of embedding word-level features extracted from images and news. We performed a very comprehensive study in order to validate the hypothesis that images and news (obtained from an external source) may boost the task on noisy data, revealing very interesting findings. When our proposed architecture is used: (1) We beat SOTA in *precision* with simple CRFs models (2) The overall performance of decision trees-based models can be drastically improved. (3) Our approach overcomes off-the-shelf models for this task. (4) Images and text consistently increased *recall* over different datasets for SOTA, but at cost of *precision*. All experiment configurations, data and models are publicly available to the research community at horus-ner.org

Keywords: Named Entity Recognition · WNUT · Noisy Text · Information Retrieval · Images · Text · Multi-modal

1 Introduction

In this paper, we address the problem of recognizing named-entity (NE) types in noisy data. While NER on formal domain (e.g. CoNLL) has been shown to be reasonably accurate – achieving average F_1 measure up to 90% [34] – most of the approaches for noisy data designed in the past years still heavily rely on carefully constructed orthographic features and language-specific resources, such as *gazetteers*. To bridge this gap, more recent work have proposed architectures based on LSTM networks. Although this not necessarily introduces SOTA

performance [16,19], the trained networks achieved very similar performance on a popular *newswire* corpora (respectively 88.83% and 90.94% on CoNLL-2003 test set). Besides supporting different languages with low effort, the great advantage of such (end-to-end) approaches lies in the fact that specific knowledge resources are not required (excepting for specific *embeddings*, which are - usually - language dependent), alleviating the dependency on manually annotated data and encoded rules. However, unlike *newswire*, *microblogs* often deal with more informal languages, which do not have such implicit linguistic formalism [29,22,14]. With respect to that – not surprisingly – the performance of SOTA degrades significantly in the noisy data domain, evidencing the sensibility of the proposed models when dealing with noisy and out-of-domain text. In recent work [5,21,29,20], F_1 ranging from 0.19 to 0.52 have been reported in the noisy domain. Hence, devising models to deal with linguistically complex scenarios such as *twitter* remains an open and very challenging problem to tackle, regardless of the architecture's design. In this paper, we extend previous work [12] to face this challenge through a novel perspective: we develop a framework that learns latent features from images and textual information to detect named entities, without requiring further engineering effort. This is obtained by extracting related information from an external source, given an input query string (e.g., a token). In this work we use the Web and DBPedia as external sources. We argue that images and text associated to a given token may contain missing information required to improve performance of NER on noisy data. Our main contribution is a framework that implements an enhanced methodology to extract, pre-process and generate feature vectors based on images and text - associated to each single token of a sentence. These vectors are then concatenated and used throughout several different NER architectures. Furthermore, a great advantage of our proposed model is that challenging (pre-processing) tasks, such as *text normalization* [1], is bypassed. To the best of our knowledge, this is the first *comprehensive* study in an attempt to derive and explore features based on images and news to improve NER on noisy data. The proposed methodology does not rely on *gazetteers*, *lookups* and *normalization* and also does not implement any encoded rules. Due to the nature of the generated feature vectors, we argue the outcomes of this work are of high relevance not only for NER on social media, but also to related (e.g., *entity linking* [27]) and also other downstream tasks [13]. Our experiments show that this has a direct positive impact in CRF and Decision Trees-based models, and the potential to improve overall B-LSTMs performance when more training data is available. As a contribution to the community we also released all metadata. The result is a word-level feature database for based on image and text. This database contains approx. 3 millions data features for more than 72.000 distinct English tokens and has been explored over 5.904 experiments in several different configurations. As consequence, we built an open-source framework dubbed HORUS, which we detail in the following sections. The data, metadata and code is released open-source and available at the project website: horus-ner.org.

2 Methodology and Features

First, one needs to note, that for each category of interest (i.e., a named entity class) one can identify a certain set of representative contents or objects, which have a high chance of being present in images belonging to nouns of this category. For instance, a name of a person has a high correlation to images containing *faces* whereas a name of a country has a high correlation to images containing *maps* or *landscapes*. Thus, named entities can be classified as belonging to a certain category by detecting these representative objects in the related images. Therefore, for each token $t \in$ a sentence $\mathcal{S}$ we extract a set of image and text feature vectors $\mathcal{F} = (\mathcal{F}_1, \ldots, \mathcal{F}_n)$ that serve as input features to a NER classifier. Following the foundations of our previous work [12], we use the Web to obtain (top 10) images and websites associated to a given token t. In this paper, we have extended and explored this methodology in a variety of ways: (1) We explored other clustering-based features (*Brown Clusters*); (2) We proposed and extended new visual features; (3) we performed *several* new experiments, obtaining further (valuable) insights; (4) we extended and included SOTA neural mechanisms in the underlying framework: (4.1) Topic Modeling + Convolution Neural Networks (CNN) for text classification [37] (4.2) CNN for object detection [32] (4.3) Topic Modeling over Word2vec [26] top v tokens (4.4) Cross-similarity measure over top v tokens and (4.5) Basic NN prediction statistics (5) We benchmark different NER classifiers in different gold-standard datasets.

Baseline Methodology: In our previous work [12] we perform the following steps: for each defined named entity category $c \in \mathcal{C}$ a set of text-based or image-based classifiers $\xi^c_m, m = 1, \ldots, M$ and $\Phi^c_l, l = 1, \ldots, L$, respectively, are applied. Given an element from $d \in \mathcal{D}_t$ (or $i \in \mathcal{I}_t$) each binary classifier outputs a prediction if the text (or image) belongs to a certain category c or not, i.e. $\Phi^c_l(i), \xi^c_m(d) \in \{-1, 1\}$. The text-based and image-based models produce the following feature sets: $\mathcal{TX}$ and $\mathcal{CV}$, respectively. These scores (feature vectors) $R^c_{\mathcal{D}_t}$ and $R^c_{\mathcal{I}_t}$ for all $c \in \mathcal{C}$ can now be used to construct the features $F(t)$ for the final classifier.

Improved Methodology: In the following we detail each additional feature implemented in our Framework. **1. Brown Clusters ($\mathcal{B}$)**: *Brown hierarchical word clustering* algorithm uses distributional information to group similar words [4]. It takes a *corpus* and outputs K clusters of word types in a hard-fashion, i.e., each token only appears in one cluster k. Essentially, it derives a tree graph with two kinds of information – the cluster of a word and the hierarchy between classes. Since the default number of clusters $K = 1000$ may not often yield optimal results (although widely considered as default value) [8], we performed some exploratory experiments to obtain the better hyper-parameter based on Derczynski et al. findings. The features are extracted by truncating the patches at [1:bits−2], e.g., the cluster path 1100101 yields features {1,11,110,1100,11001}. **2. Standard Features ($\mathcal{S}$)**: Besides *lexicon-based* such as Part-of-speech (POS) and *stop words*, *character-based* features, such as: "is numeric?", "initial capital?", "special character?" are also part of the classical features we study. For example the token "P@rty" would lead to vector

similar to $[0, 1, 1, \dots]$ **3. Topic Modeling + CNNs ($\mathcal{TX}_{nn}$)**: Originally designed for computer vision, Convolutional Neural Networks (CNNs) have subsequently been shown to be effective for NLP, achieving excellent results in diverse tasks, including sentence classification [17]. We trained a convolution neural networks with Topic Modeling [35] for text-classification due to its state of the art performance, which presents excellent results even with low hyper-parameter tuning [18]. The main idea is to classify each document (website) linked to a given token into a pre-defined number of "topics" (in this case, the labels PER, LOC and ORG), similarly to the $\mathcal{TX}$ module. Likewise, we used DBPedia to collect data for training the model. In practice, for each returned website w_i, we return a confidence score for w_i being labelled as one of the pre-defined classes. As result, we have a vector similar to: $[(0, 0.40170625), (1, 0.06669136), (2, 0.39819494), (3, 0.06670282), (4, 0.066704586)]$ where each key represents a certain topic (PER, LOC, ORG and OTHERS). **4. Seeds x Word2Vec ($\mathcal{TX}_{emb}$)** This model extracts the correlation between a pre-defined number of tokens (seeds) related to a certain class and nearest tokens to a given token t. We compute the distance in the intersection of the top 5 most similar words ($\mathcal{W}^t_{top} = s^t_1 \dots s^t_5$) to a given t with a set of *seeds* $\mathcal{E}$ defined by commonsense: $\mathcal{TX}^c_{emb} = \mathcal{W}^t_{top} \cap \mathcal{E}^c$. For instance, if - hypothetically - the token $t = Berlin$ has the following (5) nearest words $\mathcal{W}_5 = [Munich, Hamburg, Frankfurt, Germany, Dusseldorf]$. For each, we compute the average distance from each $e \in \mathcal{E}$. For e.g. LOC, we set the following vector = ["city", "country", "place", "beach", "mountain", "forest", "location"]. **5. Keyword Extraction ($\mathcal{TX}_{stats}$)** Similar to Section 2, this model outputs the likelihood of a certain token t belonging to a certain class c based on word distance. We extract the most frequent tokens from the set of documents $\mathcal{D}$ (websites) and cross-compute the distance from terms in $\mathcal{E}$. **6. Convolutional Neural Nets ($\mathcal{CV}_{nn}$)** As mentioned, CNNs is a state of the art technique for image recognition (e.g., detecting people or objects in a given image). For instance, the Inception model [33] achieves state of the art position, reaching 5.6% top-5 error rate on the ILSVR [30] classification challenge. Also, Places365 [38] performs state of the art in several datasets for place recognition. Another major advantage is that CNNs require little pre-processing when compared to standard approaches, such as SIFT [24] and SURF [2]. We re-trained this architecture to detect a list of pre-defined objects associated to each class c (as proposed in our previous work [12]). The classifier ϕ returns the probability distribution of a given image contain one of the desired classes.

3 Experimental setup

We benchmark our approach in four different gold-standard datasets ($\mathcal{DS}$) for NER in social media. The Ritter dataset and three datasets from the most famous Workshop on Noisy User-generated Text: WNUT-15, WNUT-16 and WNUT-17. Figure 1 depicts the pipeline that 1) performs the mapping of 3-MUC entities for all datasets. 2) enhances each of them with POS annotations.

3) Finally we get images and news associated with each of potential entity candidate.

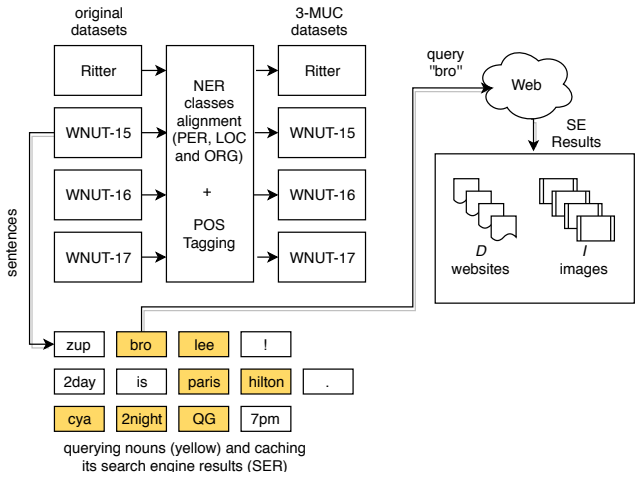

Fig. 1: Pre-processing step for the benchmark setup: adding POS tagger and filtering NNs. In the sequence, searching the Web to obtain images and websites for each filtered noun candidate and its compounds.

4) As a last step, we run the feature extraction modules (Section 2 - *Improved Methodology*) to generate the feature vectors associated to each token t. Then we have training data for our NER task. Following this last step, we implemented different weak and strong NER baselines, as follows:

(1) Off-the-shelf NER As a sanity check for defining baselines, we also briefly reported the performance of some off the shelf frameworks that claim state-of-the-art performance on NER: NLTK, Spacy, MITIE [5], OSU Twitter NLP, Stanford CoreNLP.

(2) Weak NER Baselines Two standard algorithms were used as weak baseline. A classical solution for sequence-to-sequence problems (CRF) and a Decision Tree-based method.

(3) Strong NER Baselines LSTMs represent cutting-edge architectures for NER both in *formal domains* [19,16] but also in *noisy data* [10] (despite performance drop when compared to *formal domains*). We implemented different SOTA LSTM-based models (B-LSTM+CRF [16], B-LSTM+CNN+CRF [25], Char+B-LSTM+CRF [19]).

In order to fully assess the impact of the proposed features and have a fair and proper comparison study, we performed a comprehensive benchmark on several *local* and *global* features. The full set of input features that we feed in our final classifier is given by the concatenation of two or more possible feature

[5] https://github.com/mit-nlp/MITIE

sets $\mathcal{F} = (\mathcal{TX} \cup \mathcal{CV} \cup \mathcal{TX}_{cnn} \cup \mathcal{CV}_{cnn} \cup \mathcal{TX}_{emb} \cup \mathcal{TX}_{stats} \cup \mathcal{B} \cup \mathcal{S})$. We grouped these features into several *experiment configurations* (`cfg01` to `cfg41`) (Table 1).

4 Results and Discussion

We first evaluate the performance of off the shelf tools on the selected datasets. As expected, results indicate that these solutions underperform when confronted with noisy data (AVG F1 from 0.2961 to 0.4878). This also confirms findings by Derczynski et al. [9].Therefore, as the average F1-measures are below current SOTA for the task on noisy data [12] and corroborating with past studies, we do not move on experiments for these frameworks[6].

In order to have a comprehensive and fair environment to benchmark different weak and SOTA NER algorithms, we split different feature configurations. The complete benchmark configuration has the following dimensions: $cfg \times (\mathcal{DS}_{train} + (\mathcal{DS}_{train} \times \mathcal{DS}_{test})) \times \mathcal{A}$; where cfg is the total of feature sets (i.e., distinct configurations), $\mathcal{D}_{train}$ is the total of training sets, $\mathcal{D}_{test}$ is the total of test sets and finally $\mathcal{A}$ is the total number of algorithms. This leads to the following number of experiments: $41 \times (4 + (4 \times 3)) \times 9 = 5.904$[7]. Table 1 summarizes the groups of experiments performed, i.e., different experiment dimensions. It helps to understand the the impact of images and textual features. It is worth mentioning that experiment configurations from 30 to 41 include the best Brown cluster in theirs respective pairs (e.g., `cfg30` represents `cfg18` including the best brown cluster). Therefore, they are let out of this table to improve readability.

Figure 2 shows the performance of CRF in different datasets/feature sets. The x-axis represents the different feature sets, while y-axis average of F1-measure[8]. To highlight the impact of the different groups of features, we categorize F1's in four ascending scales, from worse to the best: *red*, *yellow*, *gray* and *green*. Some patterns w.r.t. the addition of images and text as input features are clearly observable. First, standard textual features ($\mathcal{TX}$) have often a much worse performance when compared to standard image features ($\mathcal{CV}$) as well as in the combination of both, as observed in the following sets `cfg02`×`cfg03`×`cfg04`, `cfg06`×`cfg07`×`cfg08` and `cfg15`×`cfg16`×`cfg17`. This is at some extend expected since the adopted committee strategy [12] to classify *news* data is not a straightforward task. In this sense, a better solution might be taking into account probabilities instead of binary values. Moreover, we notice our improvement in the $\mathcal{TX}$ component (`cfg19`, `cfg20` and `cfg21`) outperform the similar features proposed by [12]. Among those, it is worth noting that the *text correlation* ($\mathcal{TX}_{stats}$, Section 2) has a greater impact than any other textual feature. This is due to the higher level of abstraction when computing word embedding

[6] For the sake of fair comparison, 3-MUC is also the base for experiments.

[7] 41 experiment configurations, 4 training sets (`Ritter`, `WNUT-15`, `WNUT-16` and `WNUT-17`), 3 test sets (`WNUT-15`, `WNUT-16` and `WNUT-17`) and 9 NER architectures (DT, RF, CRF, CRF-PA, LSTM, B-LSTM+CRF, Char+B-LSTM+CRF and B-LSTM+CNN+CRF)

[8] 3-fold cross-validation.

	Description	Configurations	Note
1	Standard	`cfg01`, `cfg05`, `cfg09-14`	usual features
2	Brown Clusters	`cfg09-14`	usual features + Brown
3	Images	`cfg03`, `cfg07`, `cfg15` `cfg18`, `cfg26`	computer vision (only)
4	Text	`cfg02`, `cfg06`, `cfg16` `cfg19-23`	text mining (only)
5	Images	`cfg03`, `cfg07`, `cfg15`	inspired by [12]
6	Text	`cfg02`, `cfg06`, `cfg16`	inspired by [12]
7	Images and Text	`cfg04`, `cfg08`, `cfg17`	inspired by [12]
8	Images	`cfg18`	this paper (Section 2)
9	Text	`cfg19-23`	this paper (Section 2)
10	Images and Text	`cfg24`, `cfg08`, `cfg17`	this paper (Section 2)

Table 1: The impact of images and textual features grouped by different experiment configurations. More detailed information for each configuration omitted due to page limit, but available on the project website horus-ner.org.

distances across *seeds* in a distance supervision fashion. Regarding the image detection component, introducing state-of-the-art computer vision algorithms ($\mathcal{CV}_{cnn}$) has also been beneficial to beat previous strategy ($\mathcal{CV}$), although without bringing major improvements as in the $\mathcal{TX}$. This is due to the *common-sense* rules proposed by [12] in this layer. Finally, the inclusion of tuned Brown clusters[9] along with proposed features shows to be beneficial to the performance. Overall, the best results were obtained from the concatenation of the previous and proposed features in conjunction with Brown clusters (`cfg41`).

Table 2 presents detailed results for each NER model. To recap, for each model, the first column (`cfg10`) in Table 2 represents the classic NER features (e.g. *lexical*); The configuration `cfg04` representing standard image and text features (proposed in [12]); Finally, in `cfg41` we see results for the image and text features proposed in this work. As expected, CRFs and SOTA NNs architectures performed best and overall images and news (`cfg04` and `cfg41`) have a great impact in CRFs, helping to overcome SOTA (LSTMs) w.r.t. *precision*. The comparison shed light on the impact of our proposed features (best configuration, `cfg41`) when compared to the broadly implemented (standard) NER features (`cfg10`) and the features proposed in our previous work [12] (`cfg04`). We can see that overall the additional features introduced in this work clearly improves the performance of the majority of the NER models - both weak and strong baselines - (DT, RF, CRF, B-LSTM+CRF) in all data sets. CRF-PA slighly overperformed the standard CRF, confirming findings presented by Derczynski et al [7]. However, it is worth noticing the ability of NNs to improve recall, the major challenging in noisy-data [1].

The results confirm that the proposed features consistently boost the performance of the models in the majority of the experiments. It is worth noting

[9] $\mathcal{B}_{best}$, `cfg30-41`

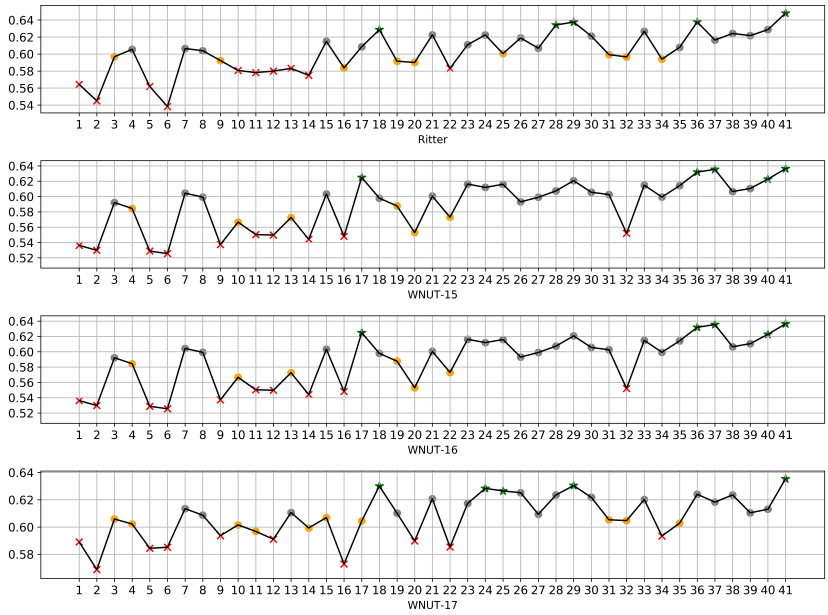

Fig. 2: The CRF performance (`cfgXX` × F1) over different feature sets. Each sub-graph representing one dataset. Performances of configurations using our methodology are positively impacted.

the substantial impact in the CRF-based model. Our proposed features (`cfg41`) improves *Lexical + Brown Cluster* and [12] in more than 90% of the cases (and at least similar in 100% of the cases). Moreover, we notice that a basic CRF architecture with the best feature configuration (`cfg41`) outperforms a state-of-the-art B-LSTM architecture w.r.t. *precision*. The same feature set also positively impacted *recall* of B-LSTM in all experiments. Finally, we trained a B-LSTM+CRF architecture with an expanded set created merging all data sets. We removed duplication from the union of the respective *training*, *dev* and *test* sets, i.e., occurrences of overlap sentences. The SOTA B-LSTM+CRF F1-measure has achieved 0.5217. Integrating our methodology has increased the results to ↑0.5352 (`cfg04`) and ↑0.5352 (`cfg41`). Despite modest results, this benchmark indicates that images and news are definitely a great asset to improve both precision and overall performance of NER architectures in noisy contexts.

5 Related work

Named Entity Recognition is a sub task of information extraction which seeks to identify entities in textual content. Over the past few years, the problem of recognizing named entities in noisy data has been addressed by different approaches that have emerged specifically designed to better perform on short and

NER Benchmark on Noisy Data																			
		Weak Baselines									Strong Baselines								
Dataset		Decision Trees			Random Forest			CRF			B-LSTM [16] CRF			B-LSTM [19] C+CRF			B-LSTM [25] C+CRF+CNN		
cfg ⟶		10	04	41	10	04	41	10	04	41	10	04	41	10	04	41	10	04	41
Ritter	P	0.48	+2%	+4%	0.51	+1%	+24%	0.73	+5%	+7%	0.77	+1%	−3%	0.81	−5%	−1%	0.81	−5%	−5%
	R	0.49	+1%	+3%	0.48	−1%	−2%	0.58	−8%	−2%	0.63	+5%	+5%	0.59	+5%	+4%	0.62	+3%	+5%
	F	0.49	+1%	+3%	0.49	+4%	+7%	0.58	+2%	+7%	0.68	+1%	+1%	0.67	+1%	+1%	0.69	−1%	+1%
WNUT-15	P	0.49	+2%	+5%	0.52	+7%	+25%	0.72	+7%	+9%	0.72	−4%	−2%	0.77	−3%	−4%	0.78	−4%	−5%
	R	0.50	+0%	+5%	0.49	+0%	+1%	0.48	−1%	+6%	0.69	+1%	+1%	0.65	+2%	+2%	0.66	+2%	+2%
	F	0.50	+0%	+5%	0.50	+5%	+9%	0.56	+2%	+8%	0.68	+0%	+0%	0.69	+0%	−1%	0.71	−1%	−2%
WNUT-16	P	0.49	+1%	+6%	0.52	+14%	+23%	0.72	+7%	+9%	0.72	−4%	−2%	0.77	−3%	−3%	0.78	−4%	−6%
	R	0.50	+1%	+6%	0.48	+0%	+2%	0.48	−1%	+6%	0.69	+0%	+1%	0.65	+2%	+2%	0.66	+2%	+2%
	F	0.49	+1%	+6%	0.50	+5%	+10%	0.56	+2%	+8%	0.69	−1%	+0%	0.69	+0%	+0%	0.71	−1%	−2%
WNUT-17	P	0.44	+3%	+7%	0.47	+13%	+24%	0.76	+2%	+1%	0.76	−2%	−2%	0.76	+0%	−2%	0.77	−3%	−3%
	R	0.45	+4%	+6%	0.44	+3%	+4%	0.50	+0%	+5%	0.63	+1%	+1%	0.64	+0%	+1%	0.62	+1%	+1%
	F	0.44	+4%	+6%	0.45	+6%	+12%	0.60	+0%	+4%	0.67	+0%	+0%	0.69	+0%	−1%	0.67	+0%	−1%

Table 2: The performance measure's improvements (*green*) and decreases (*red*) in different datasets, feature sets (`cfg`) and NER models. Results are represented in a color gradient of 5 points interval. 0% represents a tiny improvement i ($0.1\% \leq i \leq 0.99\%$), which is not representative, although technically not zero. The percentage variation both in `04` and `41` columns are according to the baseline performance for each NER architecture (column `10`).

noisy texts, such as `T-NER` [29] and `TwiterIE` [3]. The first performs tokenization, POS tagging and noun-phrase chunking before using topic models to find named entities whereas the second – an extension of GATE ANNIE [6] – implements an NLP pipeline customized to microblog texts at every stage (including Twitter-specific data import and metadata handling). Liu et al. [22] propose a gradient-descent graph-based method for text normalization and recognition. Likewise, these approaches are highly dependent on hand-crafted rules. Most recently, approaches followed Lample et al. [19] architecture based on BiLSTMs. Limsopatham and Collier [20] proposed a neural architecture for NER on *microblogs*, which combines a bidirectional LSTM with an CRF achieving a F_1 measure of 52.41 for *English* text. Models supporting other languages were proposed, however, similar performances (min-max F_1 measure) have also been observed across different languages other than English, such as French, Portuguese and Chinese, for instance ([23] - 21.28 − 58.59, [28] - 24.40 − 52.78 and [15] - 44.29 − 54.50, respectively). Esteves et al. [12] proposed a methodology to encode image and news features into NER architectures, showing promising preliminary results. [36] followed the same idea to detect entities in Twitter, but just analyzing existing images associated to a given tweet, which drastically restricts the approach. In the three years the NER in social media benchmarking workshop W-NUT ran, in social media[10], modest results have been reported by a vast number of different NER architectures: 16.47 − 56.41, 19.26 − 52.41 and

[10] http://noisy-text.github.io

$39.98 - 41.86$ (min-max F_1 in WNUT 2015, 2016, 2017, respectively) [1,31,11]. Therefore, although neural architectures pose a good choice to outperform standard architectures (e.g. CRFs), the task is still far from being solved in noisy contexts.

6 Conclusion

In this paper, we benchmark and extend a novel multilevel NER approach in different ways. We integrate features which rely on state-of-the-art computer vision and text mining techniques. We show that its major advantage is the fact that it does not rely on hand-crafted features and domain-specific knowledge. In order to support this claim, we conducted a massive number of experiments in the same computational environment with different feature sets and over different gold-standard data. In traditional NER architectures (e.g. CRF), the proposed features have proved feasible to notably improve its overall model performance (i.e. $F1$) and, when compared to SOTA, beat in *precision*. SOTA had improved in *recall*, but at expense of *precision*. However, when benchmarking the models across different training-test sets (which is often not tested in most research publications) the images and news also proved to be beneficial for the task. We also confirmed that this solution performs better than existing off-the-shelf frameworks on the noisy context, as expected. The main issue w.r.t. SOTA neural networks for this domain seems to be the size of the available training data sets (WNUT). As future work we plan to explore the combination of the models given their probability distributions, extending also the analysis to more named entity classes. Also, since it shows to be language-agnostic, we would like to explore new languages other than English. Finally, we plan to integrate this architecture into Named Entity Disambiguation and Linking frameworks.

References

1. Baldwin, T., de Marneffe, M.C., Han, B., Kim, Y.B., Ritter, A., Xu, W.: 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. pp. 126–135 (2015)
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Comput. Vis. Image Underst. **110**(3), 346–359 (Jun 2008). https://doi.org/10.1016/j.cviu.2007.09.014, http://dx.doi.org/10.1016/j.cviu.2007.09.014
3. Bontcheva, K., Derczynski, L., Funk, A., Greenwood, M.A., Maynard, D., Aswani, N.: Twitie: An open-source information extraction pipeline for microblog text. In: RANLP. pp. 83–90 (2013)
4. Brown, P.F., Desouza, P.V., Mercer, R.L., Pietra, V.J.D., Lai, J.C.: Class-based n-gram models of natural language. Computational linguistics **18**(4), 467–479 (1992)
5. Chang, Y.s., Sung, Y.H.: Applying name entity recognition to informal text. Recall **1**, 1 (2005)

6. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., et al.: Text processing with gate (version 6)(2011). University of Sheffield Department of Computer Science **15** (2011)
7. Derczynski, L., Bontcheva, K.: Passive-aggressive sequence labeling with discriminative post-editing for recognising person entities in tweets. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers. pp. 69–73 (2014)
8. Derczynski, L., Chester, S., Bøgh, K.S.: Tune your brown clustering, please. In: International Conference Recent Advances in Natural Language Processing, RANLP. vol. 2015, pp. 110–117. Association for Computational Linguistics (2015)
9. Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., Bontcheva, K.: Analysis of named entity recognition and linking for tweets. Information Processing & Management **51**(2), 32–49 (2015)
10. Derczynski, L., Nichols, E., van Erp, M., Limsopatham, N.: Results of the wnut2017 shared task on novel and emerging entity recognition. In: Proceedings of the 3rd Workshop on Noisy User-generated Text. pp. 140–147. Association for Computational Linguistics (2017). https://doi.org/10.18653/v1/W17-4418, http://aclweb.org/anthology/W17-4418
11. Derczynski, L., Nichols, E., van Erp, M., Limsopatham, N.: Results of the wnut2017 shared task on novel and emerging entity recognition. In: Proceedings of the 3rd Workshop on Noisy User-generated Text. pp. 140–147 (2017)
12. Esteves, D., Peres, R., Lehmann, J., Napolitano, G.: Named entity recognition in twitter using images and text. In: ICWE Workshops (2017)
13. Esteves, D., Reddy, A.J., Chawla, P., Lehmann, J.: Belittling the source: Trustworthiness indicators to obfuscate fake news on the web. In: Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). pp. 50–59 (2018)
14. Gattani, A., Lamba, D.S., Garera, N., Tiwari, M., Chai, X., Das, S., Subramaniam, S., Rajaraman, A., Harinarayan, V., Doan, A.: Entity extraction, linking, classification, and tagging for social media: A wikipedia-based approach. Proc. VLDB Endow. **6**(11), 1126–1137 (Aug 2013). https://doi.org/10.14778/2536222.2536237, http://dx.doi.org/10.14778/2536222.2536237
15. He, H., Sun, X.: A unified model for cross-domain and semi-supervised named entity recognition in chinese social media. In: AAAI. pp. 3216–3222 (2017)
16. Huang, Z., Xu, W., Yu, K.: Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991 (2015)
17. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
18. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751. Association for Computational Linguistics (2014). https://doi.org/10.3115/v1/D14-1181
19. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360 (2016)
20. Limsopatham, N., Collier, N.: Bidirectional lstm for named entity recognition in twitter messages. WNUT 2016 p. 145 (2016)
21. Liu, X., Zhang, S., Wei, F., Zhou, M.: Recognizing named entities in tweets. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. pp. 359–367. Association for Computational Linguistics (2011)

22. Liu, X., Zhou, M., Wei, F., Fu, Z., Zhou, X.: Joint inference of named entity recognition and normalization for tweets. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. pp. 526–535. Association for Computational Linguistics (2012)
23. Lopez, C., Partalas, I., Balikas, G., Derbas, N., Martin, A., Reutenauer, C., Segond, F., Amini, M.R.: Cap 2017 challenge: Twitter named entity recognition. arXiv preprint arXiv:1707.07568 (2017)
24. Lowe, D.G.: Object recognition from local scale-invariant features. In: Computer vision, 1999. The proceedings of the seventh IEEE international conference on. vol. 2, pp. 1150–1157. Ieee (1999)
25. Ma, X., Hovy, E.: End-to-end sequence labeling via bi-directional lstm-cnns-crf. arXiv preprint arXiv:1603.01354 (2016)
26. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
27. Moussallem, D., Usbeck, R., Röder, M., Ngonga Ngomo, A.C.: MAG: A Multilingual, Knowledge-base Agnostic and Deterministic Entity Linking Approach. In: K-CAP 2017: Knowledge Capture Conference. p. 8. ACM (2017)
28. Peres, R., Esteves, D., Maheshwari, G.: Bidirectional lstm with a context input window for named entity recognition in tweets. In: Proceedings of the Knowledge Capture Conference. p. 42. ACM (2017)
29. Ritter, A., Clark, S., Etzioni, O., et al.: Named entity recognition in tweets: an experimental study. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1524–1534. Association for Computational Linguistics (2011)
30. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
31. Strauss, B., Toma, B., Ritter, A., de Marneffe, M.C., Xu, W.: Results of the wnut16 named entity recognition shared task. In: Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT). pp. 138–144 (2016)
32. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., et al.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Cvpr (2015)
33. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR. pp. 1–9. IEEE Computer Society (2015)
34. Tkachenko, M., Simanovsky, A.: Named entity recognition: Exploring features. In: KONVENS. pp. 118–127 (2012)
35. Wallach, H.M.: Topic modeling: beyond bag-of-words. In: Proceedings of the 23rd international conference on Machine learning. pp. 977–984. ACM (2006)
36. Zhang, Q., Fu, J., Liu, X., Huang, X.: Adaptive co-attention network for named entity recognition in tweets. In: AAAI (2018)
37. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. In: Advances in neural information processing systems. pp. 649–657 (2015)
38. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)