Improving Named Entity Recognition using Deep Learning with Human in the Loop

Ticiana L. Coelho da Silva
Insight Data Science Lab
Fortaleza, Ceara, Brazil
ticianalc@ufc.br

Regis Pires Magalhães
Insight Data Science Lab
Fortaleza, Ceara, Brazil
regismagalhaes@ufc.br

José Antônio F. de Macêdo
Insight Data Science Lab
Fortaleza, Ceara, Brazil
jose.macedo@dc.ufc.br

David Araújo Abreu
Insight Data Science Lab
Fortaleza, Ceara, Brazil
araujodavid@lia.ufc.br

Natanael da Silva Araújo
Insight Data Science Lab
Fortaleza, Ceara, Brazil
natanael_silva@alu.ufc.br

Vinicius Teixeira de Melo
Insight Data Science Lab
Fortaleza, Ceara, Brazil
viniciusteix@alu.ufc.br

Pedro Olímpio Pinheiro
Insight Data Science Lab
Fortaleza, Ceara, Brazil
pedro.olimpio@alu.ufc.br

Paulo A. L. Rego
Federal University of Ceara
Fortaleza, Ceara, Brazil
pauloalr@ufc.br

Aloisio Vieira Lira Neto
Brazilian Federal Highway Police
Fortaleza, Ceara, Brazil
aloisio.lira@prf.gov.br

ABSTRACT

Named Entity Recognition (NER) is a challenging problem in Natural Language Processing (NLP). Deep Learning techniques have been extensively applied in NER tasks because they require little feature engineering and are free from language-specific resources, learning important features from word or character embeddings trained on large amounts of data. However, these techniques are data-hungry and require a massive amount of training data. This work proposes Human NERD (stands for Human Named Entity Recognition with Deep learning) which addresses this problem by including humans in the loop. Human NERD is an interactive framework to assist the user in NER classification tasks from creating a massive dataset to building/maintaining a deep learning NER model. Human NERD framework allows the rapid verification of automatic named entity recognition and the correction of errors. It takes into account user corrections, and the deep learning model learns and builds upon these actions. The interface allows for rapid correction using drag and drop user actions. We present various demonstration scenarios using a real world data set.

1 INTRODUCTION

Named Entity Recognition (NER) is a challenging problem in Natural Language Processing (NLP). It corresponds to the ability to identify the named entities in documents, and label them with one of entity type labels such as person, location or organization. Given the sentence "Trump lives in Washington DC", traditional NER taggers would identify the mentions 'Trump' and 'Washington DC' to person and location labels, respectively. NER is an important task for different applications such as topic detection, speech recognition, to name a few.

However, there is a long tail of entity labels for different domains. It is relatively simple to come up with entity classes that do not fit the traditional four-class paradigm (PER, LOC, ORG, MISC), such as, in Police report documents, *weapon type* is none of the above. For these cases, labeled data may be impossible to find. Even, orthographic features and language-specific knowledge resources as gazetteers are widely used in NER, such approaches are costly to develop, especially for new languages and new domains, making NER a challenge to adapt to these scenarios[10].

Deep Learning models have been extensively used in NER tasks [3, 5, 9] because they require little feature engineering and they may learn important features from word or character embeddings trained on large amounts of data. These techniques from Deep Learning are data-hungry and require a massive amount of training data. While the models are getting deeper and the computational power is increasing, the size of the datasets for training and evaluating are not growing as much [14].

In this work, we address this problem by including humans in the loop. Several methods have been proposed to improve the efficiency of human annotations, for instance in computer vision applications [11, 14] and NER tasks via active learning [2, 7, 8, 12, 13]. Those methods are promising for NER but still leave much room for improvements by assuming the annotation cost for a document measured regarding its length, the number of entities or the number of user annotation actions, for instance. While these are important factors in determining the annotation and misclassification cost, none of them provide the ability to create and incrementally maintain deep learning models based on iterative annotation. Indeed, all of them expect NER tasks to have very few labels. Prodigy [1] is a promising annotation tool that works on entity recognition, intent detection, and image classification. It can help to train and evaluate models faster. However, we could not explore Prodigy, since it is not free. In this work, our goal is to provide an interactive framework called Human NERD (stands for Human Named Entity Recognition with Deep learning) to assist the user in NER classification tasks from creating a massive dataset to building/maintaining a deep learning NER model. Human NERD provides an interactive user interface that allows both the rapid verification of automatic named entity recognition (from a pre-trained deep learning NER model) and the correction of errors. In cases where there are multiple errors, Human NERD takes into account user corrections, and the deep learning model learns and builds upon these actions. The interface allows for rapid correction using drag and drop user actions. We need to point out that our framework consider two types of user: reviewer and data scientist. The reviewer is a domain

© 2019 Copyright held by the owner/author(s). Published in Proceedings of the 22nd International Conference on Extending Database Technology (EDBT), March 26-29, 2019, ISBN 978-3-89318-081-3 on OpenProceedings.org.
Distribution of this paper is permitted under the terms of the Creative Commons license CC-by-nc-nd 4.0.

expert that can correct possible classification errors and enrich the labels while data scientist focuses on tuning the models. In the next Sections, we provide more details and a screencast is available at YouTube[1].

To the best of the authors' knowledge, this work is the first that simplifies the task of annotating datasets, minimizing supervision, and improving the deep learning NER model, which in turn will make the overall system more efficient. To achieve this end, we first propose the framework in Section 2. Then, we present the demonstration scenarios in Section 3. Section 4 draws the final conclusions.

2 HUMAN NERD

For a large document collection, Human NERD keeps the user engaged in the process of building a large named entity dataset. The input to the framework is a set of documents to annotate and a set of NER labels. The output is a set of annotated documents, a deep learning model for entity recognition and the evaluation metrics values that can estimate the *operational cost* during the annotation time and the *gain* regarding model accuracy.

We incorporate deep learning NER models as the Entity Recognizer models from Spacy [2] framework into Human NERD to reduce the cost of human time dedicated to the annotation process. Indeed, these models have led to a reduction in the number of hand-tuned features required for achieving state-of-the-art performance [6]. Human NERD can also incorporate models such as [3, 5].

Human NERD suggests a potential entity annotation in every interaction loop, and the user as a reviewer can accept or reject individual entities. He/she can also tag a new excerpt from the document text with an entity that was not suggested by the NER model. The feedback is then used to refine the model for the next iteration and enrich the dataset. Our framework simplifies the task of building large-scale datasets, minimizing supervision, and improving the deep learning NER model, which in turn will make the overall system more efficient.

The general workflow of Human NERD follows five main steps (overview in Figure 1): (1) collecting a large set of unlabeled documents; (2) the current NER model recognizes and annotates entities in the document according to labels drawn from a given set of entity classes L (i.e., person, location, among others); (3) user as a reviewer can accept or reject individual entities; he/she can also manually label the document according to L; (4) generating a deep learning NER model for each iteration; (5) estimating the *gain* over the iterations and the *loss*, for improving the model accuracy and the operational cost during annotation time, respectively.

First step. Starting from a large pool of unlabeled documents $T = \{t_1, ..., t_m\}$ collected from different and heterogeneous resources (as Twitter, Wikipedia, Police reports, among others), where t_i is a variable-length sequence of input symbols $t_i = \{w_1, ..., w_n\}$. A sequence of consecutive w_i with the same label λ_j are treated as one mention of such label. Input symbols are word tokens w_i drawn from a given vocabulary V. Let $L = \{\lambda_j : j = 1...q\}$ denotes the finite set of labels in a learning task. We aim at annotating t_i with a sequence of output symbols $Y = \{y_1, ..., y_p\}$. Output symbols are labels λ_j drawn from a given set of entity classes L.

[1]https://youtu.be/KGJeWKO_3Xw
[2]https://spacy.io/

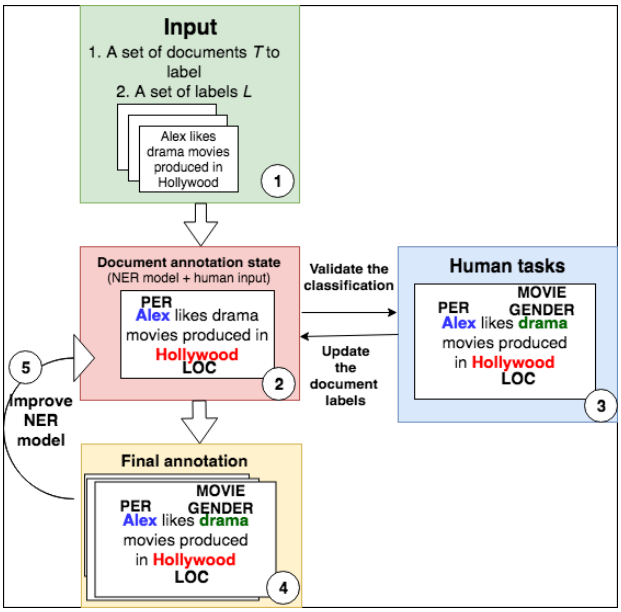

Figure 1: Overview of Human NERD. Given a set of documents for annotating as input, the system alternates between NER model classification and requesting user feedback through human tasks. The outcomes are the documents annotated to improve the NER model.

Second step. Human NERD acquires entity classes (i.e., person, location, among others) for T from a deep learning NER model as [3, 5] or using Spacy's models (i.e., Entity Recognizer model - which is trained using multilinear perceptron and convolutional network). The deep learning model is initially trained with $D = \{(x_i, Y_i) : i = 1...m\}$, a set of labeled training examples x_i, where $Y_i \subseteq L$ the set of labels of the i-th example. At this step, the pre-trained model classifies the entity mentions on $T = \{t_1, ..., t_m\}$ using the labels described on L and outputs $O = (t_i, Y_i) : i = 1...m$, where $t_i \in T$ and $Y_i \subseteq L$ the set of labels of the i-th document.

Third step. Human NERD presents to the user an interactive web-based annotation interface used for adding entity annotations or editing automatic pre-annotations in O. As the entities are labeled in O, users (as reviewers) then accept or reject these to indicate which ones are true. Each document $t_i \in O$ is presented to one user. Thus no two users labeled the same document at the same instant of time. This step outputs O with its user corrections. Human NERD logs both the time elapsed during the labeling process, and the number of labeling *actions* taken for each document t_i, i.e., it keeps track of actions like labeling an entity or removing a label incorrectly assigned to an entity.

Fourth step. Based on the user corrections, the NER model can learn and improve from O. In the interactive interface, the user as a data scientist can demand Human NERD to incrementally update the pre-trained deep NER model or build a new one from O. At this step, the system logs the *accuracy* and *loss* over the iterations of the model construction. These data are useful in the next step.

Fifth step. By putting humans in the loop, Human NERD has a *gain*, since the users help to improve the NER model by validating a new massive training set over time. With such data, we expect to increase the NER model accuracy and decrease its

error in short-term. On the other hand, the *drawback* is the human effort by adding entity annotations or editing pre-annotations (third step). To estimate the framework *loss*, we measure the user efforts for annotating the document regarding its length, number of characters, number of entities or number of user annotation actions (editing or adding new entities). As much the deep NER model learns, Human NERD framework becomes more efficient and minimizes the user supervision.

To measure the agreement between the deep NER model (second step outputs) and the user (third step outputs), Human NERD computes the kappa coefficient (k). Cohen's kappa [4] measures the agreement between two raters who each classifies N items into L mutually exclusive categories.

$$k = \frac{p_0 - p_e}{1 - p_e} \tag{1}$$

such that

$$p_0 = \sum_{i=1}^{|L|} \frac{n_{ii}}{n}; p_e = \frac{1}{n^2} \sum_{i=1}^{|L|} n_{i.} \times n_{.i}; \tag{2}$$

where n_{ii}, $n_{.i}$ and $n_{i.}$ are the number of entities: labelled by the NER model and the user in category i, labelled by the NER model in category i, labelled by the user in category i, respectively. Let n be the total number of entities in T. The closer k is to one, the greater the indication that there is an agreement between the model and the user (as a reviewer). On the other hand, if k is closer to zero, the greater the chance of the agreement be purely random. We expect to increase the NER model accuracy over time and to improve the kappa coefficient to a value as close to one as possible.

3 DEMONSTRATION DETAILS AND SCENARIOS

We cover various scenarios that demonstrate the usefulness of Human NERD. An interactive interface is the access point for the user to: (i) upload several unlabeled documents; (ii) validate each individual entity annotation from the output generated by a pre-computed model executed over those documents; The user can also manually label new entities; (iii) re-build the deep NER model to learn from the annotated documents after the user feedback; and (iv) estimate the *gain* and *loss* regarding the improved NER model and the human efforts.

Human NERD considers two types of user: reviewer and data scientist. The former can perform only the task (ii) described above. The later can perform (i), (iii) and (iv). Moreover, the data scientist can remove or add new labels and texts into L and T, respectively.

To examine the quality of our framework, we use a real dataset with unlabelled texts from Police reports. The dataset contains real-world stories in Portuguese language regarding homicides from Fortaleza city (Brazil). We started by using a pre-computed NER model called Wikiner from Spacy framework which includes only four labels: PER, LOC, ORG, and MISC. We removed MISC and added more than 20 label classes like firearm, melee weapon, wrongful death, among others. After that, the Wikiner model classified the reports according to the labels, and the expert reviewers by means of a web interface added and edited entity annotations. From those reviews, Human NERD created a new deep learning NER model for police domain. This data example confirms that Human NERD can be applied in different contexts and languages.

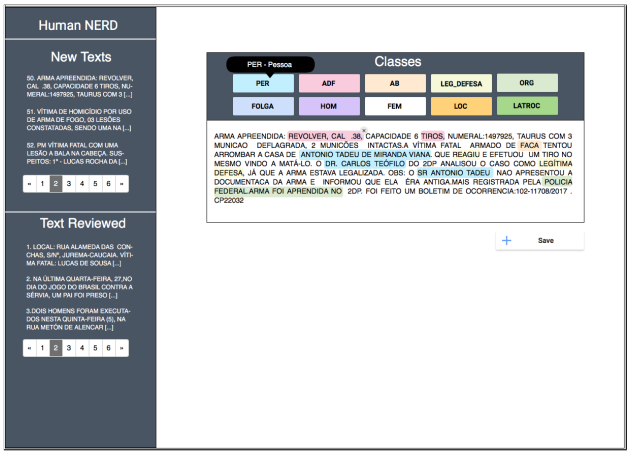

Figure 2: User (as a reviewer) validates the documents annotated by the pre-trained NER model.

It is worth to mention that Human NERD improved Wikiner model with the human help, including its extension for covering new labels and a new data domain. The demonstrated scenarios are as follows.

A. Reviewer in the Loop

Human NERD puts the pre-trained model and the human in the loop, so they can actively participate in the process of improving the NER model, using what both know. The model learns as the user goes, based on the labels the user assigns to text excerpts. As shown in Figure 2, the reviewer as a user interacts with the framework through a click-and-drag interface. The framework initially presents a set of documents annotated by the pre-trained model according to a set of label classes L. Each label receives a different color for better visualization.

The reviewer validates one document per time. If he/she agrees with the annotations of the current document (shown in Figure 2), he/she saves the document. Human NERD appends each validated document in a historical dataset, which will be used to improve the deep learning NER model. The reviewer can reject individual entities, in this case, he/she can click on the "x" button. If the reviewer identifies an entity not annotated by the model, he/she can manually label it. In this case, first, he/she should click on the class label (on top of the Figure 2), then the class will appear in evidence. After that, the reviewer selects the sequence of words in the document to annotate. Most annotation tools avoid making any suggestions to the user, to prevent biasing the annotations. Human NERD takes the opposite approach: demands the user to annotate but as little as possible considering that the NER model is continuously improving over time.

B. Data scientist in the Loop

Human NERD offers a full view of the imported or already trained NER models to the data scientist (Figure 3). A model is active if the framework is currently using it to annotate the documents during the user review (the previous scenario). The framework reports the status of review and train processing. The former corresponds to how many documents the reviewer already validated, and the later to how many epochs already finished during the model training. The data scientist can request the framework to train, duplicate, remove, edit the settings and visualize the statistics of NER models which were imported or trained by the framework.

Figure 3: Data scientist visualization.

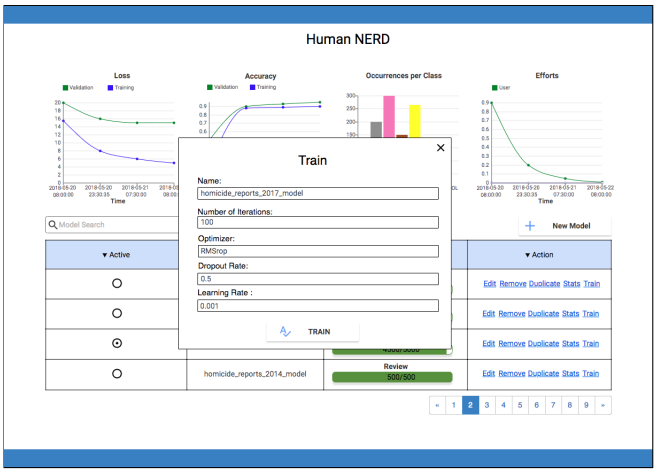

Figure 4: Human NERD enables the data scientist to train a model.

Figure 4 shows the interface presented to the data scientist which enables him/her to train a new model or update a previous one based on the texts validated by the reviewer. Human NERD requires for this functionality the inputs: number of iterations (epochs), optimizer, dropout rate, and learning rate. When the data scientist asks to edit the model settings, he/she can remove or add new label classes.

When the data scientist requests the visualization of statistics, Human NERD framework reports the *gain*, since we expect that the NER model improves over time, and the *user efforts* for annotating documents. Figure 5 summarizes those statistics collected over time when Human NERD trains a model or during documents annotation/validation performed by the reviewer.

4 CONCLUSION

In this demonstration, we proposed a framework called Human NERD to improve named entity recognition models by using a human in the loop. Human NERD relies on deep neural architectures for NER tasks that are free from language-specific resources (e.g., gazetteers, word clusters id, part-of-speech tags) or handcrafted features (e.g., word spelling and capitalization patterns). We validate Human NERD framework with a real data set from

Figure 5: Statistics reports: the model *gain* and *loss*.

Police reports in the Portuguese language, and we built upon Wikiner from Spacy a new deep learning NER model for this domain. A future work would be to improve the deep learning NER models by using ensemble techniques. Another direction is to provide a collaborative framework to allow multiple concurrent active models and reviewers.

ACKNOWLEDGMENTS

This work has been supported by FUNCAP SPU 8789771/2017 research project.

REFERENCES

[1] 2017. Prodigy: A new tool for radically efficient machine teaching. https://explosion.ai/blog/prodigy-annotation-tool-active-learning. (2017). [Online; accessed 9-January-2019].
[2] Shilpa Arora, Eric Nyberg, and Carolyn P Rosé. 2009. Estimating annotation cost for active learning in a multi-annotator environment. In *Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*. Association for Computational Linguistics, 18–26.
[3] Jason PC Chiu and Eric Nichols. 2015. Named entity recognition with bidirectional LSTM-CNNs. *arXiv preprint arXiv:1511.08308* (2015).
[4] Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement* 20, 1 (1960), 37–46.
[5] Cıcero dos Santos, Victor Guimaraes, RJ Niterói, and Rio de Janeiro. 2015. Boosting Named Entity Recognition with Neural Character Embeddings. In *Proceedings of NEWS 2015 The Fifth Named Entities Workshop*. 25.
[6] John Foley, Sheikh Muhammad Sarwar, and James Allan. 2018. Named Entity Recognition with Extremely Limited Data. *arXiv preprint arXiv:1806.04411* (2018).
[7] Ashish Kapoor, Eric Horvitz, and Sumit Basu. 2007. Selective Supervision: Guiding Supervised Learning with Decision-Theoretic Active Learning.. In *IJCAI*, Vol. 7. 877–882.
[8] Trausti Kristjansson, Aron Culotta, Paul Viola, and Andrew McCallum. 2004. Interactive information extraction with constrained conditional random fields. In *AAAI*, Vol. 4. 412–418.
[9] Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354* (2016).
[10] Xuezhe Ma and Fei Xia. 2014. Unsupervised dependency parsing with transferring distribution via parallel guidance and entropy regularization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 1337–1348.
[11] Olga Russakovsky, Li-Jia Li, and Li Fei-Fei. 2015. Best of both worlds: human-machine collaboration for object annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2121–2131.
[12] Burr Settles, Mark Craven, and Lewis Friedland. 2008. Active learning with real annotation costs. In *Proceedings of the NIPS workshop on cost-sensitive learning*. Vancouver, Canada, 1–10.
[13] Dan Shen, Jie Zhang, Jian Su, Guodong Zhou, and Chew-Lim Tan. 2004. Multi-criteria-based active learning for named entity recognition. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 589.
[14] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. 2015. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365* (2015).