Multilingual Named Entity Recognition

Tony DiPadova and Steven Jiang

May 30, 2018

Department of Computer Science, Dartmouth College

Abstract: Mapping companies, such as TomTom, OpenStreetMap, and MapBox, face the difficult issue of how to extract pertinent information from news and social media to automatically update their maps. Tweets about new restaurants or Facebook postings about road construction introduce novel information relevant to map updates. Named Entity Recognition (NER) is a key part of processing and discerning the relevancy of this information. Additionally, given the global and interconnected natures of news and social media, this kind of information may be presented in different languages. We propose a Language-Independent Named Entity Recognition (NER) model using Facebook Multilingual Unsupervised and Supervised Embeddings (MUSE word embeddings) as features for a bi-directional Long Short Term Memory (Bi-LSTM) network with a Conditional Random Field (CRF) classifier. Using the CoNLL shared task corpora as our training data, we train a bi-LSTM to extract features from sentences and implement a classifier to tag words with Named Entity Inner-Outer-Beginning (IOB) tags.

1. Introduction

The prevalence of Spanish is rapidly approaches that of English in the United States. The internet allows people, who reside in different countries and speak different languages, to instantaneously communicate with each other. We live in a multilingual world and there exists a growing need to be able to process text in multiple languages. One of the most important Natural Language Processing (NLP) tasks is Named Entity Recognition (NER). The task of Named Entity Recognition is multi-faceted in its goals and benefits. For example, it automates scraping social media for mapping companies and indexing regions of interest for search engine companies. Additionally, it forms the basis of many other Natural Language Processing tasks, such as knowledge representation and question-answer tasks.

Traditionally, Named Entity Recognition models require extensively annotated corpora, which include named entity tags, parts-of-speech, and other relational features. Although these corpora do exist English, they don't exist in every language. Additionally, these corpora are expensive to compile and annotate, especially if they are created for specialized tasks. Furthermore, models trained using these corpora typically do not generalize well to more recent Named Entity Recognition applications, such as recognizing named entities in Tweets. As a result, our goal is to create a generalized Named Entity Recognition model that can be used across multiple languages and does not require to be trained on custom built and annotated corpora.

2. Related Work

The Conference on Natural Language Learning (CoNLL) is a top-tier conference, yearly organized by SIGNLL, ACL's Special Interest Group on Natural Language Learning. Language-Independent Named Entity Recognition was presented at CoNLL in both 2002 and 2003 as the Shared Task in which researched competed. In 2002, twelve teams of researchers attempted to create models, using words and parts-of-speech, to generalize NER for Spanish and Dutch[1]. The researchers mostly used simple classifiers such as decision trees, SVMs, AdaBoost, and Hidden Markov Models. The best results from the CoNLL-2002 shared task were achieved using an AdaBoost classifier with fixed-depth decision trees. In 2003, sixteen teams of researchers attempted the same task on English and German[2]. This time, the researchers mostly used Maximum Entropy Models, Hidden Markov Models, Conditional Markov Models, Voting Classifiers, and Bagging Classifiers. All of these models used supervised classification based on word, part-of-speech, named entity tag. Additionally, these models incorporated n-gram approaches based on the aforementioned three features. Still, in 2003, the majority of the models relied on external sources such as gazetteers or other look up tables.

Current state-of-the-art techniques, as described by Lample et al., use Long Short Term Memory (LSTM) networks for feature extraction with Conditional Random Fields (CRF) as classifiers[3]. In the paper, Lample et al. use pre-trained word embeddings, without any additional feature engineering, as inputs. They were able to achieve state-of-the-art performance in the task of Named Entity Recognition for Spanish, Dutch, English, and German, using the datasets as the CoNLL-2002 and CoNLL-2003 Shared Tasks.

3. Data

Splitting the Datasets

In order to train our language-independent Named Entity Recognition model, we used the corpora from the CoNLL-2002 and the CoNLL-2003 Shared Tasks. More specifically, we used the English, German, and Spanish datasets in order to train and validate our model in multiple languages. We were particularly interested in evaluating model performance on these datasets, in order to compare and contrast the results of training in Germanic versus Romance languages. The datasets were split, by language, into training, validation, and test sets. We created a full training set by combining the training sets for the separate languages; however, we kept the validation sets separated by language to evaluate our model's performance on different languages separately. As a result, our training dataset consisted of $\sim$10,000 sentences per language and each validation dataset consisted of $\sim$2,000 sentences in the corresponding language.

Corpus Format

The corpora we used contained sentences that were split up into individual words, each of which was tagged with its associated part-of-speech, phrase feature (i.e. Noun Phrase, Verb Phrase), and named entity feature. Each named entity feature is comprised of an Inner-Outer-Beginning (IOB) tag and the type of entity that the word represents. The IOB tags are Inner (I), Outer (O), and Beginning (B), representing that word is part of a named entity, not part of a named entity, and the start of a named entity, respectively. The types of entities represented in the corpora are PER (person), ORG (organization), LOC (location), and MISC (miscellaneous). For example, the first word in a person's name would be tagged "B-PER", and all following words that are part of that named entity would be tagged "I-PER". Additionally, singleton named entities are tagged with the "B" IOB tag. For example, "SIGNLL" would be tagged "B-ORG". Finally, all words that are not named entities would be tagged with "O". Some corpora included other relational information; however, we ignored that information for the purposes of this project. Figure 1 shows an example of a tree parse of IOB tags, in relation to their representation in the corpus.

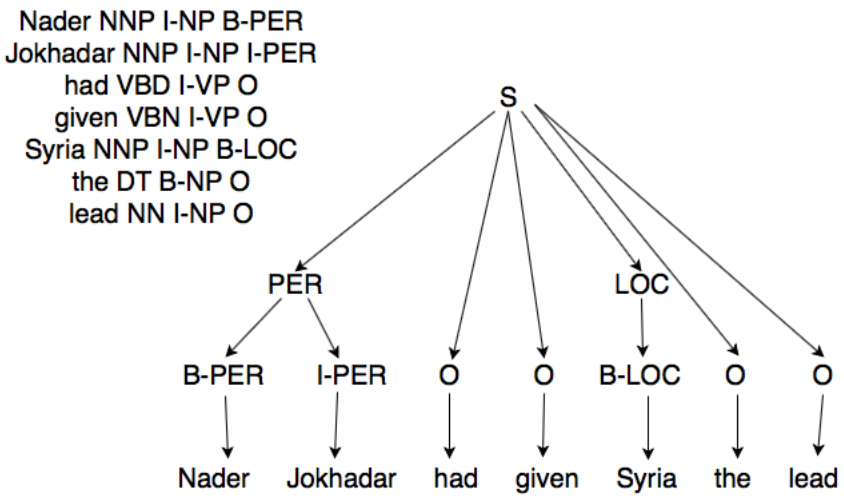

Figure 1: NER parsed sentence from the normalized CoNLL-2003 English Test-A corpus

Data Preprocessing

The standard format for representing Named Entity tags in corpora is described above. However, the English and German versions of the CoNLL-2003 corpus was originally formatted slightly differently. Rather than using "B" tags to denote the beginnings of named entities, the corpus only used "B" tags to separate consecutive "I" tags that were not part of the same entity. Additionally, the corpus tagged singleton named entities with "I" tags. Since the rest of our data was formatted in the way described above, we preprocessed and standardized the English and German versions of the CoNLL-2003 corpus to use "B" tags for singleton named entities and to denote the beginning of *any* named entity.

4. Methods

Feature Engineering

Similar to Lample et al., we elected to use pretrained word-embeddings. We sourced our word embeddings from the Facebook Research Multilingual Unsupervised Supervised Embedding (MUSE) project. MUSE uses Facebook's FastText algorithm to generate word embeddings from Wikipedia text in 30 languages and then uses supervised and unsupervised techniques to align these 30 vector spaces into a single vector space. The resulting vectors have 300 dimensions. We elected to use these embeddings as features to ensure that words from different languages with similar meanings are clustered together in the embedding vector space. In this way, we are able to reduce the effect of differences in vocabulary between languages.

Handling Out-of-Vocabulary (OOV) Words

We normalized our corpus by converting every word to lowercase and replacing all numbers with '0', a method described in Lample et al. To handle out-of-vocabulary (OOV) words, or words for which our pre-trained embeddings have no vector, we use a method described in *Convolutional Neural Networks for Sentence Classification*[4]. For each OOV word, we randomly initialize and generate a new 300-dimensional vector in a "hyperball", with a radius of 0.25 around the origin of our vector space. We chose to use 0.25 to ensure that these randomly generated vectors have similar variances

as the pre-trained vectors. It should also be noted that, even if the same out-of-vocabulary word appears multiple time, we reassign it to a new vector every time it is processed. For example, if we see a sentence with the word "thneed", we assign it to a randomly generated 300-dimensional vector. The next time we process the word "thneed" we assign it a *different* random 300-dimensional vector. We do this because assigning the word "thneed" to a static vector is tantamount to adding it to our vocabulary and training our model to recognize it based on its semantic meaning. Then, any vector that is similar to our randomly generated vector for "thneed" might be categorized in a similar way, resulting in over-fitting problem for out-of-vocabulary words. As a result, we choose to treat each out-of-vocabulary word as if we have never processed it before.

The Model

We architected and built a neural network with a bi-directional Long-Short Term Memory (LSTM) layer, a time-distributed dense layer, and a Conditional Random Field (CRF) classifier. The bi-directional LSTM layer represents the words in context by looking at previous and next words. The time-distributed dense layer has 100 hidden nodes and acts as a fully connected layer. Finally, the CRF layer acts as a classifier and enforces the IOB tag ordering rules. We implemented our LSTM-CRF model using Python and the Keras library, with Tensorflow backend. After tuning our hyperparameters on our validation set, we chose to use a dropout rate of 0.1 and a recurrent dropout rate of 0.3. We used dropout to introduce noise in order to better generalize our model to multiple languages. We optimized our model using the Root Mean Square Propagation (RMSProp) optimizer and the CRF loss function. Figure 2 shows a diagram of our model.

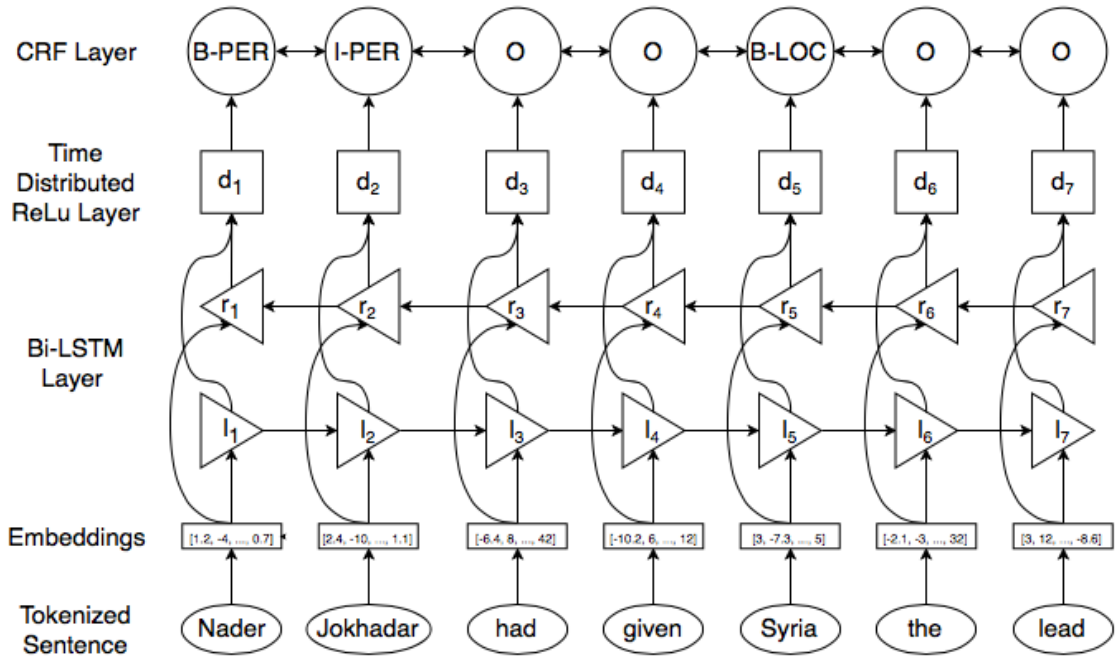

Figure 2: Bi-LSTM model given word embeddings as features with a Time Distributed Dense Layer with ReLu activation and a CRF layer for prediction

The bi-directional nature of our LSTM allows us to represent words in context. Through using MUSE word embeddings, we are able to reduce the magnitude of cross-vocabulary problems

that are typically present in Language-Independent Named Entity Recognition tasks. However, the word embeddings don't necessarily eliminate the issue of grammatical differences between different languages. For example, in German and English, adjectives come before nouns, but, in Spanish, adjectives come after nouns. Grammatical differences, similar to the one described, could negatively affect the ability of our model to recognize named entities. We address this problem by using a bi-LSTM layer. The bi-LSTM layer processes the embeddings in the sentence from left to right, and then from right to left. Furthermore, it concatenates the resulting vectors together to represent the word in context with every other word, preserving the order of the sentence. This allows us to not only model the relationships between words in a sentence, but also generalize our model across languages to different grammatical structures.

Experiments

First, we trained our model on embedding vectors generated from the English CoNLL-2003 training dataset and evaluated our model using the validation datasets of English, German, and Spanish, separately. Next, we trained our model on embedding vectors generated from the combined English, German, and Spanish training dataset from both the CoNLL-2002 and CoNLL-2003 Shared Tasks. Once again, we evaluated our model using the validation datasets of English, German, and Spanish, separately. Then, we compared the results of the two experiments to determine the extent to which our English-trained model could generalize the task of predicting named entities in other languages.

Additionally, we also experimented with formulating the task of Named Entity Recognition in two different ways, as a multi-class classification task and as a binary classification task. For the multi-class classification task, we used all available named entity tags within the corpus (PER, ORG, LOC, and MISC), as well as the IOB tags (I, O, B). For the binary classification task, we attempted to tag each word with NE or O, representing Named Entity and Not Named Entity, respectively. It is worth noting that this task was not strictly binary because due to the nature of IOB tags, we tagged words with O, B-NE, or I-NE, giving us three tags in total. However, the essence of the task was binary because we were interested in determining whether or not a word was a part of a named entity. We evaluated each task using each of the aforementioned models, which resulted in four distinct combinations of tasks and models (English-Binary, English-Multiclass, Full-Binary, Full-Multiclass). We measured the accuracy, precision, recall, and F1 score of each model performing each task on each different language.

5. Results

Table 1 shows the accuracy, precision, recall, and F1 score for each model's (English-trained and Full corpora trained) performance on each task (binary tags and all tags) in each different language (English, German, and Spanish). First, we compare the results of training our model on the English corpus and the results of training our model on the full corpora of English, German, and Spanish. As expected, we found that training our model on the full corpora significantly improved accuracy, precision, recall, and F1 score in both the binary classification and multi-class classification tasks when evaluating the model in German and Spanish. Additionally, for the multi-class classification task, when evaluating in English, we notice that training on the full corpora results in only a slight decrease in precision, recall, and F1 score, while maintaining the same level of accuracy, compared to training on the English corpus. Interestingly enough, for the binary classification task, training our model on the full corpora achieved a higher tagging precision and a comparable F1 score to training the model on the English corpus, when evaluated in English. We believe that the F1 scores for evaluating the binary classification task in English and Spanish, which we achieve through training our model on the full corpora, being greater than 0.85 indicates relatively strong results. Furthermore, we believe that the F1 scores for evaluating the multi-class classification task in English and Spanish, which we achieve through training our model on the full corpora, being greater than 0.70 indicates moderately strong results.

Table 1: Results

		English				German				Spanish			
	Data	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
All Tags	English	**0.990**	**0.887**	**0.742**	**0.801**	0.868	0.224	0.344	0.219	0.954	0.585	0.402	0.452
	Full	**0.990**	0.879	0.741	0.792	**0.979**	**0.723**	**0.440**	**0.517**	**0.975**	**0.817**	**0.680**	**0.723**
Binary Tags	English	**0.991**	0.939	**0.881**	**0.908**	0.870	0.404	0.594	0.424	0.961	0.832	0.653	0.717
	Full	**0.991**	**0.947**	0.874	0.907	**0.980**	**0.874**	**0.660**	**0.737**	**0.980**	**0.885**	**0.874**	**0.879**

After plotting confusion matrices for each of the test sets (English, German, and Spanish) from the fully trained model, we found that some types of tags were easier to predict than others. Figures 3, 4, and 5 show confusion matrices for testing in English, German, and Spanish, respectively. The axes are labeled with the IOB tags such that the y-axis represents the true label of the tag and the x-axis represents the predicted label. The cells along the diagonal show the "hit rate" of each label; a perfect prediction would show 1.0 in every cell along the diagonal. Additionally, the the cells not along the diagonal show the "miss rate" of each label; a perfect prediction would show 0.0 in every cell not along the diagonal. These confusion matrices clearly indicate that some tags, such as PER, were easier to predict in every language, compared to other tags, such as MISC, which were difficult to predict in every language. Additionally, we noticed that some tags are easier to predict in certain languages, such as LOC in English or ORG in Spanish. Finally, we found that our models trained on the full corpora of languages did an excellent job of predicting when a word was not a part of a named entity.

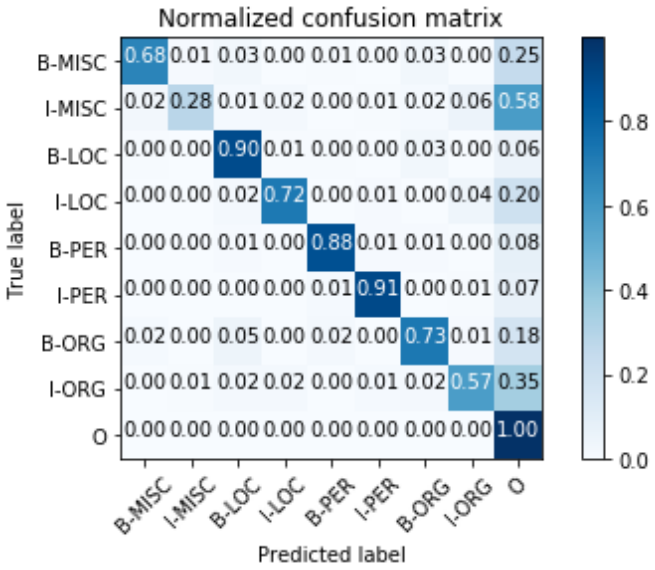

Figure 3: English Results from Full Training

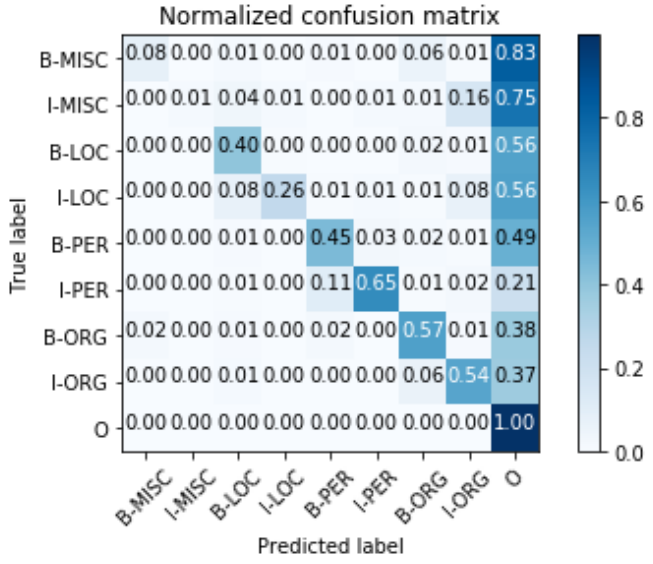

Figure 4: German Results from Full Training

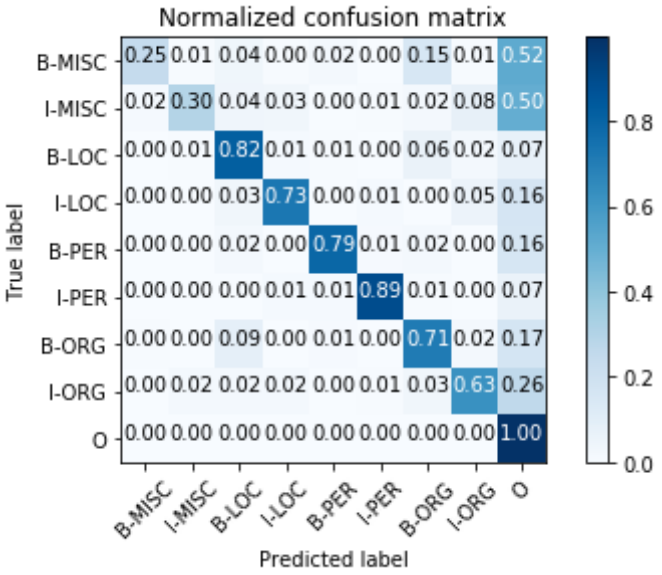

Figure 5: Spanish Results from Full Training

Discussion

We present a model for multi-lingual Named Entity Recognition. The model is comprised of a bi-directional LSTM layer, a time-distributed dense layer, and a conditional random field classifier. We train our model on the CoNLL-2003 English corpus, as well as the full corpora, which consists of the CoNLL 2002 and 2003 corpora in English, Spanish, and German. We evaluate the model's performance on both binary classification and multi-class classification of named entities. We find that our model trained on the full corpora yields strong results for the binary classification task when evaluated in English and Spanish. Additionally, we find that our model trained on the full corpora yields moderate results for the multi-class classification task when evaluated in English and Spanish.

When the model processes a sentence such as "My name is Tony DiPadova and I study at Dartmouth College", it tags "Tony DiPadova" with the PER tag and "Dartmouth College" with the ORG tag. Furthermore, when processing "Mi chiamo Tony DiPadova e studio a Dartmouth College" using the Italian MUSE vectors as features, the model still tags "Tony DiPadova" as PER and "Dartmouth College" as ORG, despite not having been trained on Italian. This is even more interesting due to the fact that "Tony", "DiPadova", and "Dartmouth" are all unknown words for all the embeddings, while "College" is also an unknown word in non-English embeddings.

These results indicate that our model is able to generalize semantic structures that indicate named entities, rather than just recognizing named entity clusters. Since the MUSE vectors are aligned in a single vector space, similar words occupy similar vector spaces. For example, vectors of words such as "name", "llamo", and "chiamo" or "study", "estudio", and "studio" would have relatively high cosine similarity measures. This allows us to analyze semantic structure and meaning of words in context, in languages that our model has never before seen. We should note that we did not have an Italian corpus, so we were not able to record extensive test metrics for Italian. However, we were able to see the efficacy of the model in practice as a proof of concept.

6. Limitations

Due to the fact that our model uses MUSE word embeddings as inputs, we must load the word embeddings for every intended language before making predictions. As each language-embedding file is a little more than 0.5GB in size, loading more than four languages into memory at a single time is not feasible for an average user. Our model would therefore be most effective if run as a REST API from a machine that could keep the word embeddings in memory for long periods of time.

Additionally, we were limited in the set of languages we could train and test in due to the fact that the CoNLL 2002 and 2003 Shared Tasks only provided corpora in English, Spanish, and German. Furthermore, due to time and resource constraints, we were also limited in the amount of time we could dedicate toward training our model. It's entirely possible that our results could have improved with more epochs of training.

7. Future Work

As previously mentioned, the scope of our research was limited by the availability of data and time/resource constraints. Future work could expand upon the current work by training and testing a multilingual Named Entity Recognition model on more languages than just English, Spanish, and German. Additionally, there has been some work on completely unsupervised NER methods.[5] While our model is able to generalize to multiple languages, it is still a supervised model and therefore does require an annotated corpus in at least one language. Future work in this area might not require annotated corpora at all. It would be interesting to see whether or not we can create models that rely solely on unsupervised methods to extract named entities from multilingual texts. Finally, since recent work in NER systems has been able to achieve a relatively high level of accuracy, much of the research on named entities has shifted towards Named Entity Disambiguation (NED). Named Entity Disambiguation is the task of linking named entities to instances in a knowledge base.[6] This is particularly useful for knowledge representation and question-answer systems. Future work could extrapolate from the lessons learned in this study and apply them to the task of multilingual Named Entity Disambiguation.

References

1. Sang, Erik F Tjong Kim. Intro duction to the CoNLL Shared Task Language-Independent Named Entity Recognition. CNTS. University of Antwerp. ACL. USA. 2002. http://www.aclweb.org/anthology/W02-2024
2. Sang, Erik F. Tjong Kim & Fien De Meulder. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. CNTS. University of Antwerp. ACL. USA. 2003. http://delivery.acm.org/10.1145/1120000/1119195/p142-tjong_kim_sang.pdf
3. Lample, Guillaume & Miguel Ballesteros & Sandeep Subramanian & Kazuya Kawakami & Chris Dyer. Neural Architectures for Named Entity Recognition. NLP Group. Carnegie Mellon University. DARPA. USA. 2016. https://arxiv.org/pdf/1603.01360.pdf
4. Kim, Yoon. Convolutional Neural Networks for Sentence Classification. NYU. USA. 2014. http://www.aclweb.org/anthology/D14-1181
5. Munro, Robert & Christopher D. Manning. Accurate Unsupervised Joint Named-Entity Extraction from Unaligned Parallel Text. Stanford NLP Group. Stanford University. USA. 2012. https://nlp.stanford.edu/pubs/MunroManning2012ner.pdf
6. Chang, Angel X. & Valentin I. Spitkovsky & Christopher D. Manning & Eneko Agirre. A comparison of Named-Entity Disambiguation and Word Sense Disambiguation. Stanford NLP Group. Stanford University. IXA. USA. 2016. https://nlp.stanford.edu/pubs/chang2016entity.pdf