# UQAM-NTL: Named entity recognition in Twitter messages

**Ngoc Tan Le**
Computer Science Faculty
UQÀM, Montreal, Canada
le.ngoc_tan@courrier.uqam.ca

**Fatma Mallek**
Computer Science Faculty
UQÀM, Montreal, Canada
mallek.fatma@courrier.uqam.ca

**Fatiha Sadat**
Computer Science Faculty
UQÀM, Montreal, Canada
sadat.fatiha@uqam.ca

## Abstract

This paper describes our system used in the 2nd Workshop on Noisy User-generated Text (WNUT) shared task for Named Entity Recognition (NER) in Twitter, in conjunction with Coling 2016. Our system is based on supervised machine learning by applying Conditional Random Fields (CRF) to train two classifiers for two different evaluations. The first evaluation aims at predicting the 10 fine-grained types of named entities; while the second evaluation aims at predicting no type of named entities. The experimental results show that our method has significantly improved Twitter NER performance.

**Keywords:** *Twitter, named entity recognition, machine learning, CRF.*

## 1 Introduction

Named entity recognition is one of the key information extraction tasks. This concerns the identification of named entities and the classification of named entities such as person, organisation, location, time and event (Nadeau and Sekine, 2007). The existing standard NER systems are usually trained on formal texts, such as the newswire. However, these linguistic tools do not work well on the new and challenging noisy tweet messages because the style of Tweet messages is short (length upto 140 characters) and unstructured. The content is highly noisy, contains many ill-formed words and covers several topics. Sometimes even human annotators do not have enough context to disambiguate the entities reliably (Baldwin et al., 2015).

Our team at UQAM is interested by social media analysis research within the NLP context (Sadat et al, 2014a; Sadat et al, 2014b; Sadat, 2013). Thus, our participation at the 2nd Workshop on Noisy User-generated Text (WNUT) shared task for Named Entity Recognition in Twitter, in conjunction with Coling 2016, is very fruitful. This shared task consists of two separate evaluations aiming at: (1) predicting the 10 fine-grained types of named entities, and (2) predicting the no-type of named entities. For both evaluation, our system is based on supervised machine learning and trained with a sequential labeling algorithm, using Conditional Random Fields (CRF). Our contribution here consists of the proposal about the new features regarding the polysemy count based on an lexicalized semantic network and ontology, such as an encyclopedic dictionary, and the longest n-gram sequence length of each word in a tweet based on a language model about actualities, news and also syntactic features by parsing each tweet.

The present paper is organized as follows: In the section 2, we report the proposed system about the features extraction. The experimentations and the evaluations are presented in the sections 3 and 4 respectively. Finally, the section 5 summarizes our work and gives future perspective.

## 2 Features extraction

In this section, we describe a variety of features that are used during the modelling of our NER system for tweets. Our model is composed of the following features: *(1) orthographic, (2) lexical and (3)*



*Proceedings of the 2nd Workshop on Noisy User-generated Text*,
pages 197–202, Osaka, Japan, December 11 2016.

*syntactic features as well as (4) POS (part-of-speech) tags, (5) polysemy count and (6) longest n-gram length*.

(1) The **orthographic features** templates are as follows:

- **Affixes:** The suffixes of the current word are extracted with length upto 4 characters from its last character. The prefixes of the current word are extracted with length upto 4 characters from its firts character.
- **Capitalization:** There are three patterns for the current word and two other patterns for the previous word and the next word.
- **Punctuation** and **Digit:** There are six patterns in order to check whether the current word, the previous word and the next word contain punctuation marks and/or numbers.

(2) The **lexical features** consist of the number of occurrences of a word in the sentence, the number of occurrences of the lemma of a word in the sentence and the word in lowercase format.

(3) The **syntactic features** consist of the constituent labels and the distance of a word to root. The word's constituent label (*Constituent Label*) and its depth in the constituent tree (*Distance to Root*) are extracted using a syntactic parser. We used Berkerley parser (Petrov and Klein, 2007).

(4) The **POS tags features** in the task of NE recognition contain many useful information for classifying and predicting named entities. In this work, we use a POS tagger, named TwitIE (Bontcheva et al., 2013), providing the output in the same format given by the organizers. Predicted tags are used as features as follows:

- **POS tags:** The search space windows is 4. There are a combination of patterns with the current word, the two previous words and the two next words and their corresponding POS tags: $(w_{-2}, p_{-2})$, $(w_{-1}, p_{-1})$, $(w_0, p_0)$, $(w_{+1}, p_{+1})$, $(w_{+2}, p_{+2})$.

(5) The **polysemy count**: We extract the polysemy count, which is the number of meanings of a word in a given language. The BabelNet (Navigli et al., 2012) API is used to extract this feature.

(6) The **longest n-gram length**: We seek to get the length (n + 1) of the longest left sequence ($w_{i-n}$) concerned by the current word ($w_i$) and known by the language model (LM) concerned. For example, if the longest left sequence $w_{i-2}$, $w_{i-1}$, $w_i$ appears in the longest n-gram value for $w_i$ will be 3. This value ranges from 0 to the max order of the LM concerned (Servan et al., 2015). We used a language model from WMT-2016[1]:

| n-gram | #tokens |
|---|---|
| 1 | 167 333 |
| 2 | 3 330 169 |
| 3 | 5 129 254 |

Table 1. Statistics of the n-gram of the language model from WMT-2016

These features were chosen because of their relevance in several NLP tasks such as POS tagging, chunk tagging and NE recognition, following the WNUT 2015 workshop[2]. The features for the tokens, in the patterns, were based on uni-grams, bi-grams, tri-grams and within a context window of size 3 (previous token, current token, next token).

## 3 Experimental setup

### 3.1 Preparation of corpus

Our model is trained with the data provided by the 2nd shared task workshop organizers.

[1] Language model from WMT-2016: http://www.quest.dcs.shef.ac.uk/quest_files_16/lm.tok.en.tar.gz
[2] WNUT 2015 workshop: http://noisy-text.github.io/2015/

| Dataset | #tweets | #tokens |
|---|---|---|
| *train_2016* (= *train_2015* + *dev_2015*) | 2 394 | 37 619 |
| *dev_2016* (= *test_2015)* | 1 000 | 16 429 |
| *new_dev_2016* | 420 | 14 400 |
| *test_2016* | 3 856 | 48 782 |

Table 2. Statistics of the corpora provided by the 2nd shared task at WNUT-2016

There are 2 datasets corresponding to two seperate evaluations: one where the task is to predict fine-grained types and the other in which no type information is to be predicted. The training data consists of the *train_2015* and *dev_2015* data. The first dataset is annotated with 10 fine-grained NER categories such as person, geo-location, company, facility, product, music artist, movie, sports team, tv show and other. The second dataset is annotated without any type information, just only with B, I, O tags.

The training corpus consists of 2 394 tweets while the development corpora consist of 420 tweets. The testing data consists of 3 856 tweets (see Table 2). A total of 3 590 NEs are manually annotated in the corpora with 1 128 NEs in the training data and 2 462 NEs in the development data respectively (see Table 3).

| NE category | Training data | Development data |
|---|---|---|
| person | 266 | 664 |
| geo-location | 158 | 325 |
| company | 49 | 207 |
| facility | 77 | 209 |
| product | 158 | 177 |
| music artist | 76 | 116 |
| movie | 30 | 80 |
| sports team | 83 | 74 |
| tv show | 2 | 65 |
| other | 229 | 545 |
| **Total** | **1 128** | **2 462** |

Table 3. Statistics of the 10 fine-grained types of named entities in the training data and the development data provided by the 2nd shared task at WNUT-2016

### 3.2 Experimentations

In the preprocessing phase, we apply Twitter tokenization, POS tagging on tokenized data with TwitIE[3] (Bontcheva et al., 2013).

Once the final feature extraction has been completed, in the training phase, we make use of Conditional Random Fields (CRF) (Lafferty et al., 2001) as machine learning technique. We use the CRF implementation Wapiti[4], version 1.5.0 toolkit to create our model. The optimization algorithm is *l-bfgs (Limited-memory Broyden-Fletcher-Goldfarb-Shanno)*. During decoding phase, our model classifies, from a test corpus, whether a word should be labelled as named entities. Our model was evaluated on two tasks: (1) to predict 10 fine-grained types of named entities and (2) to predict no type of named entities.

We propose three templates with different features as follows:

[3] TwitIE: https://gate.ac.uk/wiki/twitie.html
[4] Wapiti 1.5.0: https://wapiti.limsi.fr

| Features | Template 1 (with unigrams) | Template 2 (with bigrams) | Template 3 (with trigrams) |
|---|---|---|---|
| (1) Orthographic features | x | x | x |
| (2) Lexical features | x | x | x |
| (3) Syntactic features | | x | x |
| (4) POS tags | x | x | x |
| (5) Polysemy count | | | x |
| (6) Longest n-gram length | | | x |

Table 4. Three CRF templates for the evaluations

The features for the tokens in the templates are in uni-grams (*Template 1*), bi-grams (*Template 2*), tri-grams (*Template 3*) and within a context window of size 3 (previous token, current token, next token) (see Table 4). We combine the *train* data, the *dev* data for training our model.

We use the metrics of precision, recall and weighted harmonic mean of precision and recall (F1) to evaluate the performance of the proposed system in two cases: (1) 10 types of named entities and (2) no-types of named entities.

## 4 Evaluations

The experiments are performed by training the model on all the features defined in section 2. We have trained, tested and evaluated iteratively the system in order to find out the best fitting feature sets. The results are presented in table 5 with models trained on the combination of training and development data with **2 814** tweets, then tested on the *dev_2015* with **1 000** tweets which are used in the baseline system (see README[5] file of WNUT-2016 workshop).

| | 10-types | | | No type | | |
|---|---|---|---|---|---|---|
| | **P** | **R** | **F1** | **P** | **R** | **F1** |
| **Baseline** (provided by WNUT-2016) | 40.34 | 32.22 | **35.83** | 54.21 | 49.62 | 51.82 |
| **Exp1 (template 1)** | 38.91 | 23.91 | 29.62 | 74.33 | 59.33 | 65.99 |
| **Exp2 (template 2)** | 40.36 | 23.82 | 29.96 | 74.33 | 60.00 | 66.40 |
| **Exp3 (template 3)** | 36.82 | 27.36 | 31.39 | 76.00 | 69.00 | **72.33** |

Table 5. Evaluations with model trained by applying tree templates with (train, dev, test) = (2 814, 0, 1 000) tweets

We realised that the performance of NER system was improved by applying the second template and the third template versus the first template with **F1 of 65.99%, 66.40%, 72.33%** for no-type of NEs and with **F1 of 29.62%, 29.96%, 31.39%** for 10-types of NEs respectively (see Table 5). We observed that the combination of all features with trigrams (*Template 3*) provided the best performance of NER model with **F1 of 72.33%** for no-type of NEs and with **F1 of 31.39%** for 10-types of NEs. While the experiments 1, 2, 3 give a better performance than the baseline with a gain of **+14.17%, +14.58%, +20.51% of F1**, respectively, in the evaluation of no-type of NEs, the experiments 1, 2, 3 give a performance less than the performance of the baseline, with a loss of **-6.21%, -5.87%, -4.44% of F1,** respectively, in the evaluation of 10-types of NEs.

Table 6 illustrated the effect of each features on the classifier when added to the baseline system which contains only the orthographic features and the POS tags features. We observed that not all features are equally useful in the classification and identification tasks. Some features get more

[5] README file of WNUT-2016 workshop: https://www.dropbox.com/s/yaoy7zi9vz71nki/wnut_ner_evaluation.tgz?dl=0

improvement in one context than in another, i.e. the syntactic features (**+1.72%** of F1) versus the lexical features **(+0.45%** of F1) for 10-types of NEs but the syntactic features **(+0.33%** of F1) versus the lexical features **(+1.00%** of F1) for no-types of NEs. And features can vary in effacity depending on the classification paradigm in which they are used. We noticed that each feature had separately a little bit improvement versus the baseline. The overall features get the significant enhancement upto **+5.03%** of F1 for 10-types of NEs and **+8.66%** of F1 for no-types of NEs.

| | **10-types** | | | **No type** | | |
|---|---|---|---|---|---|---|
| | **P** | **R** | **F1** | **P** | **R** | **F1** |
| **Baseline : orthographic + POS tags features** | 42.64 | 22.91 | 26.36 | 73.67 | 57.67 | 63.67 |
| **+ lexical features** | 40.73 | 23.73 | 26.82 *(+0.45)* | 74.33 | 59.33 | 64.67 *(+1.00)* |
| **+ syntactic features** | 43.00 | 24.18 | 27.64 *(+1.27)* | 73.00 | 58.67 | 64.00 *(+0.33)* |
| **+ the polysemy count** | 44.45 | 23.55 | 27.18 *(+0.82)* | 74.33 | 58.67 | 64.33 *(+0.67)* |
| **+ the longest n-gram length** | 42.18 | 23.55 | 26.91 *(+0.55)* | 74.33 | 58.67 | 64.33 *(+0.67)* |
| **All features** | 36.82 | 27.36 | **31.39** *(+5.03)* | 76.00 | 69.00 | **72.33** *(+8.66)* |

Table 6. Effect of each features on the classifier when added to the baseline system with (*train, dev, test*) = (*2 814, 0, 1 000*) tweets

Moreover, we observed that the language model of WMT-2016 was trained on newswire and actualities information. So the data are similar to the contents inside Twitter messages. Indeed, the new feature of polysemy count allows to disambiguate a word by checking the number of meanings of this word in a Twitter message in an encyclopedic dictionary.

This is one reason why we have submitted our model trained with the template 3 for the WNUT-2016 workshop. The results are showed in the table 7. We have performed two experiments with different datasets as follows:

- Experiment 1: *(train, dev, test) = (2 814, 0, 3 856)* tweets,
- Experiment 2: *(train, dev, test) = (3 534, 280, 3 856)* tweets, where *train* set is composed by *train_2016* (*2 394* tweets), *dev_2016* (*1 000* tweets) and some of *new_dev_2016* (*140* tweets).

We have purposely experimented with different size of the training set and the development set, but the same size of the testing set in order to examine the performance of our model. We observed that the variation of size in the training set and the development set gives an impact on our model performance. We noticed that the model trained in the experiment 2 performed globally better for the both evaluations than the model trained in the experiment 1. The experiment 2 gains **+5.49% of F1** more than the experiment 1 about the evaluation of 10-types of NEs, but slightly **+1.96% of F1** more than the experiment 1 about the evaluation of no-types of NEs. We realised that the larger the size of in the training set and the development set is, the better the performance of our model.

Official evaluation, with our model trained in the experiment 2 (submitted), have shown **F1 of 29.82%** for the 10-types fine grained named entities types and **F1 of 44.30%** for the no-type of named entities, as explained in Table 7.

| | 10-types | | | No type | | |
|---|---|---|---|---|---|---|
| | **P** | **R** | **F1** | **P** | **R** | **F1** |
| **Experiment 1** | 34.73 | 26.36 | 29.97 | 76.33 | 67.67 | 71.74 |
| **Experiment 2** (submitted) | 40.73 | 23.52 | 29.82 | 53.21 | 37.95 | 44.30 |
| **Experiment 2** | 40.90 | 31.30 | **35.46** | 77.00 | 70.67 | **73.70** |

Table 7. Evaluations with model trained with template 3,
Experiment 1: (train, dev, test) = (2 814, 0, 3 856) tweets,
Experiment 2: (train, dev, test) = (3 534, 280, 3 856) tweets

## 5 Conclusion

In this paper, we have presented our work in WNUT-2016 – The 2nd shared task of named entities recognition in Twitter. We have proposed a set of features which improves the NER system performance. We have considered various combinations of features. Our system shows the official evaluation results of **29.82% (F1)** for the 10 fine-grained types of named entities and **44.30% (F1)** for the no-type of named entities.

Further work will focus on adding more domain-specific features and additional features such as word embeddings, to improve the accuracy of the system. In addition, we would like to investigate neural network architectures such as bidirectional LSTM, that have shown great promise in many NLP tasks, for named entities recognition and co-reference resolution.

## Reference


David Nadeau And Satoshi Sekine. 2007. *A Survey of Named Entity Recognition and Classification*. Lingvisticae Investigationes 30 (1):3-26.

Timothy Baldwin, Marie-Catherine de Marneffe, Bo Han, Young-Bum Kim, Alan Ritter and Wei Xu. 2015. *Shared Tasks of the 2015 Workshop on Noisy User-generated Text: Twitter Lexical Normalization and Named Entity Recognition.* Proceedings of the ACL 2015 Workshop on Noisy User-generated Text, Beijing, China, July 31, 2015:126–135.

J. Lafferty, A. McCallum, and F. Pereira. 2001. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data.* In Proceedings of 18th International Conference on Machine Learning. Morgan Kaufmann, San Francisco, Californie, Etats-Unis d'Amerique, 2001: 282–289.

K. Bontcheva, L. Derczynski, A. Funk, M.A. Greenwood, D. Maynard and N. Aswani. 2013. *TwitIE: An Open-Source Information Extraction Pipeline for Microblog Text*. In Proceedings of the International Conference on Recent Advances in Natural Language Processing, ACL.

R. Navigli and S. P. Ponzetto. 2012. *BabelNet: The Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network.* Artificial Intelligence, 2012 (193):217–250.

S. Petrov and D. Klein. 2007. *Improved Inference for Unlexicalized Parsing.* in HLT-NAACL, 2007.

Christophe Servan, Ngoc-Tien Le, Ngoc Quang Luong, Benjamin Lecouteux, Laurent Besacier. 2015. *An Open Source Toolkit for Word-level Confidence Estimation in Machine Translation.* The 12th International Workshop on Spoken Language Translation (IWSLT'15), Dec 2015, Da Nang, Vietnam.

Sadat, F., Kazemi, F., Farzindar, A. 2014a. *Automatic identification of arabic language varieties and dialects in social media.* SocialNLP 2014, 22.

Sadat, F., Mallek, F., Sellami, R., Boudabous, M. M., Farzindar, A. 2014b. *Collaboratively constructed linguistic resources for language variants and their exploitation in nlp applications–the case of tunisian arabic and the social media.* In Proceedings of the Workshop on lexical and grammatical resources for language processing, The 25th International Conference on Computational Linguistics, COLING 2014, August 2014, Dublin, Ireland, 102-110.

Sadat Fatiha. 2013. *Arabic social media analysis for the construction and the enrichment of NLP tools.* In Corpus Linguistics 2013. Lancaster University, UK. Jul. 22-26, 2013.