Feature-Rich Twitter Named Entity Recognition and Classification

Utpal Kumar Sikdar and **Björn Gambäck**
Department of Computer and Information Science
Norwegian University of Science and Technology
Trondheim, Norway
{sikdar.utpal, gamback}@idi.ntnu.no

Abstract

Twitter named entity recognition is the process of identifying proper names and classifying them into some predefined labels/categories. The paper introduces a Twitter named entity system using a supervised machine learning approach, namely Conditional Random Fields. A large set of different features was developed and the system was trained using these. The Twitter named entity task can be divided into two parts: i) Named entity extraction from tweets and ii) Twitter name classification into ten different types. For Twitter named entity recognition on unseen test data, our system obtained the second highest F_1 score in the shared task: 63.22%. The system performance on the classification task was worse, with an F_1 measure of 40.06% on unseen test data, which was the fourth best of the ten systems participating in the shared task.

1 Introduction

Social media such as Twitter has become an everyday part of many people's lives, and play a major role in modern society. There are currently 1.65 billion monthly active users on Facebook (Facebook, 2016) and 310 million on Twitter (Twitter, 2016). The number of people involved leads to a vast amount of data being shared, with almost 500 million short messages, *tweets*, being produced daily on Twitter (Internet Live Stats, 2016). Due to the ease and spontaneity of the message creation, and the restriction of a maximum length of 140 characters on the tweets, the language used on Twitter is often very noisy, with tweets containing many grammatical and spelling mistakes, short form of the words, multiple words merged together, special symbols and characters inserted into the words, etc. Hence it is difficult to analyse and monitor all types of tweets, and to sift through the vast number of tweets: specific tweets may need to be filtered out from millions of tweets. Named entity extraction plays a vital role when filtering out relevant tweets from a collection. It is also useful as a pre-processing step in many other language processing tasks, such as machine translation and question-answering.

Several approaches to Twitter named entity extraction have already been tried, but it is still a challenging task due to noisiness of the texts. Liu et al. (2011) proposed a semi-supervised learning framework to identify Twitter names. They used a k-Nearest Neighbors (kNN) approach to label the Twitter names and gave these labels as an input feature to a Conditional Random Fields (CRF) classifier, achieving almost 80% accuracy on their own annotated data. A supervised model was proposed for Twitter named entity recognition by Ritter et al. (2011) who applied Labeled LDA (Ramage et al., 2009) to recognize the possible types of Twitter names, and also showed that part-of-speech and chunk information are important components in Twitter named identification. Li et al. (2012) introduced an unsupervised Twitter named entity extraction strategy based on dynamic programming.

The present work addresses the second version of a Twitter named entity shared task that first was organized at the ACL 2015 workshop on noisy user-generated text (Baldwin et al., 2015), with two subtasks: Twitter named entity identification and classification of those named entities into ten different types. Of the eight systems participating in the first workshop, the best (Yamada et al., 2015) achieved an F_1 score of 70.63% for Twitter name identification and 56.41% for classification, by combining supervised machine learning with high quality knowledge obtained from several open knowledge bases such as Wikipedia. Another team, (Akhtar et al., 2015) used a strategy based on differential evolution, getting F_1 scores of 56.81% for Twitter name identification and 39.84% for the classification task.

Proceedings of the 2nd Workshop on Noisy User-generated Text,
pages 164–170, Osaka, Japan, December 11 2016.

A related shared task on Twitter named entity recognition and linking to DBpedia was held in conjunction with the 2016 WWW conference at #Microposts2016 (Cano et al., 2016). Five teams submitted their systems to the workshop, with the best (Waitelonis and Sack, 2016) achieving recall, precision and F-measure values of 49.4%, 45.3% and 47.3%. In that system, each token is mapped to gazetteers developed from the DBpedia database. Tokens are discarded if they match stop words or are not nouns.

The present paper outlines a supervised machine learning approach to Twitter named entity identification and classification. A Conditional Random Field (Lafferty et al., 2001) classifier was trained on a rich set of features and used for identification of Twitter names from the tweets, giving an F_1 score of 63.22%. In order to classify the named entities into ten different types, we developed two models and combined their output, achieving an F-measure of 40.06%. The rest of the paper is organized as follows: The Twitter name identification methodology and the different features used are introduced in Section 2. Results are presented and discussed in Section 3, while Section 4 addresses future work and concludes.

2 Twitter Named Entity Recognition

The Twitter Named Entity Recognition shared task (Strauss et al., 2016) at W-NUT 2016, the COLING workshop on noisy user-generated text, was divided into two subtasks: ‘notypes’ and ‘10types’. In the ‘notypes’ subtask, named entities should just be identified in the tweets, while in the ‘10types’ subtask the aim was to identify and classify Twitter names into ten different categories: *facility, geo-loc, movie, musicartist, person, company, product, sportsteam, tvshow*, and *other*. We hence first identify the Twitter names from the tweets, and then in a second step classify the names according to the ten given labels.

2.1 Twitter Named Entity Identification for ‘notypes’

The Twitter named entities were first extracted from the tweets using a supervised machine learning approach, namely Conditional Random Fields, CRFs (Lafferty et al., 2001). We used the C^{++} based CRF^{++} package,[1] a simple, customizable, and open source implementation of CRF for segmenting or labelling sequential data. The ‘notypes’ system is shown at the top of Figure 1.

A range of different features were developed for identifying the Twitter names. These features are described below. Note that some of the feature types are implemented as several different features and that the first five types mainly are lexicon-based, while the last five types primarily relate to the context. The eight feature types in the middle of the list are predominantly word internal and character-based. The contributions of these three different groups of feature types will be compared to each other in the experimental section (Section 3).

Lexicon-based features

Lexical data: This binary feature was extracted from the lexical data supplied by the shared task organisers. The feature is set to 1 if the current word belongs to the lexical data, otherwise 0.

Babelfy named entities: Each tweet was passed to the Babelfy (Moro et al., 2014) named entity recognition system for recognizing Twitter names. If the current word belongs to the Babelfy named entities, this binary feature is set.

Part-of-speech (POS): The TweeboParser[2] was used for generating part-of-speech tags for each token in the tweets and the POS tag of the current word was used as a feature. The POS tags of the previous two tokens and following two tokens were also used as features, so in total there are five POS tag features for each token.

Stop word match: All tokens are checked against a stop word list collected from the web.[3] The binary feature is set if the current token matches one of these stop words.

Word frequency: Less frequent words were found to often belong to named entities. If the pre-calculated frequency from the training data of the current word is less than a certain threshold, this binary feature is set.

[1] http://crfpp.sourceforge.net

[2] http://www.cs.cmu.edu/~ark/TweetNLP/

[3] http://www.ranks.nl/stopwords

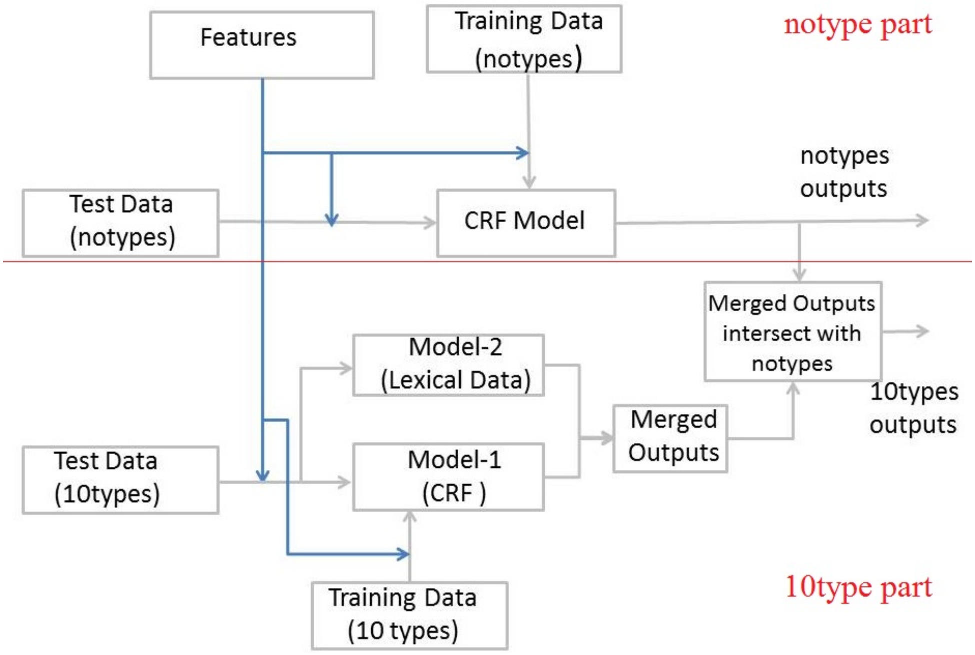

Figure 1: Twitter Named Entity Recognition system(s)

Character-based features

Numeric: Many Twitter names were found to contain numeric characters, so this binary feature shows whether the current token contains numeric characters or not.

Initial capital: Proper nouns in general tend to start with capital letters, so this feature flags if the current word has an initial capital letter.

Inner capital: This feature is set if the token contains any capital letter in a word-internal position.

Word normalisation: The word mapped to its equivalent class: each capital letter in the token is mapped to 'A', all lower-case letters to 'a', and digits to '0'. Other characters are kept unaltered.

Special character followed by token: Many Twitter names follow certain special characters (e.g., '@' and '#'). The feature checks if the token is following any special character or not.

Word length: If the current word's length is greater than some threshold, this binary feature is set.

Word suffix and prefix: A fixed maximum number of characters are stripped from the beginning and the end of the current word, with the remainder parts being used as two features.

Context-based features

Local context: Local contexts play an important role in identifying Twitter names. Here we used the three previous and the three next words as local contexts (so there are six context features).

Chunk information: Chunk information was collected using annotated chunk data from the OSU Twitter NLP Tools.[4] A CRF-based chunk model was developed using prefix, suffix and part-of-speech information. Each tweet was passed to the chunk model and chunk labels generated for that tweet. This information was then used as a feature for the current token. In addition, the chunk information of the previous token and the following token were also used as features.

First word: This binary feature checks if the current token is at the beginning of a sentence or not.

Last word: This binary feature is set if the current token is the last word of a sentence, without considering sentence ending symbols and punctuation marks (i.e., '.', '?', '!', etc.).

Previous label: Finally, the previous token's calculated named entity label was used as a feature.

[4] https://github.com/aritter/twitter_nlp/blob/master/data/annotated/chunk.txt

Dataset	Tweets	Named Entities
Training	2,814	1,768
Development	1,000	661
Test	3,850	3,473

Table 1: Number of tweets and named entities in the W-NUT 2016 datasets

For the actual named entity system, some of the limits used for the features were set based on testing on the training data. Hence, the thresholds for the word frequency and word length features were empirically set to ≥ 10 and ≥ 5, respectively, while the prefix and suffix strip lengths were fixed to four characters.

2.2 Twitter Named Entity Classification for '10types'

For the '10types' subtask, the goal was to identify and classify Twitter names into ten given categories. Two models were developed for the Twitter name classification, as follows:

Model-1: In Model-1, the tokens are classified into ten categories using CRF. The same features were utilised as described in Section 2.1 above.

Model-2: This model was based on the lexical data given by the shared task organisers (Strauss et al., 2016): the Twitter named entity classes are categorized into 10types based on the supplied file '`dictionaries.conf`'. In addition, Twitter names were extracted from the training data and merged with the lexical data for each category. All tokens were passed to the lexical data and classified into the ten categories.

The output of these two models was first merged and the merged output was later checked as to whether it belonged to 'notypes' or not. When merging the output from the models, highest priority was given to Model-2, if the two models generated different named entity classes for a particular token. For example, 'Washington Navy Yard' is recognized as belonging to the *other* category by the Model-1, but Model-2 recognizes it as *facility*. So the category actually assigned to this entity by the overall system will be *facility*, as suggested by Model-2. If an entity matches more than one class in Model-2, we randomly assign the class of the entity among the matched classes.

Next, the merged output was compared to the 'notypes' entities. If the merged output belongs to those 'notypes' entities (fully matched), the output entity is considered to be a Twitter name, otherwise it is discarded. For example, the entity 'Nobu Restaurant' is classified as *other*, but this entity is not identified by 'notypes', so it is not considered as a Twitter named entity. The '10types' system is outlined in Figure 1, and consists of processes shown mainly in the lower half of the figure, but thus also utilises the 'notype' subsystem (the upper part of the figure).

3 Experiments and Discussion

A range of experiments were conducted based on the datasets provided by the shared task organizers (Strauss et al., 2016). The statistics of the datasets are given in Table 1.

3.1 The 'notypes' Named Entity Identification Task

For the 'notypes' task, the system was developed on the training data and evaluated on the development data. The 'notypes' model was initially developed using all the training data, but that gave low recall in relation to the precision. To increase the recall performance, the tweets that did not contain any named entities were removed from the training data, and a new model was built using all the features described in Section 2. This system achieved recall, precision and F-measure values of 68.68%, 65.14% and 66.86%, respectively. This final F-measure value represents an increase by almost 7% compared to the model which was built using all the training data, as can be seen in Table 2.

Training Data	Recall	Precision	F_1
all training data	54.01	67.74	60.10
removing tweets without Twitter names	68.68	65.14	66.86

Table 2: Performance on the development data for Twitter Named Entity Identification ('notypes')

Features	Recall	Precision	F_1
Lexical	57.19	49.48	53.05
Word internal	44.18	49.32	46.61
Context	16.79	57.81	26.03

Table 3: Feature class contributions to Twitter Named Entity Identification ('notypes' development data)

Team	Accuracy	Recall	Precision	F_1
CambridgeLTL	95.34	73.49	59.72	65.89
NTNU	94.38	64.18	62.28	63.22
Talos	94.54	70.53	52.58	60.24
akora	94.51	64.75	54.28	59.05
ASU	94.08	57.55	52.98	55.17

Table 4: Comparison of results (top five systems) for 'notypes' on the unseen test data

Table 3 looks at the feasibility of the different groups of features, that is, how much each feature group actually contribute in isolation. In the table, the features are sorted into three different classes: 'lexical', 'word internal' and 'context'. Referring to Section 2, the 'lexical' group consists of the features lexical data, babelfy, POS, stopword and word frequency. The 'word internal' (character-based) feature group contains the alphanumeric, initial/inner capital, normalisation, special character, word length and pre-/suffix features, while the 'context' group is made up of the context window words features, chunk, first/last words, and the previous token's label.

As can be seen in Table 3, the context features do not contribute much at all to improving the recall, but are the most helpful features for improving the precision. The lexical and word internal features contribute roughly equally to the precision score, but the lexical features very clearly are the most useful for achieving good recall.

The full-feature set 'notypes' model built only on the tweets with named entities was entered in the shared task as the named entity extraction system 'NTNU'. When applied to the unseen test data it achieved state-of-the-art results by obtaining the second highest score in the task. The comparable test data results (top five of the ten participating systems) are reported in Table 4. The NTNU system achieved recall, precision and F_1 values of respectively 64.18%, 62.28% and 63.22%, and thus scored over 2 points less on F-measure than the best performing system, but on the other hand scored 3 points more than the third ranked system.

3.2 The '10types' Named Entity Classification Task

For the '10types' classification task, we developed the two models described in Section 2.2 and also merged their output. The results are shown in Table 5. When using all the feature classes (lexical, word internal and context), Model-1 produced the best recall (28.74%) with an F-measure of 37.40%. Model-2 generated a better precision (85.16%), but due to bad recall still had a lower F_1: 27.63%. Combining the two models boosted the results on the development data on all measures, giving a 43.81% F-score.

Table 6 indicates the contribution from each of the three types of feature sets to the performance of Model-1. Just as for the identification task, it is also for classification clear that the context features are most helpful for boosting precision, while the lexical features help the recall most.

Classification Model	**Recall**	**Precision**	**F$_1$**
Model-1	28.74	53.52	37.40
Model-2	16.49	85.16	27.63
Combined Model	35.10	58.29	43.81

Table 5: Results for Twitter Named Entity Classification (‘10types’) on the development data

Features	**Recall**	**Precision**	**F$_1$**
Lexical	22.39	34.58	27.18
Word internal	15.13	49.50	23.17
Context	6.20	95.35	11.65

Table 6: Feature contributions to Model-1 for Named Entity Classification (‘10types’ development data)

Team	**Test Data**			
	Accuracy	Recall	Precision	F$_1$
CambridgeLTL	93.52	60.77	46.07	52.41
Talos	92.95	58.51	38.12	46.16
akora	92.73	51.70	39.48	44.77
NTNU	92.48	53.19	32.13	40.06
ASU	92.42	40.58	37.58	39.02

Table 7: Comparison of results (top five systems) for ‘10types’ on the unseen test data

Applying the NTNU ‘10types’ model to the previously unseen test data, it achieved recall, precision and F-measure values of 53.19%, 32.13% and 40.06%, respectively. The system came in fourth place in this subtask and the results of the top five systems are shown in Table 7. One of the main reasons why our system is outperformed by the top systems is that not all the named entities identified by the system were classified into any of the ten categories.

3.3 Error Analysis

The outputs were analysed in order to understand the nature of the errors encountered. In the ‘notypes’ Twitter named entity extraction subtask, several named entities were not identified by the system, while many tokens were wrongly identified as names by the system. The probable reason for this is that the system could not find enough relevant examples in the training data.

In the ‘10types’ classification task, very few of the entities were identified and classified into the ten categories. In the development data, only 398 entities were identified out of the 661 named entities that actually were in the data, and out of those 398, only 232 entities were correctly identified. Again, this may be due to insufficient number of relevant training instances in the training data.

4 Conclusion

This paper has proposed a system for Twitter named entity identification and classification. A range of different features were developed to extract Twitter names from the tweets. Two systems were built, one for the ‘notypes’ named entity extraction task and the other for the ‘10types’ classification task. The systems were built around a CRF-based classifier and lexical data, and both systems achieved state-of-the-art results. In the future, we will analyse the errors in more detail and aim to use external resources (e.g., DBpedia and Wikipedia) to reduce the misclassification of the tokens, as well as to identify more entities from the tweets. We will also try to generate more models and later ensemble these model to improve the system performance.

References

Md Shad Akhtar, Utpal Kumar Sikdar, and Asif Ekbal. 2015. IITP: Multiobjective differential evolution based Twitter named entity recognition. In Wei Xu, Bo Han, and Alan Ritter, editors, *Proceedings of the ACL 2015 Workshop on Noisy User-generated Text*, pages 106–110, Stroudsburg, PA, USA, July. Association for Computational Linguistics.

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. In Wei Xu, Bo Han, and Alan Ritter, editors, *Proceedings of the ACL 2015 Workshop on Noisy User-generated Text*, pages 126–135, Stroudsburg, PA, USA, July. Association for Computational Linguistics.

Amparo E. Cano, Daniel Preoţiuc-Pietro, Danica Radovanović, Katrin Weller, and Aba-Sah Dadzie. 2016. #Microposts2016 — 6th workshop on 'making sense of microposts'. In *Proceedings of the 25th World Wide Web Conference (WWW'16)*, volume Companion, pages 1041–1042, Montréal, Canada, April. Association for Computing Machinery.

Facebook. 2016. Company Information. `http://newsroom.fb.com/company-info/`.

Internet Live Stats. 2016. Twitter Usage Statistics.
`http://www.internetlivestats.com/twitter-statistics`.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann.

Chenliang Li, Jianshu Weng, Qi He, Yuxia Yao, Anwitaman Datta, Aixin Sun, and Bu-Sung Lee. 2012. TwiNER: Named entity recognition in targeted Twitter stream. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 721–730, New York, NY, USA. Association for Computing Machinery.

Xiaohua Liu, Shaodian Zhang, Furu Wei, and Ming Zhou. 2011. Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 359–367, Stroudsburg, PA, USA. Association for Computational Linguistics.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, volume 1, pages 248–256, Stroudsburg, PA, USA. Association for Computational Linguistics.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Stroudsburg, PA, USA. Association for Computational Linguistics.

Benjamin Strauss, Bethany E. Toma, Alan Ritter, Marie Catherine de Marneffe, and Wei Xu. 2016. Results of the WNUT16 named entity recognition shared task. In Bo Han, Alan Ritter, Leon Derczynski, Wei Xu, and Timothy Baldwin, editors, *Proceedings of the Workshop on Noisy User-generated Text (WNUT 2016)*, Stroudsburg, PA, USA, December. Association for Computational Linguistics.

Twitter. 2016. Company Information. `https://about.twitter.com/company`.

Jörg Waitelonis and Harald Sack. 2016. Named entity linking in #tweets with KEA. In Daniel Preoţiuc-Pietro, Danica Radovanović, Amparo E. Cano-Basave, Katrin Weller, and Aba-Sah Dadzie, editors, *Proceedings of 6th Workshop on Making Sense of Microposts (#Microposts2016)*, pages 61–63, Montréal, Canada, April. Sun SITE Central Europe (CEUR) Workshop Proceedings.

Ikuya Yamada, Hideaki Takeda, and Yoshiyasu Takefuji. 2015. Enhancing named entity recognition in Twitter messages using entity linking. In Wei Xu, Bo Han, and Alan Ritter, editors, *Proceedings of the ACL 2015 Workshop on Noisy User-generated Text*, pages 136–140, Stroudsburg, PA, USA, July. Association for Computational Linguistics.