# Shared Tasks of the 2015 Workshop on Noisy User-generated Text: Twitter Lexical Normalization and Named Entity Recognition

**Timothy Baldwin**
University of Melbourne
tb@ldwin.net

**Marie Catherine de Marneffe**
The Ohio State University
demarneffe.1@osu.edu

**Bo Han**
IBM Research
bohan.ibm@au1.ibm.com

**Young-Bum Kim**
University of Wisconsin
ybkim@cs.wisc.edu

**Alan Ritter**
The Ohio State University
ritter.1492@osu.edu

**Wei Xu**
University of Pennsylvania
xwe@cis.upenn.edu

## Abstract

This paper presents the results of the two shared tasks associated with W-NUT 2015: (1) a text normalization task with 10 participants; and (2) a named entity tagging task with 8 participants. We outline the task, annotation process and dataset statistics, and provide a high-level overview of the participating systems for each shared task.

## 1 Introduction

As part of the 2015 ACL-IJCNLP Workshop on Noisy User-generated Text (W-NUT), we organized two shared tasks: (1) a text normalization task (Section 2); and (2) a named entity tagging task (Section 3).

In the text normalization task, participants were asked to convert non-standard words to their standard forms for English tweets. Participating systems were classified by their use of resources, into a constrained and an unconstrained category: constrained systems were permitted to use only the provided training data and off-the-shelf tools; unconstrained systems, on the other hand, were free to use any public tools and resources. There were 6 official submissions in the constrained category, and 5 official submissions in the unconstrained category. Overall, deep learning methods and methods based on lexicon-augmented conditional random fields (CRFs) achieved the best results. The winning team achieved a precision of 0.9061 precision, recall of 0.7865, and F1 of 0.8421.

The named entity recognition task attracted 8 participants. The majority of teams built their systems using linear-chain conditional random fields (Lafferty et al., 2001), and many teams also used brown clusters and word embedding features (Turian et al., 2010). Notable new techniques for named entity recognition in Twitter include a semi-Markov MIRA trained tagger (nrc), an end-to-end neural network using no hand-engineered features (multimedialab), an approach that weights training data to compensate for concept drift (USFD), and a differential evolution approach to feature selection (iitp). The submission from the winning team (ousia) achieved suprisingly good performance on this difficult task, near the level of inter-rater agreement.

## 2 Text Normalization Shared Task

In this section, we outline the Twitter Text Normalization Shared Task, describing the data and annotation process, and outlining the approaches adopted by participants.

### 2.1 Background

Non-standard words are present in many text genres, including advertisements, professional forums, and SMS messages. They can be the cause of reading and understanding problems for humans, and degrade the accuracy of text processing tools (Han et al., 2013; Plank et al., 2014a; Kong et al., 2014). Text normalization aims to transform non-standard words to their canonical forms (Sproat et al., 2001; Han and Baldwin, 2011) as shown in Figure 1. Common examples of non-standard words include abbreviations (e.g., *u* "you"), and non-standard spellings (e.g., *cuming* "coming" or *2mr* "tomorrow"). The prevalence of non-standard words in social media text results in markedly higher out-of-vocabulary (OOV) rates; normalizing the text brings OOV rates down to more conventional levels and makes the text more amenable to automatic processing with off-the-shelf tools which have been trained on edited text.

Text normalization over Twitter data has been addressed at different granularities. For instance, non-standard words can be considered as spelling errors at the character (Liu et al., 2011) or word level (Wang and Ng, 2013). Text normalization can also be approached as a machine



*Proceedings of the ACL 2015 Workshop on Noisy User-generated Text*, pages 126–135,
Beijing, China, July 31, 2015. 

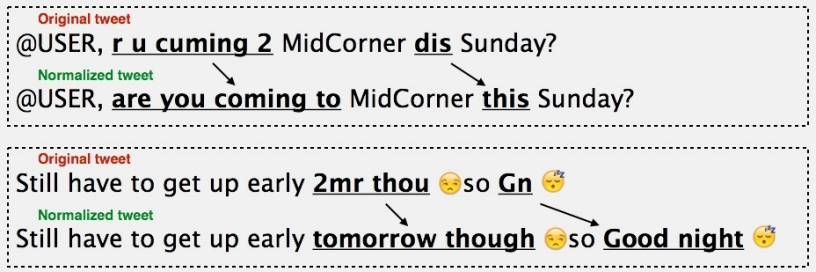


Figure 1: Normalization examples

translation task, whereby non-standard words are mapped to more canonical expressions (Aw et al., 2006). Other approaches have involved deep learning (Chrupała, 2014), cognitively-inspired approaches (Liu et al., 2012), random walks (Hassan and Menezes, 2013), and supervision using automatically-mined parallel data (Ling et al., 2013).

One major challenge in text normalization research has been the lack of annotated data for training and evaluating methods. As a result, most Twitter text normalization methods have been unsupervised or semi-supervised (Cook and Stevenson, 2009; Han et al., 2012; Yang and Eisenstein, 2013), and evaluated over small-scale hand-annotated datasets. This has hampered analysis of the strengths and weaknesses of individual methods, and was our motivation in organizing the lexical normalization shared task.

### 2.2 Shared Task Design

This lexical normalization shared task is focused exclusively on English, and was designed with three primary desiderata in mind: (1) to construct a much larger dataset than existing resources; (2) to allow all of 1:1, 1:$N$ and $N$:1 word $n$-gramm appings; and (3) to cover not just OOV non-standard words but also non-standard words that happen to coincide in spelling with standard words. In all three regards, the shared task expands upon the scope of the de facto evaluation datasets of Han and Baldwin (2011) and Liu et al. (2011).

One constraint that was placed on candidate tokens for normalization was that they should be all-alphanumeric. For normalization, we adopted American spelling.

In order to establish a more level playing field for participants, but also encourage the use of a wide range of resources, participants were required to nominate their system categories:

- **Constrained:** participants could not use any data other than the provided training data to perform the text normalization task. They were allowed to use pre-trained tools (e.g., Twitter POS taggers), but no normalization lexicons or extra tweet data.

- **Unconstrained:** participants could use any publicly accessible data or tools to perform the text normalization task.

Evaluation was based on token-level precision, recall and F-score.

#### 2.2.1 Preprocessing

We first collected tweets using the Twitter Streaming API over the period 23–29 May, 2014, and then used `langid.py` (Lui and Baldwin, 2012)[1] to remove all non-English tweets. Tokenization was performed with `CMU-ARK tokeniser`.[2] To ensure that tweets had a high likelihood of requiring lexical normalization, we filtered out tweets with less than 2 non-standard words (i.e. words not occurring in our dictionary — see Section 2.2.3). While this biases the sample of tweets, the decision was made at a pragmatic level to ensure a reasonable level of lexical normalization and "annotation density". This was based on a pilot study over a random sample of English tweets, in which we found that many non-standard words were actually unknown named entities which did not require normalization. In all, 5,200 randomly-sampled English tweets were annotated for the shared task dataset.

#### 2.2.2 Annotation

12 interns and employees at IBM Research Australia were involved in the data annotation. All

[1] `https://github.com/saffsd/langid.py`
[2] `https://github.com/myleott/ark-twokenize-py`

annotators had a high level of English proficiency (IELTS $\geq$ 6.0) and were reasonably familiar with Twitter data. Each annotator labeled at least 200 tweets, and each tweet was independently labeled by two annotators based on the annotation guidelines.[3] As part of this, any non-English tweets misclassified by `langid.py` were manually removed from the dataset. This resulted in the final size of the annotated dataset dropping to 4,917 tweets. All annotations were completed within two weeks, and achieved an average Cohen's $\kappa$ of 0.5854.

For all instances of annotator disagreement, an annotator who was not involved in the first-pass annotation process was asked to adjudicate in the following week. During the course of the shared task, we additionally examined and incorporated a small number of annotation corrections reported by participants.

#### 2.2.3 English Lexicon

It is impossible to reach consensus on the dividing line between standard words and non-standard words (e.g. are *footie*, *y'all* and *youse* non-standard or standard words?). We artificially arrive at such a dividing line via membership in a prescribed lexicon of English. Specifically, we use the `SCOWL` database with American spellings as the default English lexicon.[4] The `SCOWL` database integrates words from multiple sources and also contains valid word spelling variations, which makes it an excellent English lexicon for this shared task. As suggested in the database guidelines, we used a dictionary size of 70%, such that the lexicon contains words found in most dictionaries, but also many high-frequency proper nouns such as *Obama* and *Facebook*.

The overall English lexicon (after de-duplication) contains 165,458 words. This lexicon was used: (a) to pre-filter data, i.e., tweets with less than two tokens not in this lexicon are dropped from our annotations; and (b) as the basis of the standard words for normalization.

#### 2.2.4 Dataset Statistics

The dataset was randomly split 60:40, into 2,950 tweets for the training data and 1,967 tweets for the test data. Table 1 details the number of (possibly multi-word) tokens in each of the training and test data that were normalized based on a 1:1, 1:$N$ or $N$:1 mapping. We additionally include the proportion of tokens in each category that were contained in the test data, to confirm that the dataset is relatively balanced in composition between the training and test partitions.

| Category | 1:1 | 1:$N$ | $N$:1 | Overall |
|---|---|---|---|---|
| Training | 2,875 | 1,043 | 10 | 3,928 |
| Test | 2,024 | 704 | 10 | 2,738 |
| Training ratio | 0.587 | 0.597 | 0.500 | 0.589 |

Table 1: Numbers of non-standard words in the training and test datasets for the lexical normalization task, broken down into 1:1, 1:$N$ and $N$:1 mappings from non-standard words to standard words. "Training ratio" represents the number of non-standard words in the training data divided by the overall non-standard words in that category.

| Rank | Training | | Test | | Combined | |
|---|---|---|---|---|---|---|
| 1 | u | 333 | u | 236 | u | 569 |
| 2 | lol | 272 | lol | 197 | lol | 469 |
| 3 | im | 182 | im | 154 | im | 336 |
| 4 | dont | 92 | nigga | 60 | dont | 149 |
| 5 | omg | 67 | dont | 57 | nigga | 117 |
| 6 | nigga | 57 | lmao | 45 | omg | 101 |
| 7 | niggas | 52 | n | 43 | lmao | 96 |
| 8 | lmao | 51 | niggas | 42 | niggas | 94 |
| 9 | n | 49 | omg | 34 | n | 92 |
| 10 | ur | 46 | ur | 28 | ur | 74 |

Table 2: Top-10 most frequent non-standard words in each partition of the lexical normalization dataset.

Overall, 373 non-standard word types were found in the intersection of the training and test data. The number of non-standard word types unique to the training and test partitions was 777 and 488, respectively. We further show the top-10 most frequent non-standard words and their token frequencies in the training, test and combined datasets in Table 2. Despite the large number of unique non-standard word in the training and test partitions, there is relatively strong agreement in the high-frequency non-standard words across the dataset partitions.

[3] `http://noisy-text.github.io/files/annotation_guideline_v1.1.pdf`

[4] Version 2014.11.17 was used.

### 2.3 Normalization Approaches and Discussion

Overall, 10 teams submitted official runs to the shared task: 6 teams participated in the constrained category, 5 teams in the unconstrained category, and 1 team in both categories.[5] The normalization results for each category are shown in Tables 3 and 4. Overall, common approaches were lexicon-based methods, CRFs, and neural network-based approaches. Among the constrained systems, neural networks achieved strong results, even without off-the-shelf tools. In contrast, CRF- and lexicon-based approaches were shown to be effective in the unconstrained category. Surprisingly, the best overall result was achieved by a constrained system, suggesting that the relative advantage in accessing additional datasets or resources has less impact than the quality of the underlying model that is used to model the task.

**NCSU_SAS_NING (Jin, 2015)** Normalization candidates were generated based on the training data, and scored based on Jaccard index over character *n*-gram[ s]. Candidates were evaluated using random forest classifiers to offset parameter sensitivity, using features including normalization statistics, string similarity and POS.

**NCSU_SAS_WOOKHEE (Min et al., 2015)** Word-level edits are predicted based on long-short term memory (LSTM) recurrent neural networks (RNN), using character sequences and POS tags as features. The LSTM is further complemented with a normalization lexicon induced from the training data.

**NCSU_SAS_SAM (Leeman-Munk et al., 2015)** Two forward feed neural networks are used to predict: (1) the normalized token given an input token; and (2) whether a word should be normalized or left intact. Normalized tokens are further edited by a "conformer" which down-weights rare words as normalization candidates.

**IITP (Akhtar et al., 2015b)** A CRF model is trained over the training data, with features including word sequences, POS tags and morphology features. Post-processing heuristics are used to post-edit the output of the CRF.

[5]One team (GIGO) didn't submit a description paper.

**DCU-ADAPT (Wagner and Foster, 2015)** A generalized perceptron method is used generate word edit operations, with features including character *n*-gram[ s], character classes, and RNN language model hidden layer activation features. The final normalization word is selected based on the noisy channel model with a character language model.

**IHD_RD (Supranovich and Patsepnia, 2015)** non-standard words are identified using a CRF tagger, using features such as token-level features, contextual tokens, dictionary lookup, and edit distance. Multiple lexicons are combined to generate normalization candidates. A query misspelling correction module (i.e., DidYouMean) is used to post-process the output.

**USZEGED (Berend and Tasnádi, 2015)** A CRF model is used to identify tokens requiring normalization, and determine the type of normalization required. Normalization candidates are then proposed based on revised edit distance. The final normalization candidate is selected on the basis of *n*-grams tatistics.

**BEKLI (Beckley, 2015)** A substitution dictionary is constructed in which keys are non-standard words and values are lists of potential normalizations. Frequent morphology errors are captured by hand-crafted rules. Finally, the Viterbi algorithm is applied to bigram sequences to decode the normalized sentence with maximum probability.

**LYSGROUP (Mosquera et al., 2015)** A system originally developed for Spanish text normalization was adapted to English text normalization. The method consists of a cascaded pipeline of several data adaptors and processors, such as a Twitter POS tagger and a spell checker.

## 3 Named Entity Recognition over Twitter

The second shared task of WNUT2015 is named entity recognition over Twitter data. Named entity recognition is a crucial component in many information extraction pipelines, however the majority of available NER tools were developed for newswire text and perform poorly on informal text genres such as Twitter. While performance on named entity recognition in newswire is quite high (Tjong Kim Sang and De Meulder, 2003), state-

| Team name | Precision | Recall | F1 | Method highlights |
|---|---|---|---|---|
| NCSU_SAS_NING | 0.9061 | 0.7865 | 0.8421 | Random Forest |
| NCSU_SAS_WOOKHEE | 0.9136 | 0.7398 | 0.8175 | Lexicon + LSTM |
| NCSU_SAS_SAM | 0.9012 | 0.7437 | 0.8149 | ANN |
| IITP | 0.9026 | 0.7191 | 0.8005 | CRF + Rule |
| DCU-ADAPT | 0.8190 | 0.5509 | 0.6587 | Generalized Perceptron |
| LYSGROUP | 0.4646 | 0.6281 | 0.5341 | Spanish Normalization Adaption |

Table 3: Results of the constrained systems for the lexical normalization shared task

| Team name | Precision | Recall | F1 | Method highlights |
|---|---|---|---|---|
| IHS_RD | 0.8469 | 0.8083 | 0.8272 | Lexicon + CRF + DidYouMean |
| USZEGED | 0.8606 | 0.7564 | 0.8052 | CRF + *n*-gram[ s] |
| BEKLI | 0.7743 | 0.7416 | 0.7571 | Lexicon + Rule + Ranker |
| GIGO | 0.7593 | 0.6963 | 0.7264 | N/A |
| LYSGROUP | 0.4592 | 0.6296 | 0.5310 | Spanish Normalization Adaption |

Table 4: Results of the unconstrained systems for the lexical normalization shared task

of-the-art performance on Twitter data lags far behind.

The diverse and noisy style of user-generated content presents serious challenges. For instance tweets, unlike edited newswire text, contain numerous nonstandard spellings, abbreviations, unreliable capitalization, etc.

Another challenge is concept drift (Dredze et al., 2010; Fromreide et al., 2014); the distribution of language and topics on Twitter is constantly shifting leading to degraded performance of NLP tools over time. To evaluate the effect of drift in a realistic scenario, the current evaluation uses a test set from a separate time period, which was not announced to participants until the (unannotated) test data was released at the beginning of the evaluation period.

To address these challenges, there has been an increasing body of work on adapting named entity recognition tools to noisy social media text (Derczynski et al., 2015b; Plank et al., 2014a; Cherry and Guo, 2015; Ritter et al., 2011; Plank et al., 2014b), however different research groups have made use of different evaluation setups (e.g. training / test splits) making it challenging to perform direct comparisons across systems. By organizing a shared evaluation we hope to help establish a common evaluation methodology (for at least one dataset) and also promote research and development of NLP tools for user-generated social media text genres.

### 3.1 Training and Development Data

The training and development data for our task was taken from previous work on Twitter NER (Ritter et al., 2011), which distinguishes 10 different named entity types (see Table 5 for the set of types). The data was split into 1,795 annotated tweets for training (`train`) and 599 as a development set (`dev`). Participants were allowed to use the development data for training purposes in their final submissions. This data was gathered in September 2010 and annotated by the 5th author.

### 3.2 Test Data Annotation

The test data was randomly sampled from December 2014 through February 2015. Two native English speakers were recruited to independently annotate the test data. The annotators were presented with a set of simple guidelines[6] that cover common ambiguous cases and also instructed to refer to the September 2010 data for reference. The BRAT tool[7] was used for annotation. A screenshot of the interface presented to annotators is shown in Figure 2. During an initial training period, both annotators independently labeled a set of 200 tweets after which disagreements were discussed and resolved before moving on to annotate the final test set. This initial annotation was only done

[6] `http://bit.ly/1FSP6i2`
[7] `http://brat.nlplab.org/`

for the purpose of training the annotators and the resulting data was discarded.

The annotators then went on to double-annotate a set of 1,425 messages. An adjudicator, the annotator of the training and dev sets, went through each message and resolved disagreements. The dataset was randomly split into 425 messages as an additional development set (`dev2015`) which was released to participants at the beginning of the evaluation period. The remaining 1,000 messages (`test`) were used for the final evaluation; annotations on the test data were withheld from participants until the end of the evaluation period.

Table 5 presents precision and recall for each of the 10 categories treating one annotator's labels as gold and the other's as predicted. This exposes the challenging nature of this annotation task and can be viewed as a kind of human upper bound on possible system performance, though we believe the consistency of the final annotations to be somewhat higher due to the second pass made by the adjudicator. The value of Cohen's $\kappa$ as measured on word-level annotations is 0.607.

A baseline system was provided to participants which takes a simple approach based on CRFsuite[8] using a standard set of features which include contextual, orthographic and gazetteers generated from Freebase (Bollacker et al., 2008). The evaluation consisted of 2 sub-tasks: one in which participants' systems were required to segment and classify 10 named entity types and one where the task is only to predict entity segmentation (no types).

### 3.3 Approaches

Eight teams (Table 6) participated in the named entity recognition shared task. A wide variety of approaches were taken to tackle this task. Table 7 summarizes the features used by each team and the machine learning approach taken. Many teams made use of word embeddings and Brown clusters as features. One team (multimedialab) used absolutely no hand-engineered features, relying entirely on word embeddings and a feed-forward neural-network (FFNN) architecture (Godin et al., 2015). Other new approaches to Twitter NER include a semi-Markov MIRA trained tagger developed by the NRC team (Cherry and Guo, 2015) and the use of entity-linking based features by ousia (Yamada et al., 2015). All the other teams used CRFs. On top of a CRF, the iitp team used a differential evolution based technique to obtain an optimal feature set.

Most systems used the training data as well as both dev sets provided to train their system, except multimedialab which did not use `dev2015` as training data and NRC which only used `train`. [9]

Tables 8 and 9 report the results obtained by each team for segmentation and classification of the 10 named entity types and for segmentation only, respectively.

### 3.4 System Descriptions

Following is a brief description of the approach taken by each team:

| | Precision | Recall | $F_{\beta=1}$ |
|---|---|---|---|
| company | 41.46 | 33.33 | 36.96 |
| facility | 50.00 | 66.67 | 57.14 |
| geo-loc | 63.57 | 70.09 | 66.67 |
| movie | 35.71 | 35.71 | 35.71 |
| musicartist | 60.98 | 47.17 | 53.19 |
| other | 48.21 | 50.00 | 49.09 |
| person | 60.42 | 80.56 | 69.05 |
| product | 44.83 | 19.12 | 26.80 |
| sportsteam | 75.00 | 71.74 | 73.33 |
| tvshow | 55.56 | 50.00 | 52.63 |
| Overall | 56.64 | 57.52 | 57.07 |

Table 5: Precision and recall comparing one annotator against the other. Cohen's kappa between the annotators was 0.607. Disagreements between the annotators resolved by a 3rd adjudicator for the final datasets.

| Team ID | Affiliation |
|---|---|
| Hallym | Hallym University |
| iitp | Indian Institute of Technology Patna |
| lattice | University Paris 3 |
| multimedialab | UGent - iMinds |
| NLANGP | Institute for Infocomm Research |
| nrc | National Research Council Canada |
| ousia | Studio Ousia |
| USFD | University of Sheffield |

Table 6: Team ID and affiliation of the named entity recognition shared task participants.

[8] `http://www.chokkan.org/software/crfsuite/`

[9] A post-competition analysis of the effect of training on development sets is presented in the NRC system description paper (Cherry et al., 2015).

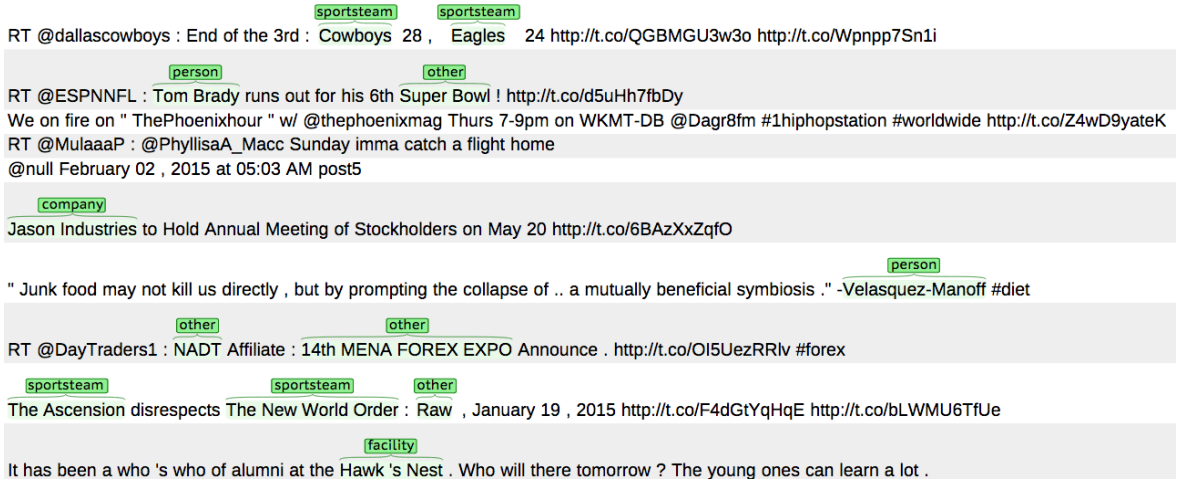


Figure 2: Annotation interface.

| | POS | Orthographic | Gazetteers | Brown clustering | Word embedding | ML |
|---|---|---|---|---|---|---|
| BASELINE | – | ✓ | ✓ | – | – | CRFsuite |
| Hallym | ✓ | – | – | ✓ | correlation analysis | CRFsuite |
| iitp | ✓ | ✓ | ✓ | – | – | CRF++ |
| lattice | ✓ | ✓ | – | ✓ | – | CRF wapiti |
| multimedialab | – | – | – | – | word2vec | FFNN |
| NLANGP | – | ✓ | ✓ | ✓ | word2vec & GloVe | CRF++ |
| nrc | – | – | ✓ | ✓ | word2vec | semi-Markov MIRA |
| ousia | ✓ | ✓ | ✓ | – | ✓ | entity linking |
| USFD | ✓ | ✓ | ✓ | ✓ | – | CRF L-BFGS |

Table 7: Features and machine learning approach taken by each team.

| | Precision | Recall | $F_{\beta=1}$ |
|---|---|---|---|
| ousia | 57.66 | 55.22 | 56.41 |
| NLANGP | 63.62 | 43.12 | 51.40 |
| nrc | 53.24 | 38.58 | 44.74 |
| multimedialab | 49.52 | 39.18 | 43.75 |
| USFD | 45.72 | 39.64 | 42.46 |
| iitp | 60.68 | 29.65 | 39.84 |
| Hallym | 39.59 | 35.10 | 37.21 |
| lattice | 55.17 | 9.68 | 16.47 |
| BASELINE | 35.56 | 29.05 | 31.97 |

Table 8: Results segmenting and categorizing entities into 10 types.

| | Precision | Recall | $F_{\beta=1}$ |
|---|---|---|---|
| ousia | 72.20 | 69.14 | 70.63 |
| NLANGP | 67.74 | 54.31 | 60.29 |
| USFD | 63.81 | 56.28 | 59.81 |
| multimedialab | 62.93 | 55.22 | 58.82 |
| nrc | 62.13 | 54.61 | 58.13 |
| iitp | 63.43 | 51.44 | 56.81 |
| Hallym | 58.36 | 48.5 | 53.01 |
| lattice | 58.42 | 25.72 | 35.71 |
| BASELINE | 53.86 | 46.44 | 49.88 |

Table 9: Results on segmentation only (no types).

**Hallym (Yang and Kim, 2015)** The Hallym team used an approach based on CRFs using both Brown clusters and word embeddings trained using Canonical Correlation Analysis as features.

**iitp (Akhtar et al., 2015a)** The iitp team proposed a multi-objective differential evolution based technique for feature selection in twitter named entity recognition.

**lattice (Tian, 2015)** Lattice employed a CRF model using Wapiti. The feature templates consisted of standard features used in state-of-the-art. They trained first a model with

dev_2015 and evaluated this model on train and dev.

**multimedialab (Godin et al., 2015)** The goal of the multimedia lab system was to only use neural networks and word embeddings to show the power of automatic feature learning and semi-supervised methods. A Feed-Forward Neural Network was first trained, that used only word2vec word embeddings as input. Word embeddings were trained on 400 million unlabeled tweets. Leaky ReLUs were used as activation function in combination with dropout to prevent overfitting. A context window of 5 words was used As input (2 words left and right). The output is a single tag of the middle word. Afterwards, a rule-based post-processing step was executed to ensure every I-tag has a B-tag in front of it and that all tags within a single span are of the same type. Train and dev were used as training data and used dev_2015 as validation set.

**NLANGP (Toh et al., 2015)** The NLANGP team modeled the problem as a sequential labeling task and used Conditional Random Fields. Several post-processing steps (e.g. rule-based matching) were applied to refine the system output. Besides Brown clusters, K-means clusters were also used; the K-means clusters were generated based on word embeddings.

**nrc (Cherry et al., 2015)** NRC applied a MIRA-trained semi-Markov tagger with Gazetteer, Brown cluster and Word Embedding features. The Word Embeddings were built over phrases using Word2Vec's phrase finder tool, and were modified using an auto-encoder to be predictive of Gazetteer membership.

**ousia (Yamada et al., 2015)** The main characteristics of the ousia method is enhancing the performance of Twitter named entity recognition using entity linking. Once entity mentions are disambiguated to the knowledge base entries, high-quality knowledge can be easily extracted from a knowledge base such as the popularity of the entity, the classes of the entity, and the likelihood that the entity appears in the given context. They adopted supervised machine-learning with features including the results of NER and various information of the entity in knowledge bases. We use Stanford NER was used for the NER and in-house end-to-end entity linking software was applied for entity linking.

**USFD (Derczynski et al., 2015a)** Feature extraction was based on large Brown clusters, gazetteers tuned to the input data, and distant supervision from Freebase. The representation was tuned for drift by down-weighting temporally distant training examples. The classifier was a linear chain CRF with hyperparameters tuned for Twitter.

## 4 Summary

In this paper, we presented two shared tasks on Twitter text processing: Lexical Normalization and Named Entity Recognition. We detailed the task setup and datasets used in the respective shared tasks, and also outlined the approach taken by the participating systems. Both shared tasks were of a scale substantially larger than what had previously been attempted in the literature, with two primary benefits. First, we are able to draw stronger conclusions about the true potential of different approaches. Second, through analyzing the results of the participating systems, we are able to suggest potential research directions for both future shared tasks and noisy text processing in general.

## Acknowledgments

We would like to thank Svitlana Volkova and Junming Xu for feedback on a previous draft of this paper. We also thank Javier Angel and Gabriella Talvy for annotating the test data for the named entity recognition shared task, and IBM Research Australia for the generous support in doing the annotation for the lexical normalization shared task.

## References

Md Shad Akhtar, Utpal Kumar Sikdar, and Asif Ekbal. 2015a. Iitp: Multiobjective differential evolution based twitter named entity recognition. In *proceedings of WNUT*.

Md Shad Akhtar, Utpal Kumar Sikdar, and Asif Ekbal. 2015b. Iitp: Hybrid approach for text normalization in twitter. In *proceedings of WNUT*, Beijing, China.

AiTi Aw, Min Zhang, Juan Xiao, and Jian Su. 2006. A phrase-based statistical model for SMS text normalization. In *Proceedings of COLING/ACL 2006*, pages 33–40, Sydney, Australia.

Russell Beckley. 2015. Bekli:a simple approach to twitter text normalization. In *proceedings of WNUT*, Beijing, China.

Gabor Berend and Ervin Tasnádi. 2015. Uszeged: Correction type-sensitive normalization of english tweets using efficiently indexed n-gram statistics. In *proceedings of WNUT*, Beijing, China.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM.

Colin Cherry and Hongyu Guo. 2015. The unreasonable effectiveness of word representations for twitter named entity recognition. NAACL.

Colin Cherry, Hongyu Guo, and Chengbi Dai. 2015. Nrc: Infused phrase vectors and updated gazetteers for named entity recognition in twitter. In *proceedings of WNUT*.

Grzegorz Chrupała. 2014. Normalizing tweets with edit scripts and recurrent neural embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pages 680–686, Baltimore, USA, June.

Paul Cook and Suzanne Stevenson. 2009. An unsupervised model for text message normalization. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity (CALC '09)*, pages 71–78, Boulder, USA.

Leon Derczynski, Isabelle Augenstein, and Kalina Bontcheva. 2015a. Usfd: Twitter ner with drift compensation and linked data. In *proceedings of WNUT*.

Leon Derczynski, Diana Maynard, Giuseppe Rizzo, Marieke van Erp, Genevieve Gorrell, Raphaël Troncy, Johann Petrak, and Kalina Bontcheva. 2015b. Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32–49.

Mark Dredze, Tim Oates, and Christine Piatko. 2010. We're not in kansas anymore: detecting domain changes in streams. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 585–595. Association for Computational Linguistics.

Hege Fromreide, Dirk Hovy, and Anders Søgaard. 2014. Crowdsourcing and annotating ner for twitter# drift. *European language resources distribution agency*.

Fréderic Godin, Baptist Vandersmissen, Wesley De Neve, and Rik Van de Walle. 2015. multimedialab @ acl wnut ner shared task: Named entity recognition for twitter microposts using distributed word representations. In *proceedings of WNUT*.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL HLT 2011)*, pages 368–378, Portland, USA.

Bo Han, Paul Cook, and Timothy Baldwin. 2012. Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning 2012 (EMNLP-CoNLL 2012)*, pages 421–432, Jeju Island, Korea, July.

Bo Han, Paul Cook, and Timothy Baldwin. 2013. Lexical normalisation for social media text. *ACM Transactions on Intelligent Systems and Technology*, 4(1):5:1–5:27.

Hany Hassan and Arul Menezes. 2013. Social text normalization using contextual graph random walks. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pages 1577–1586, Sofia, Bulgaria, August.

Ning Jin. 2015. Ncsu-sas-ning: Candidate generation and feature engineering for supervised lexical normalization. In *proceedings of WNUT*, Beijing, China.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and A. Noah Smith. 2014. A dependency parser for tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1001–1012.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*.

Samuel Leeman-Munk, James Lester, and James Cox. 2015. Ncsu_sas_sam: Deep encoding and reconstruction for normalization of noisy text. In *proceedings of WNUT*, Beijing, China.

Wang Ling, Chris Dyer, Alan W Black, and Isabel Trancoso. 2013. Paraphrasing 4 microblog normalization. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 73–84, Seattle, USA, October.

Fei Liu, Fuliang Weng, Bingqing Wang, and Yang Liu. 2011. Insertion, deletion, or substitution? Normalizing text messages without pre-categorization nor supervision. In *Proceedings of the 49th Annual*

*Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL HLT 2011)*, pages 71–76, Portland, USA.

Fei Liu, Fuliang Weng, and Xiao Jiang. 2012. A broad-coverage normalization system for social media language. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, pages 1035–1044, Jeju Island, Korea, July.

Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012) Demo Session*, pages 25–30, Jeju, Republic of Korea.

Wookhee Min, Bradford Mott, James Lester, and James Cox. 2015. Ncsu_sas_wookhee: A deep contextual long-short term memory model for text normalization. In *proceedings of WNUT*, Beijing, China.

Yerai Doval Mosquera, Jesús Vilares, and Carlos Gómez-Rodríguez. 2015. Lysgroup: Adapting a spanish microtext normalization system to english. In *proceedings of WNUT*, Beijing, China.

Barbara Plank, Dirk Hovy, Ryan McDonald, and Anders Søgaard. 2014a. Adapting taggers to twitter with not-so-distant supervision. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1783–1792.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014b. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of EACL*.

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

Richard Sproat, Alan W. Black, Stanley Chen, Shankar Kumar, Mari Ostendorf, and Christopher Richards. 2001. Normalization of non-standard words. *Computer Speech and Language*, 15(3):287–333.

Dmitry Supranovich and Viachaslau Patsepnia. 2015. Ihs_rd: Lexical normalization for english tweets. In *proceedings of WNUT*, Beijing, China.

Tian Tian. 2015. Data adaptation for named entity recognition on tweets with features-rich crf. In *proceedings of WNUT*.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Zhiqiang Toh, Bin Chen, and Jian Su. 2015. Improving twitter named entity recognition using word representations. In *proceedings of WNUT*.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics.

Joachim Wagner and Jennifer Foster. 2015. Dcu-adapt: Learning edit operations for microblog normalisation with the generalised perceptron. In *proceedings of WNUT*, Beijing, China.

Pidong Wang and Hwee Tou Ng. 2013. A beam-search decoder for normalization of social media text with application to machine translation. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2013)*, pages 471–481, Atlanta, USA, June.

Ikuya Yamada, Hideaki Takeda, and Takefuji Yoshiyasu. 2015. Enhancing named entity recognition in twitter messages using entity linking. In *proceedings of WNUT*.

Yi Yang and Jacob Eisenstein. 2013. A log-linear model for unsupervised text normalization. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 61–72, Seattle, USA, October.

Eun-Suk Yang and Yu-Seop Kim. 2015. Hallym: Named entity recognition on twitter. In *proceedings of WNUT*.