Proceedings of the

Concept Extraction Challenge

at the 3rd Workshop on

Making Sense of Microposts (#MSM2013)

Big things come in small packages

at the 22nd International Conference on the World Wide Web (WWW'13)
Rio de Janeiro, Brazil
13th of May 2013

edited by

Amparo Cano, Matthew Rowe, Milan Stankovic, Aba-Sah Dadzie

Preface

The 3rd Workshop on *Making Sense of Microposts (#MSM2013)* was held in Rio de Janeiro, Brazil, on the 13th of May 2013, as part of the 22nd International Conference on the World Wide Web (WWW'13). #MSM2013 is the third in a series of successful workshops. The #MSM workshop was first held at the 8th Extended Semantic Web Conference (ESWC 2011), and with approximately 50 participants, was the most popular workshop at ESWC 2011. The second workshop was held at the 21st International Conference on the World Wide Web (WWW'12), and had approximately 60 participants, as did this year's workshop.

The #MSM series of workshops is unique in targeting both Semantic Web researchers and other fields, within Computer Science, such as Human-Computer Interaction and Visualisation, and in other areas, particularly the Social Sciences. The aim is to harness the benefits different fields bring to research involving Microposts. The workshop also encourages the demonstration of the generation and use of Microposts through different physical and online media, as well as application of research, and re-use of Micropost data in real-world scenarios. Continuing to hold the workshop at WWW allows us to reach a wider and more varied audience and target research and applications at the leading edge of technology. The 2013 edition was an occasion to expand our community, and with the conference in Rio de Janeiro, to connect with local researchers from Brazil and South America, opening the way for new synergy and interesting discussions within the local cultural context.

In a world where more and more data is becoming available to machines, questions related to the use of this data for increasing machine intelligence naturally arise. Big Data treatment efforts exploit masses of data using statistical approaches in order to conceive anticipatory systems able to predict future human behaviour and adapt to it. Semantic analysis of Web content, including Microposts, is another complementary perspective to the goal of making machines more intelligent and more capable of supporting daily human activity, decision making and communication. We are seeing a very large increase in systems relying on Semantic Web technologies being deployed: Intelligent Assistants, such as Siri[1], rely on Semantic Data Graphs to provide users with factual responses to their questions. Facebook Graph Search[2] allows users to formulate complex queries over a socio-semantic graph constructed from people's *likes* and structural knowledge about things being *liked*. While static knowledge bases are largely employed in such systems, exploiting the dynamic, evolving knowledge that resides in the growing masses of Microposts, invaluable as they are acknowledged to be, remains a major challenge.

Each year we make a little step toward resolving this challenge, due largely to what makes publishing via Microposts so popular – their brevity, and as a

[1] `http://www.apple.com/ios/siri`

[2] `https://www.facebook.com/about/graphsearch`

result, the use of non-standard abbreviations, informal language and grammar. With each workshop we have found that our research community continues to open exciting new possibilities for constructing increasingly intelligent and useful services.

New to this year's workshop is the ***Concept Extraction Challenge***, sponsored by eBay. Existing concept extraction tools are intended for use over news corpora and similar document-based corpora with relatively long length. The aim of the challenge was to foster research into novel, more accurate concept extraction for (much shorter) Micropost data. The keen interest in concept extraction that is shared by our community motivated this challenge, focused for this first time on a rather general task. The interest shown in the challenge by both academia and industry has confirmed its relevance. We aim to pursue the challenge in the future editions of #MSM, and are investigating new challenge tasks and the use of different collections of data, prompted by the challenge results and further research it continues to foster.

This first run of the challenge has been a learning curve, with contributions from participants, not just in their formal submissions, but also to corrections in the training data that fed into the cycle of updates that resulted in the final gold standard. The #MSM2013 Concept Extraction Challenge received 22 complete submissions, out of which 6 were accepted for presentation at the workshop, and a further 7 for presentation as posters. Submissions came from institutions across 12 countries, with 13% of submitting authors from Brazilian institutions.

Many hearty thanks to all our contributors and participants, and also the Programme Committees whose valued feedback resulted in a rich collection of work, each of which adds to the state of the art in leading edge research in the challenging task of information extraction from Microposts. Especial thanks to Andrea Varga, who was largely responsible for generating the challenge dataset, and Pablo Mendes who gave us very useful suggestions on collaborative annotation of the data. We are confident that the #MSM series of workshops will continue to foster a vibrant community, and target the rich body of information generated by the many and varied authors whose social and working lives span the physical and online worlds.

Amparo E. Cano KMi, The Open University, UK
Matthew Rowe Lancaster University, UK
Milan Stankovic Université Paris-Sorbonne, France
Aba-Sah Dadzie The University of Sheffield, UK
#MSM2013 Concept Extraction Challenge Organising Committee, May '13

Summary of Other Contributions to #MSM2013

Published with ACM as a companion volume to the WWW'13 proceedings, the main track[3] received 13 paper submissions, out of which 4 full and 2 short papers were accepted. This was in addition to a poster and demo session, to exhibit practical application in the field, and foster further discussion about the ways in which data extracted from Microposts is being reused. The accepted submissions cover an array of topics, including a variety of approaches to concept extraction – again reinforcing its importance with respect to research on Microposts, among these, rule-based, machine learning and hybrid methods. Other topics covered range from research from a social science perspective, on the use of Microposts to publicise and discuss trending events and topics, and the extraction of intent, meaning and sentiment. Submissions came from 9 countries, with 29% of all authors from institutions in Brazil. Thanks to our local chair, Bernardo Pereira Nunes, who helped, among other things, to promote the workshop and challenge to local institutions.

The main track proceedings include also the keynote abstract, '*Urban*: Crowdsourcing for the good of London*'[4], presented by Daniele Quercia, of the Cambridge Networks Network at the University of Cambridge, England, UK.

The #MSM2013 award for the best paper, based on nominations by the reviewers and confirmed by the workshop chairs, was awarded to:

> *Lisa Posch, Claudia Wagner, Philipp Singer & Markus Strohmaier*
> for the paper:
>
> > ***Meaning as Collective Use: Predicting Hashtag Semantics on Twitter***[5]

Introduction to the #MSM2013 Challenge Proceedings

This volume includes first a challenge report, with a summary of the state of the art and a comparison of the performance of the approaches taken for the 13 submissions accepted. This provides an overview of the capability of the state of the art in Concept Extraction approaches to date. This introductory paper details the challenge objectives and task, and the dataset construction and validation processes. We also provide a comprehensive description of the

[3] #MSM2013 welcome: `http://dl.acm.org/citation.cfm?id=2490000.2487998`

[4] #MSM2013 keynote: `http://dl.acm.org/citation.cfm?id=2487788.2488000`

[5] Best paper, main track: `http://dl.acm.org/citation.cfm?id=2487788.2488008`

quantitative evaluation methodology followed and the performance and ranking metrics used.

Participants' descriptions of the systems implemented complete the proceedings. Each submission was peer reviewed, to provide the authors with feedback on their approach and to identify interesting and promising work to present at the workshop. The quantitative evaluation described in the report was also carried out to rank submission runs – this was the final criterion, with a cut-off for acceptance, and the key measure for the challenge award.

Concept Extraction Challenge Award

eBay[6] sponsored the challenge award: US$ 1,500, for the best submission. Nominations were sought from the reviewers, and a final decision agreed by the challenge chairs, based on their nominations, review scores and the results of the quantitative evaluation. The #MSM2013 Concept Extraction Challenge Award went to:

Mena Habib, Maurice Van Keulen & Zhemin Zhu
for their submission entitled:
University of Twente at #MSM2013

[6] http://www.ebayinc.com

Challenge Evaluation Committee

Naren Chittar eBay, USA
Óscar Corcho Universidad Politécnica de Madrid, Spain
Danica Damljanovic Kuato Studios, London, UK
Anna Lisa Gentile The University of Sheffield, UK
Diana Maynard The University of Sheffield, UK
Peter Mika Yahoo! Research, Spain
Enrico Motta KMi, The Open University, UK
Daniel Preotiuc The University of Sheffield, UK
Alan Ritter University of Washington, USA
Guiseppe Rizzo Eurecom, France
Raphaël Troncy Eurecom, France
Victoria Uren Aston Business School, UK
Andrea Varga The University of Sheffield, UK

Additional Material

The call for participation and all challenge abstracts, in addition to those for the main workshop track, are available on the #MSM2013 website[7]. The full challenge proceedings are also available on the CEUR-WS server, as Vol-1019[8]. The proceedings for the main track are available as part of the WWW'13 Proceedings Companion[9]. The proceedings for the 1st and 2nd workshops are available as CEUR Vol-718[10] and Vol-838[11] respectively.

[7] Challenge web pages: http://oak.dcs.shef.ac.uk/msm2013/ie_challenge.html
[8] #MSM2013 Challenge proceedings: http://ceur-ws.org/Vol-1019
[9] WWW'13 Proceedings Companion: http://dl.acm.org/citation.cfm?id=2487788
[10] #MSM2011 proceedings: http://ceur-ws.org/Vol-718
[11] #MSM2012 proceedings: http://ceur-ws.org/Vol-838

Table of Contents

Summary of Results

Challenge Submissions – Section I

Challenge Submissions – Section II

Appendices

Summary of Results –

the Making Sense of Microposts (#MSM2013)

Concept Extraction Challenge

Making Sense of Microposts (#MSM2013) Concept Extraction Challenge

Amparo Elizabeth Cano Basave[1]*, Andrea Varga[2],
Matthew Rowe[3], Milan Stankovic[4], and Aba-Sah Dadzie[2]**

[1] KMi, The Open University, Milton Keynes, UK
[2] The OAK Group, Dept. of Computer Science, The University of Sheffield, UK
[3] School of Computing and Communications, Lancaster University, UK
[4] Sépage, Paris, France
a.cano_basave@aston.ac.uk,a.varga@dcs.shef.ac.uk
m.rowe@lancaster.ac.uk,milstan@gmail.com,a.dadzie@cs.bham.ac.uk

Abstract. Microposts are small fragments of social media content that have been published using a lightweight paradigm (e.g. Tweets, Facebook likes, foursquare check-ins). Microposts have been used for a variety of applications (e.g., sentiment analysis, opinion mining, trend analysis), by gleaning useful information, often using third-party concept extraction tools. There has been very large uptake of such tools in the last few years, along with the creation and adoption of new methods for concept extraction. However, the evaluation of such efforts has been largely consigned to document corpora (e.g. news articles), questioning the suitability of concept extraction tools and methods for Micropost data. This report describes the Making Sense of Microposts Workshop (#MSM2013) Concept Extraction Challenge, hosted in conjunction with the 2013 World Wide Web conference (WWW'13). The Challenge dataset comprised a manually annotated training corpus of Microposts and an unlabelled test corpus. Participants were set the task of engineering a concept extraction system for a defined set of concepts. Out of a total of 22 complete submissions 13 were accepted for presentation at the workshop; the submissions covered methods ranging from sequence mining algorithms for attribute extraction to part-of-speech tagging for Micropost cleaning and rule-based and discriminative models for token classification. In this report we describe the evaluation process and explain the performance of different approaches in different contexts.

1 Introduction

Since the first Making Sense of Microposts (#MSM) workshop at the Extended Semantic Web Conference in 2011 through to the most recent workshop in 2013

[*] A.E. Cano Basave has since changed affiliation, to: Engineering and Applied Science, Aston University, Birmingham, UK (e-mail as above).

[**] A.-S. Dadzie has since changed affiliation, to: School of Computer Science, University of Birmingham, Edgbaston, Birmingham, UK (e-mail as above).

we have received over 60 submissions covering a wide range of topics related to interpreting Microposts and (re)using the knowledge content of Microposts. One central theme that has run through such work has been the need to understand and learn from Microposts (social network-based posts that are small in size and published using minimal effort from a variety of applications and on different devices), so that such information, given its public availability and ease of retrieval, can be reused in different applications and contexts (e.g. music recommendation, social bots, news feeds). Such usage often requires identifying entities or concepts in Microposts, and extracting them accordingly. However this can be hindered by:

(i) the noisy lexical nature of Microposts, where terminology differs between users when referring to the same thing and abbreviations are commonplace;

(ii) the limited length of Microposts, which restricts the contextual information and cues that are available in normal document corpora.

The exponential increase in the rate of publication and availability of Microposts (Tweets, FourSquare check-ins, Facebook status updates, etc.), and applications used to generate them, has led to an increase in the use of third-party entity extraction APIs and tools. These function by taking as input a given text, identifying entities within them, and extracting entity type-value tuples. Rizzo & Troncy [12] evaluated the performance of entity extraction APIs over news corpora, assessing the performance of extraction and entity disambiguation. This work has been invaluable in providing a reference point for judging the performance of extraction APIs over well-structured news data. However, an assessment of the performance of extraction APIs over Microposts has yet to be performed.

This prompted the Concept Extraction Challenge held as part of the *Making Sense of Microposts Workshop (#MSM2013)* at the *2013 World Wide Web Conference (WWW'13)*. The rationale behind this was that such a challenge, in an open and competitive environment, would encourage and advance novel, improved approaches to extracting concepts from Microposts. This report describes the #MSM2013 Concept Extraction Challenge, collaborative annotation of the corpus of Microposts and our evaluation of the performance of each submission. We also describe the approaches taken in the systems entered – using both established and developing alternative approaches to concept extraction, how well they performed, and how system performance differed across concepts. The resulting body of work has implications for researchers interested in the task of extracting information from social data, and for application designers and engineers who wish to harvest information from Microposts for their own applications.

2 The Challenge

We begin by describing the goal of the challenge and the task set, and the process we followed to generate the corpus of Microposts. We conclude this section with the list of submissions accepted.

2.1 The Task and Goal

The challenge required participants to build semi-automated systems to identify concepts within Microposts and extract matching entity types for each concept identified, where *concepts* are defined as abstract notions of *things*. In order to focus the challenge we restricted the classification to four entity types:

(i) Person **PER**, e.g. Obama;
(ii) Organisation **ORG**, e.g. NASA;
(iii) Location **LOC**, e.g. New York;
(iv) Miscellaneous **MISC**, consisting of the following: film/movie, entertainment award event, political event, programming language, sporting event and TV show.

Submissions were required to recognise these entity types within each Micropost, and extract the corresponding entity type-value tuples from the Micropost. Consider the following example, taken from our annotated corpus:

```
870,000 people in canada depend on #foodbanks
−25% increase in the last 2 years − please give generously
```

The fourth token in this Micropost refers to the location *Canada*; an entry to the challenge would be required to spot this token and extract it as an annotation, as:

```
LOC/canada;
```

The complete description of concept types and their scope, and additional examples can be found on the challenge website[5], and also in the appendices in the challenge proceedings.

To encourage competitiveness we solicited sponsorship for the winning submission. This was provided by the online auctioning web site eBay[6], who offered a $1500 prize for the winning entry. This generous sponsorship is testimony to the growing industry interest in issues related to automatic understanding of short, predominantly textual posts – Microposts; challenges faced by major Social Web and other web sites, and increasingly, marketing and consumer analysts and customer support across industry, government, state and not-for-profits organisations around the world.

2.2 Data Collection and Annotation

The dataset consists of the message fields of each of 4341 manually annotated Microposts, on a variety of topics, including comments on the news and politics, collected from the end of 2010 to the beginning of 2011, with a 60% / 40% split between training and test data. The annotation of each Micropost in the training dataset gave all participants a common base from which to learn extraction patterns. The test dataset contained no annotations; the challenge task was for

[5] `http://oak.dcs.shef.ac.uk/msm2013/challenge.html`
[6] `http://www.ebay.com`

participants to provide these. The complete dataset, including a list of changes and the gold standard, is available on the #MSM2013 challenge web pages[7], accessible under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

To assess the performance of the submissions we used an underlying *ground truth*, or *gold standard*. In the first instance, the dataset was annotated by two of the authors of this report. Subsequent to this we logged corrections to the annotations in the training data submitted by participants, following which we release an updated dataset. After this, based on a recommendation, we set up a GitHub repository to simplify collaborative annotation of the dataset. Four of the authors of this report then annotated a quarter of the dataset each, and then checked the annotations that the other three had performed to verify correctness. For those entries for which consensus was not reached, discussion between all four annotators was used to come to a final conclusion. This process resulted in better quality and higher consensus in the annotations. A very small number of errors was reported subsequent to this; a final submission version with these corrections was used by participants for their last set of experiments and to submit their final results.

Figure 1 presents the entity type distributions over the training set, test set and over the entire corpus.

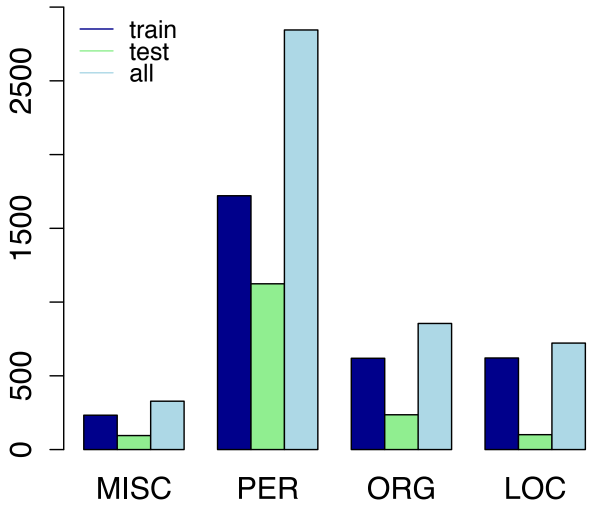


Fig. 1. Distributions of entity types in the dataset

[7] http://oak.dcs.shef.ac.uk/msm2013/ie_challenge

2.3 Challenge Submissions

Twenty-two complete submissions were received for the challenge; each of which consisted of a short paper explaining the system's approach, and up to three different test set annotations generated by running the system with different settings. After peer review, thirteen submissions were accepted; for each, the submission run with the best overall performance was taken as the result of the system, and used in the rankings. The accepted submissions are listed in Table 1, with the run taken as the result set for each.

Table 1. Submissions accepted, in order of submission, with authors and number of runs for each

Submission No.	Authors	No. of runs
submission_03	van Den Bosch, M. et al.	3
submission_14	Habib, M. et al.	1
submission_15	Van Erp, M. et al.	3
submission_20	Cortis, K.	1
submission_21	Dlugolinský, Š. et al.	3
submission_25	Godin, F. et al.	1
submission_28	Genc, Y. et al.	1
submission_29	Muñoz-García, O. et al.	1
submission_32	Hossein, A.	1
submission_30	Mendes, P. et al.	3
submission_33	Das, A. et al.	3
submission_34	de Oliveira, D. et al.	1
submission_35	Sachidanandan, S. et al.	1

2.4 System Descriptions

Participants approached the concept extraction task with *rule-based*, *machine learning* and *hybrid* methods. A summary of each approach can be found in Figure 2, with detail in the author descriptions that follow this report. We compared these approaches according to various dimensions: *state of the art (SoA) named entity recognition (NER) features* employed (columns 4-11) ([13,6]), *classifiers* used for both extraction and classification of entities (columns 12-13), additional *linguistic knowledge sources* used (column 14), special *pre-processing steps performed* (column 15), other *non-SoA NER features* used (column 16), and finally, the list of *off-the-shelf systems* incorporated (column 17).

From the results and participants' experiments we make a number of observations. With regard to the *strategy* employed, the best performing systems (from the top, 14, 21, 15, 25), based on overall F_1 score (see Section 3), were hybrid.

Strategy	System	Train	State-of-the-art features								Classifier used		Linguistic Knowledge	Prep.	Other Features	External Systems
			Token	Case	Morp.	POS	Function	Local syntax	List lookup	Cont ext	Extraction	Classification				
Rule-based	20	TW		IsCap		ANNIE Pos			DBpedia Gazetteer, ANNIE Gazetteer		ANNIE		DBpedia	RT, @, #, Slang, MissSpell		ANNIE [1]
	29	TW	Ngram			PosFreeling 2012, isNP	Token Length		Wiki Gazetteer, IsStopWord		Rules	Rules	DBpedia	RT, @, URL, #, Punct, MissSpell, LowerCase		Freeling [8]
	28	TW	Ngram			NLTKPos			Wiki Gazetteer		Rules	Rules	Wiki	Punct, Capitalise		NLTK [4]
	32	TW									BabelNet WSD	Rules	DBpedia, BabelNet			BabelNet API [7]
Data-driven	3	TW, CoNLL03, ACE04, ACE05				PosTreebank			Geonames.org Gazetteer; JRC names corpus		2 IGTree memory-based taggers			LowerCase		
	33	TW	Stem	IsCap		TwPos2011		Follows FW	Country names Gazetter, City names Gazzetter, IsOOV		CRF		Samsad & NICTA dictionary	URL, #, @, Punct		
	34	TW		IsCap	Prefix, Suffix				Wiki Gazetter, Freebase Gazetter	size of TW	CRF					
Hybrid	14	TW		IsCap, AllCap		TwPos2011			Yago, Microsoft N-grams, WordNet, TW		CRF+ SVM RBF	AIDA	Yago, Microsoft N-grams, WordNet	#, Slang, MissSpell	AIDA Scores	AIDA [15]
	21	TW		IsCap, AllCap, Lower Case		isNP, isVP	Token Length		Google Gazetter		C4.5 decision tree				ConfScores	ANNIE [1], OpenNLP, Illinois NET [9], Illinois Wikifier [10], LingPipe, OpenCalais, StanfordNER [2], WikiMiner
	15	TW		IsCap, AllCap	Prefix, Suffix	TwPos2011		First Word, Last Word			SVM SMO					StanfordNER [2], NERD [12], TWNer [11]
	25	TW									Random Forest					Alchemy, DBpedia Spotlight, OpenCalais, Zemanta
	30	TW	Ngram	IsCap, AllCap, Lower Case					DBpedia Gazetteer, BALIE Gazetteer	2	DBpedia Spotlight	CRF	DBpedia	RT, #, @, URL		DBpedia Spotlight
	35	TW	Ngram						Yago, Wiki, WordNet		Pagerank	CRF	Yago, Wiki,WordNet			

Fig. 2. Automated approaches for #MSM2013 Concept Extraction. Columns correspond to the strategies employed by the participants (Strategy), the id of the systems (System), the data used to train the concept extractors (Train), state of the art features [6], Token, Case, Morphology (Morph.), Part-of-speech (POS), Function, Local context, List lookup, Context window size (Context)), classifiers used for both entity extraction (Extraction) and classification, additional linguistic knowledge used for concept extraction (Linguistic Knowledge), preprocessing steps performed on the data (Prep.), additional features used for the extractors (Other Features), a list of off-the-self systems employed (External Systems).

The success of these models appears to rely on the application of off-the-shelf systems (e.g. AIDA [15], ANNIE [1], OpenNLP[8], Illinois NET [9], Illinois Wikifier [10], LingPipe[9], OpenCalais[10], StanfordNER [2], WikiMiner[11], NERD [12], TWNer [11], Alchemy[12], DBpedia Spotlight[5][13], Zemanta[14]) for either *entity extraction* (identifying the boundaries of an entity) or *classification* (assigning a semantic type to an entity). For the best performing system (14), the complete concept classification component was executed by the (existing) concept disambiguation tool AIDA. Other systems (21, 15, 25), on the other hand, made use of the output of multiple off-the-shelf systems, resulting in additional features (such as the confidence scores of each individual NER extractors – **ConfScores**) for the final concept extractors, balancing in this way the contribution of existing extractors.

Among the rule-based approaches, the winning strategy was also similar. Submission 20 achieved the fourth best result overall, by taking an existing rule-based system (ANNIE), and simply increasing the coverage of captured entities by building new gazetteers[15]. We also find that for entity extraction the participants used both rule-based and statistical approaches. Considering current state of the art approaches, statistical models are able to handle this task well.

Looking at *features*, the gazetteer membership and part-of-speech (POS) features played an important role; the best systems include these. For the gazetteers, a large number of different resources were used, including Yago, WordNet, DBpedia, Freebase, Microsoft N-grams and Google. Existing POS taggers were trained on newswire text (e.g. ANNIEPos [1], NLTKPos [4], POS trained on Treebank corpus (PosTreebank), Freeling [8]). Additionally, there appears to be a trend on incorporating recent POS taggers trained on Micropost data (e.g. **TwPos2011** [3]).

Considering *pre-processing* of Microposts, we find the following:

- removal of Twitter-specific markers, e.g. hashtags (#), mentions (@), retweets (**RT**),
- removal of external URL links within Microposts (**URL**),
- removal of punctuation marks (**Punct**), e.g. points, brackets,
- removal of well-known slang words using dictionaries[16] (**Slang**), e.g. "lol", "tmr", – unlikely to refer to named entities,

[8] `http://opennlp.apache.org`

[9] `http://alias-i.com/lingpipe`

[10] `http://www.opencalais.com`

[11] `http://wikipedia-miner.cms.waikato.ac.nz`

[12] `http://www.alchemyapi.com`

[13] `http://dbpedia.org/spotlight`

[14] `http://www.zemanta.com`

[15] Another off-the-shelf entity extractor employed was BabelNet API [7], in submission 32.

[16] `http://www.noslang.com/dictionary/full`
`http://www.chatslang.com/terms/twitter`
`http://www.chatslang.com/terms/facebook`

- removal of words representing exaggerative emotions (**MissSpell**), e.g. "nooooo", "goooooood", "hahahaha",
- transformation of each word to lowercase (**LowerCase**),
- capitalisation of the first letter of each word (**Capitalise**).

With respect to the *data* used for training the entity extractors, the majority of submissions utilised the challenge training dataset, containing annotated Micropost data (**TW**) alone. A single submission, (3, the sixth best system overall), made use of a large silver dataset (CoNLL 2003 [14], ACE 2004 and ACE 2005[17]) with the training dataset annotations, and achieved the best performance among the statistical methods.

3 Evaluation of Challenge Submissions

3.1 Evaluation Measures

The evaluation involved assessing the correctness of a system (S), in terms of the performance of the system's entity type classifiers when extracting entities from the test set (TS). For each instance in TS, a system must provide a set of tuples of the form: (entity type, entity value). The evaluation compared these output tuples against those in the gold standard (GS). The metrics used to evaluate these tuples were the standard precision (P), recall (R) and f-measure (F_1), calculated for each entity type. The final result for each system was the average performance across the four defined entity types.

To assess the correctness of the tuples of an entity type t provided by a system S, we performed a *strict match* between the tuples submitted and those in the GS. We consider a *strict match* as one in which there is an exact match, with conversion to lowercase, between a system value and the GS value for a given entity type t. Let $(x, y) \in S_t$ denote the set of tuples extracted for entity type t by system S, $(x, y) \in GS_t$ denote the set of tuples for entity type t in the gold standard. We define the set of True Positives (TP), False Positives (FP) and False Negatives (FN) for a given system as:

$$TP_t = \{(x, y) \mid (x, y) \in (S_t \cap GS_t)\} \tag{1}$$

$$FP_t = \{(x, y) \mid (x, y) \in S_t \wedge (x, y) \notin GS_t\} \tag{2}$$

$$FN_t = \{(x, y) \mid (x, y) \in GS_t \wedge (x, y) \notin S_t\} \tag{3}$$

Therefore TP_t defines the set of true positives considering the entity type and value of tuples; FP_t is the set of false positives considering the unexpected results for an entity type t; FN_t is the set of false negatives denoting the entities that were missed by the extraction system, yet appear within the gold standard. As we require matching of the tuples (x, y) we are looking for strict extraction matches, this means that a system must both detect the correct entity type (x)

[17] the ACE Program: `http://projects.ldc.upenn.edu/ace`

and extract the correct matching entity value (y) from a Micropost. From this set of definitions we define precision (P_t) and recall (R_t) for a given entity type t as follows:

$$P_t = \frac{|TP_t|}{|TP_t \cup FP_t|} \tag{4}$$

$$R_t = \frac{|TP_t|}{|TP_t \cup FN_t|} \tag{5}$$

As we compute the precision and recall on a per-entity-type basis, we define the average precision and recall of a given system S, and the harmonic mean, F_1 between these measures:

$$\bar{P} = \frac{P_{PER} + P_{ORG} + P_{LOC} + P_{MISC}}{4} \tag{6}$$

$$\bar{R} = \frac{R_{PER} + R_{ORG} + R_{LOC} + R_{MISC}}{4} \tag{7}$$

$$F_1 = 2 \times \frac{\bar{P} \times \bar{R}}{\bar{P} + \bar{R}} \tag{8}$$

3.2 Evaluation Results and Discussion

We report the differences in performance between participants' systems, with a focus on the differences in performance by entity type. The following subsections report results of the evaluated systems in terms of precision, recall and F-measure, following the metrics defined in subsection 3.1.

Precision. We begin by discussing the performance of the submissions in terms of precision. Precision measures the accuracy, or '*purity*', of the detected entities in terms of the proportion of false positives within the returned set: high precision equates to a low false positive rate. Table 3.2 shows that hybrid systems are the top 4 ranked systems (in descending order, 14, 21, 30, 15), suggesting that a combination of rules and data-driven approaches yields increased precision. Studying the features of the top-performing systems, we note that maintaining capitalisation is correlated with high precision. There is, however, clear variance in other techniques used (classifiers, extraction methods, etc.) between the systems.

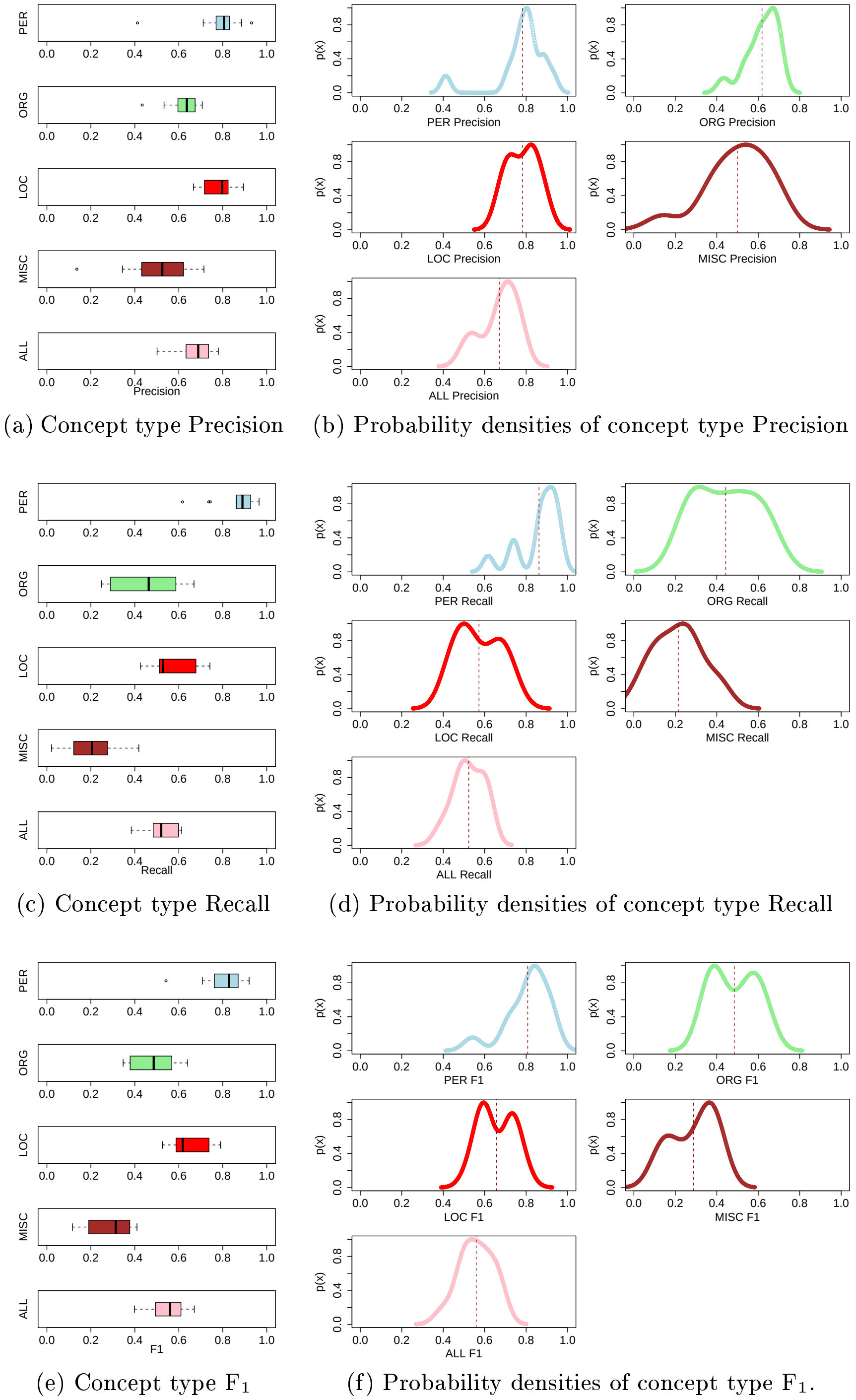
Fine-grained insight into the disparities between precision performance was obtained by inspecting the performance of the submissions across the different concept types (person, organisation, location, miscellaneous). Figure 3a presents the distribution of precision values across these four concept types and the macro average of these values. We find that systems do well (above the median of average precision values) for person and location concepts, and perform worse than the median for organisations and miscellaneous. For the entity type '*miscellaneous*', this is not surprising as it features a fairly *nuanced* definition, including films and movies, entertainment award events, political events, programming languages, sporting events and TV shows. We also note that several submissions used gazetteers in their systems, many of which were for locations; this could have contributed to the higher precision values for location concepts.

Table 2. Precision scores for each submission over the different concept types

Rank	Entry	PER	ORG	LOC	MISC	ALL
1	14 - 1	0.923	0.673	0.877	0.622	0.774
2	21 - 3	0.876	0.603	0.864	0.714	0.764
3	30 - 1	0.824	0.648	0.800	0.667	0.735
4	15 - 3	0.879	0.686	0.844	0.525	0.734
5	33 - 3	0.809	0.707	0.746	0.636	0.724
6	25 - 1	0.771	0.606	0.824	0.548	0.688
7	03 - 3	0.813	0.696	0.794	0.435	0.685
8	29 - 1	0.785	0.596	0.800	0.553	0.683
9	28 - 1	0.765	0.674	0.711	0.500	0.662
10	20 - 1	0.801	0.636	0.726	0.343	0.627
11	32 - 1	0.707	0.433	0.683	0.431	0.564
12	35 - 1	0.740	0.533	0.712	0.136	0.530
13	34 - 1	0.411	0.545	0.667	0.381	0.501

Recall. Although precision affords insight into the accuracy of the entities identified across different concept types, it does not allow for inspecting the detection rate over all possible entities. To facilitate this we also report the recall scores of each submission, providing an assessment of the entity coverage of each approach. Table 3 presents the overall recall values for each system and for each and across all concept types. Once again, as with precision, we note that hybrid systems (21, 15, 14) appear at the top of the rankings, with a rule-based approach (20) and a data driven approach (3) coming fourth and fifth respectively.

Looking at the distribution of recall scores across the submissions in Figure 3c we see a similar picture as before when inspecting the precision plots. For instance, for the person and location concepts we note that the submissions exceed the median of all concepts (when the macro-average of the recall scores is taken), while for organisation and miscellaneous lower values than the median are observed. This again comes back to the nuanced definition of the miscellaneous category, although the recall scores are higher on average than the precision score. The availability of person name and place name gazetteers also benefits identification of the corresponding concept types. This suggests that additional effort is needed to improve the *organisation* concept extraction and to provide information to seed the detection process, for instance through



(a) Concept type Precision

(b) Probability densities of concept type Precision

(c) Concept type Recall

(d) Probability densities of concept type Recall

(e) Concept type F_1

(f) Probability densities of concept type F_1.

Fig. 3. Distributions of performance scores for all submissions; dashed line is the mean.

the provision of organisation name gazetteers. Interestingly, when we look at the best performing system in terms of recall over the organisation concept we find that submission 14 uses a variety of third party lookup lists (Yago, Microsoft n-grams and Wordnet), suggesting that this approach leads to increased coverage and accuracy when extracting organisation names.

Table 3. Recall scores for each submission over the different concept types

Rank	Entry	PER	ORG	LOC	MISC	ALL
1	21 - 3	0.938	0.614	0.613	0.287	0.613
2	15 - 3	0.952	0.485	0.739	0.269	0.611
3	14 - 1	0.908	0.611	0.620	0.277	0.604
4	20 - 1	0.859	0.587	0.517	0.418	0.595
5	03 - 3	0.926	0.463	0.682	0.122	0.548
6	25 - 1	0.887	0.405	0.685	0.205	0.546
7	28 - 1	0.864	0.290	0.692	0.155	0.500
8	29 - 1	0.736	0.489	0.444	0.263	0.483
9	32 - 1	0.741	0.289	0.506	0.391	0.482
10	35 - 1	0.920	0.346	0.506	0.102	0.468
11	33 - 3	0.877	0.248	0.518	0.077	0.430
12	34 - 1	0.787	0.283	0.439	0.098	0.402
13	30 - 1	0.615	0.268	0.444	0.204	0.383

F-Measure (F_1). By combining the precision and recall scores together for the individual systems using the f-measure (F_1) score we are provided with an overall assessment of concept extraction performance. Table 4 presents the f-measure (F_1) score for each submission and performance across the four concept types. We note that, as previously, hybrid systems do best overall (top-3 places), indicating that a combination of rules and data-driven approaches yields the best results. Submission 14 records the highest overall F_1 score, and also the highest scores for the person and organisation concept types; submission 15 records the highest F_1 score for the location concept type; while submission 21 yields the highest F_1 score for the miscellaneous concept type. Submission 15 uses Google Gazetteers together with part-of-speech tagging of noun and verb phrases, suggesting that this combination yields promising results for our nuanced miscellaneous concept type.

Figure 3e shows the distribution of F_1 scores across the concept types for each submission. We find, as before, that the systems do well for person and location and poorly for organisation and miscellaneous. The reasons behind the reduced performance for these latter two concept types are, as mentioned, attributable to the availability of organisation information in third party lookup lists.

Table 4. F_1 scores achieved by each submission for each and across all concept types

Rank	Entry	PER	ORG	LOC	MISC	ALL
1	14 - 1	0.920	0.640	0.738	0.383	0.670
2	21 - 3	0.910	0.609	0.721	0.410	0.662
3	15 - 3	0.918	0.568	0.790	0.356	0.658
4	20 - 1	0.833	0.611	0.618	0.377	0.610
5	25 - 1	0.828	0.486	0.744	0.298	0.589
6	03 - 3	0.870	0.556	0.738	0.191	0.589
7	29 - 1	0.762	0.537	0.587	0.356	0.561
8	28 - 1	0.815	0.405	0.705	0.236	0.540
9	32 - 1	0.727	0.347	0.587	0.410	0.518
10	30 - 1	0.708	0.379	0.578	0.313	0.494
11	33 - 3	0.846	0.367	0.616	0.137	0.491
12	35 - 1	0.823	0.419	0.597	0.117	0.489
13	34 - 1	0.542	0.372	0.525	0.155	0.399

4 Conclusions

The aim of the MSM Concept Extraction Challenge was to foster an open initiative for extracting concepts from Microposts. Our motivation for hosting the challenge was born of the increased availability of third party extraction tools, and their widespread uptake, but the lack of an agreed formal evaluation of their accuracy when applied over Microposts, together with limited understanding of how performance differs between concept types. The challenge's task involved the identification of entity types and value tuples from a collection of Microposts. To our knowledge the entity annotation set of Microposts generated as a result of the challenge, and thanks to the collaboration of all the participants, is the largest annotation set of its type openly available online. We hope that this will provide the basis for future efforts in this field and lead to a standardised evaluation effort for concept extraction from Microposts.

The results from the challenge indicate that systems performed well which: (i) used a hybrid approach, consisting of data-driven and rule-based techniques; and (ii) exploited available lookup lists, such as place name and person name gazetteers, and linked data resources. Our future efforts in the area of concept extraction from Microposts will feature additional hosted challenges, with more complex tasks, aiming to identify the differences in performance between disparate systems and their approaches, and inform users of extraction tools on the suitability of different applications for different tasks and contexts.

5 Acknowledgments

We thank the participants who helped us improve the gold standard used for the challenge. We also thank eBay for supporting the challenge by sponsoring the prize for winning submission.

A.E. Cano is funded by the EPSRC project *violenceDet* (grant no. EP/J020427/1). A.-S. Dadzie was funded by the MRC project *Time to Change* (grant no. 129941).

References

1. H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Annual Meeting of the ACL*, 2002.
2. J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, 2005.
3. K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 42–47, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.
4. E. Loper and S. Bird. NLTK: The Natural Language Toolkit. In *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pages 62–69. Somerset, NJ: Association for Computational Linguistics, 2002.
5. P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. DBpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11, pages 1–8, 2011.
6. D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1), 2007.
7. R. Navigli and S. P. Ponzetto. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artif. Intell.*, 193, 2012.
8. L. Padró and E. Stanilovsky. Freeling 3.0: Towards wider multilinguality. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2473–2479, Istanbul, Turkey, May 2012. ACL Anthology Identifier: L12-1224.
9. L. Ratinov and D. Roth. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, CoNLL '09, 2009.
10. L.-A. Ratinov, D. Roth, D. Downey, and M. Anderson. Local and global algorithms for disambiguation to wikipedia. In *ACL*, 2011.
11. A. Ritter, S. Clark, Mausam, and O. Etzioni. Named entity recognition in tweets: An experimental study. In *EMNLP*, 2011.
12. G. Rizzo and R. Troncy. NERD: evaluating named entity recognition tools in the web of data. In *ISWC 2011, Workshop on Web Scale Knowledge Extraction (WEKEX'11), October 23-27, 2011, Bonn, Germany*, Bonn, GERMANY, 10 2011.

13. S. Sarawagi. Information extraction. *Foundations and Trends in Databases*, 1:261–377, 2008.
14. E. F. Tjong Kim Sang and F. De Meulder. 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*, CONLL '03, pages 142–147. Association for Computational Linguistics, 2003.
15. M. A. Yosef, J. Hoffart, I. Bordino, M. Spaniol, and G. Weikum. Aida: An online tool for accurate disambiguation of named entities in text and tables. *PVLDB*, 2011.

Challenge Submissions – Section I:

Concept Extraction Challenge:
University of Twente at #MSM2013

Mena B. Habib and Maurice van Keulen

Faculty of EEMCS, University of Twente, Enschede, The Netherlands
{m.b.habib,m.vankeulen}@ewi.utwente.nl

Abstract. Twitter messages are a potentially rich source of continuously and instantly updated information. Shortness and informality of such messages are challenges for Natural Language Processing tasks. In this paper we present a hybrid approach for Named Entity Extraction (NEE) and Classification (NEC) for tweets. The system uses the power of the Conditional Random Fields (CRF) and the Support Vector Machines (SVM) in a hybrid way to achieve better results. For named entity type classification we use AIDA [8] disambiguation system to disambiguate the extracted named entities and hence find their type.

1 Introduction

Twitter is an important source for continuously and instantly updated information. The huge number of tweets contains a large amount of unstructured information about users, locations, events, etc. Information Extraction (IE) is the research field which enables the use of such a vast amount of unstructured distributed information in a structured way. Named Entity Recognition (NER) is a subtask of IE that seeks to locate and classify atomic elements (mentions) in text belonging to predefined categories such as the names of persons, locations, etc. In this paper we split the NER task into two separate tasks: Named Entity Extraction (NEE) which aims only to detect entity mention boundaries in text; and Named Entity Classification (NEC) which assigns the extracted mention to its correct entity type. For NEE, we used a hybrid approach of CRF and SVM to achieve better results. For NEC, we first apply AIDA disambiguation system [8] to disambiguate the extracted named entities, then we use the Wikipedia categories of the disambiguated entities to find the type of the extracted mention.

2 Our Approach

2.1 Named Entity Extraction

For this task, we made use of two famous state of the art approaches for NER; CRF and SVM. We trained each of them in a different way as described below. The purpose of training is only for entity extraction rather recognition (extraction and classification). Results obtained from both are unionized to give the final extraction results.

Conditional Random Fields CRF is a probabilistic model that is widely used for NER [5]. Despite the successes of CRF, the standard training of CRF can be very expensive [6] due to the global normalization. In this task, we used an alternative method called *empirical training* [9] to train a CRF model. The maximum likelihood estimation (MLE) of the empirical training has a closed form solution, and it does not need iterative optimization and global normalization. So empirical training can be radically faster than the standard training. Furthermore, the MLE of the empirical training is also a MLE of the standard training. Hence it can obtain competitive precision to the standard training. Tweet text is tokenized using special tweets tokenizer [1]. For each token, the following features are extracted and used to train the CRF: (a) The Part of Speech (POS) tag of the word provided by a special POS tagger designed for tweets [1]. (b) If the word initial character is capitalized or not. (c) If the word characters are all capitalized or not.

Support Vector Machines SVM is a machine learning approach used for classification and regression problems. For our task, we used SVM to classify if a tweet segment is a named entity or not. The training process takes the following steps:

1. Tweet text is segmented using the segmentation approach as described in [4]. Each segment is considered a candidate for a named entity. We enriched the segments by looking up a Knowledge-Base (KB) (here we use YAGO [3]) for entity mentions as described in [2]. The purpose of this step is to achieve high recall. To improve the precision, we applied filtering hypotheses (such as removing segments that are composed of stop words or having verb POS).
2. For each tweet segment, we extract the following set of features in addition to those features mentioned in section 2.1: (a) The joint and the conditional probability of the segment obtained from Microsoft Web N-Gram services [7]. (b) The stickiness of the segment as described in [4]. (c) The segment frequency over around 5 million tweets [1]. (d) If the segment appears in WordNet. (e) If the segment appears as a mention in Yago KB. (f) AIDA disambiguation system score for the disambiguated entity of that segment (if any).
   The selection of the SVM features is based on the claim that disambiguation clues can help in deciding if the segment is a mention for an entity or not [2].
3. An SVM with RBF kernel is trained whether the candidate segment represents a mention of NE or not.

We take the union of the CRF and SVM results, after removing duplicate extractions, to get the final set of annotations. For overlapping extractions we select the entity that appears in Yago, then the one having longer length.

2.2 Named Entity Classification

The purpose of NEC is to assign the extracted mention to its correct entity type. For this task, we first use the prior type probability of the given mention in the training

[1] `http://wis.ewi.tudelft.nl/umap2011/` + TREC 2011 Microblog track collection.

Table 1: Extraction Results

	Pre.	Rec.	F1
Twiner Seg.	0.0997	0.8095	0.1775
Yago	0.1489	0.7612	0.2490
Twiner∪Yago	0.0993	0.8139	0.1771
Filter(Twiner∪Yago)	0.2007	0.8066	0.3214
SVM	0.7959	0.5512	0.6514
CRF	0.7157	0.7634	0.7387
CRF∪SVM	0.7166	0.7988	**0.7555**

Table 2: Extraction and Classification Results

	Pre.	Rec.	F1
CRF	0.6440	0.6324	0.6381
AIDA Disambiguation + Entity Categorization	0.6545	0.7296	**0.6900**

data. If the extracted mention is out of vocabulary (does not appear in training set), we apply AIDA disambiguation system on the extracted mentions. AIDA provides the most probable entity for the mention. We get the Wikipedia categories of that entity from the KB to form an entity profile. Similarly, we use the training data to build a profile of Wikipedia categories for each of the entity types (PER, ORG, LOC and MISC).

To find the type of the extracted mention, we measure the document similarity between the entity profile and the profiles of the 4 entity types. We assign the mention to the type of the most similar profile.

If the extracted mention is out of vocabulary and is not assigned to an entity by AIDA we try to disambiguate the first token of it. If all those methods failed to find entity type for the mention we just assign ”PER” type.

3 Experimental Results

In this section we show our experimental results of the proposed approaches on the training data. All our experiments are done through a 4-fold cross validation approach for training and testing. We used Precision, Recall and F1 measures as evaluation criteria for those results. Table 1 shows the NEE results along the extraction process phases. **Twiner Seg.** represents results of the tweet segmentation algorithm described in [4]. **Yago** represents results of the surface matching extraction as described in [2]. **Twiner∪Yago** represents results of merging the output of the two aforementioned methods. **Filter(Twiner∪Yago)** represents results after applying filtering hypothesis. The purpose of those steps is to achieve as much recall as possible with reasonable precision. **SVM** is trained as described in section 2.1 to find which of the segments represent true NE. **CRF** is trained and tested on tokenized tweets to extract any NE regardless of its type . **CRF∪SVM** is the unionized set of results of both **CRF** and **SVM**. Table 2 shows the final results of both extraction with **CRF∪SVM** and entity classification using the method presented in section 2.2 (**AIDA Disambiguation + Entity Categorization**). It also shows the **CRF** results when trained to recognize (extract and classify) NE. We considered it as our baseline. Our method of separating the extraction and classification outperforms the baseline.

4 Conclusion

In this paper, we present our approach for the IE challenge. We split the NER task into two separate tasks: NEE which aims only to detect entity mention boundaries in text; and NEC which assigns the extracted mention to its correct entity type. For NEE we used a hybrid approach of CRF and SVM to achieve better results. For NEC we used AIDA disambiguation system to disambiguate the extracted named entities and hence find their type.

References

1. K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-speech tagging for twitter: annotation, features, and experiments. In *Proc. of the 49th ACL conference*, HLT '11, pages 42–47, 2011.
2. M. B. Habib and M. van Keulen. Unsupervised improvement of named entity extraction in short informal context using disambiguation clues. In *Proc. of the Workshop on Semantic Web and Information Extraction (SWAIE)*, pages 1–10, 2012.
3. J. Hoffart, F. M. Suchanek, K. Berberich, E. L. Kelham, G. de Melo, and G. Weikum. Yago2: Exploring and querying world knowledge in time, space, context, and many languages. In *Proc. of WWW 2011*, 2011.
4. C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee. Twiner: named entity recognition in targeted twitter stream. In *Proc. of the 35th ACM SIGIR conference*, SIGIR '12, pages 721–730, 2012.
5. A. McCallum and W. Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proc. of the 7th HLT-NAACL conference*, CONLL '03, pages 188–191, 2003.
6. C. Sutton and A. McCallum. Piecewise training of undirected models. In *Proc. of UAI*, pages 568–575, 2005.
7. K. Wang, C. Thrasher, E. Viegas, X. Li, and B.-j. P. Hsu. An overview of microsoft web n-gram corpus and applications. In *Proc. of the NAACL HLT 2010*, pages 45–48, 2010.
8. M. A. Yosef, J. Hoffart, I. Bordino, M. Spaniol, and G. Weikum. Aida: An online tool for accurate disambiguation of named entities in text and tables. *PVLDB*, 4(12):1450–1453, 2011.
9. Z. Zhu, D. Hiemstra, P. M. G. Apers, and A. Wombacher. Closed form maximum likelihood estimator of conditional random fields. Technical Report TR-CTIT-13-03, University of Twente, 2013.

MSM2013 IE Challenge: Annotowatch

Stefan Dlugolinsky, Peter Krammer, Marek Ciglan, and Michal Laclavik

Institute of Informatics, Slovak Academy of Sciences
Dubravska cesta 9, 845 07 Bratislava, Slovak Republic
{stefan.dlugolinsky,peter.krammer,marek.ciglan,michal.laclavik}@savba.sk
http://ikt.ui.sav.sk

Abstract. In this paper, we describe our approach taken in the MSM2013 IE Challenge, which was aimed at concept extraction from microposts. The goal of the approach was to combine several existing NER tools which use different classification methods and benefit from their combination. Several NER tools have been chosen and individually evaluated on the challenge training set. We observed that some of these tools performed better on different entity types than other tools. In addition, different tools produced diverse results which brought a higher recall when combined than that of the best individual tool. As expected, the precision significantly decreased. The main challenge was in combining annotations extracted by diverse tools. Our approach was to exploit machine-learning methods. We have constructed feature vectors from the annotations yielded by different extraction tools and various text characteristics, and we have used several supervised classifiers to train the classification models. The results showed that several classification models have achieved better results than the best individual extractor.

Keywords: Information extraction, machine-learning, named entity recognition, microposts

1 Introduction

Most of the current Named Entity Recognition (NER) methods have been designed for concept extraction from relatively long and grammatically correct texts, such as newswire texts or biomedical texts. More and more user-generated content on the Web consists of a relatively short text which is often grammatically incorrect (e.g., microposts, on which these methods perform worse). The goal of the approach proposed in this paper is to combine several different information extraction methods in order to reach a more precise concept extraction on relatively short texts. We hypothesized that if these methods were combined properly, they would perform better than the best individual method from the pool. This assumption was partially proven through the evaluation of several available and well-known NER tools that use different entity extraction methods. The merged results of these tools showed a higher recall than that of the best tool but with a very low precision. The goal was to reduce or eliminate this tradeoff. Higher recall indicates that different methods complement each other and that there is room for improvement. We have tried various machine-learning algorithms and built several models capable of producing results based on concepts extracted by yielded tools. The goal was to produce a model with the highest possible precision approximating the recall measured for unified extracted concepts. In the following sections, we describe the NER tools that have been used and how they individually performed on the MSM2013 IE Challenge (from here on referenced as "challenge") training set (version 1.5). We briefly describe the methodology of our investigation (i.e., how our solution was built).

2 Tools Used

Our solution incorporates several available well-known NER tools: *Annie Named Entity Recognizer* [1], *Apache OpenNLP*[1], *Illinois Named Entity Tagger (with 4-label type model)* [2], *Illinois Wikifier* [3], *LingPipe (with English News - MUC-6 model)*[2], *Open Calais*[3], *Stanford Named Entity Recognizer (with 4 class caseless model)* [4], *WikipediaMiner*[4]. This list is complemented by the *Miscinator*, a tool specifically designed for the challenge. The Miscinator detects MISC concepts (i.e., entertainment/award event, sports event, movies, TV shows, political event, and programming languages). One of the tools' evaluation conclusions was that they were not performing well in detecting entertainment, award, and sports events. Therefore, we built a specialized gazetteer annotation tool for this task. The gazetteer has been constructed from the events annotations found in the challenge training set extended by Google Sets service (a method trained on web crawls) which generates list of items based on several examples. The only customization made to listed tools was the mapping of their annotation types to match target entity types (i.e., Location - LOC, Person - PER and Organization ORG) as well as filtering unimportant ones (e.g., Token). Relevant OpenCalais entities to target entities were similarly mapped. Illinois Wikifier was treated a bit differently, as it provided annotations with Wikipedia concepts and the yielded output did not comprise the type classification for the annotations. To overcome this drawback, we mapped the annotations to the DBPedia knowledge base and used DBPedia types associated with the given concepts to derive target entity types. WikipediaMiner annotations were mapped the same way.

3 Evaluation of Used Tools

All of the tools used were evaluated on the challenge training set. There were three ways of computing the Precision, Recall, and F_1 metrics used. The first method was *strict* (P_S, R_S and F_{1S}), which considered partially correct responses as incorrect; however, the second, *lenient*, considered them as correct (P_L, R_L

[1] `http://opennlp.apache.org`

[2] `http://alias-i.com/lingpipe`

[3] `http://www.opencalais.com/about`

[4] `http://wikipedia-miner.cms.waikato.ac.nz`

and F_{1L}). The third method was an *average* of the previous two (P_A, R_A, and F_{1A}). The evaluation results are shown in Fig. 1. We also evaluated the unified responses of all of the tools. Results showed that the recall was much higher ($R_S = 90\%$) than the best individual tool (Illinois NER got $R_S = 60\%$), but the precision was very poor ($P_S = 18\%$), hence the F_1 score ($F_{1S} = 30\%$). The best performing tool on microposts was OpenCalais, which scored $P_S = 70\%$, $R_S = 58\%$ and $F_{1S} = 64\%$.

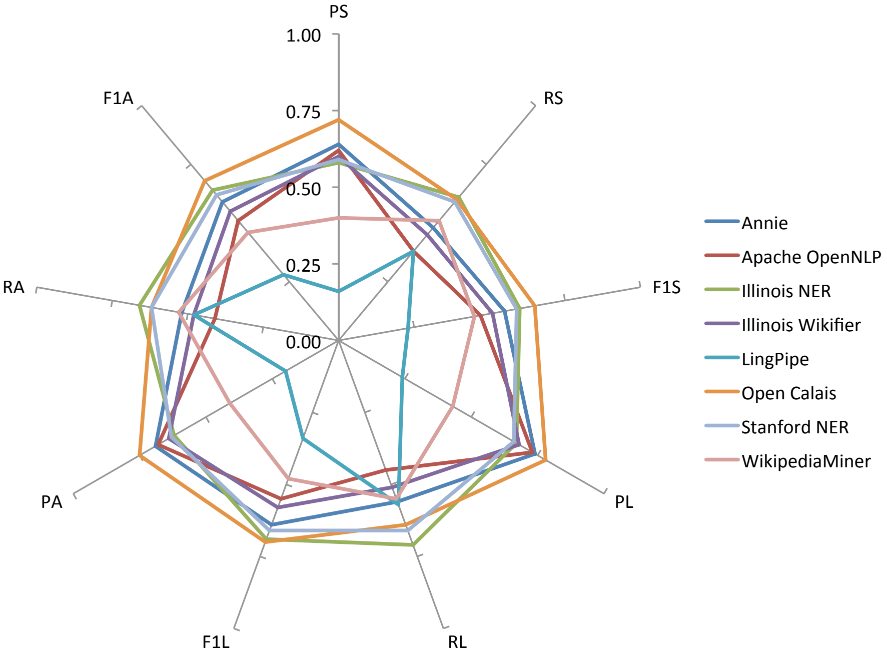


Fig. 1. Micro summary of NER tools over training set v1.5

4 Machine Learning

Our goal was to create a model that would take the most relevant results detected by each tool and perform better than the best tool did individually. We have used statistical classifiers to achieve this goal.

4.1 Input Features

We have taken the approach of describing how particular extractors performed on different entity types compared to the response of other extractors. Used as a training vector, this description was an input for training a classification model. A vector of input training features was generated for each annotation found by integrated NER tools. We called this annotation a reference annotation. The vector of each reference annotation consisted of several sub-vectors. The first sub-vector of the training vector was an annotation vector. The annotation vector described the reference annotation – whether it was uppercase or lowercase, used a capital first letter or capitalized all of its words, the word count, and the type of the detected annotation (LOC, MISC, ORG, PER, NP noun phrase, VP verb phrase, OTHER). The second sub-vector described microposts as a whole. It contained features describing whether all words longer than four characters were capitalized, uppercase, or lowercase. The rest of the sub-vectors were computed according to the overlap of the reference annotation with annotations produced by other NER tools. Such sub-vector (termed a method vector by us) was computed for each extractor and contained four other vectors (average scores per named entity type) for each target entity type (LOC, MISC, ORG, PER). The average score vector consisted of five components – *ail*: the average intersection length of a reference annotation with annotations produced by other extractors (from here on referenced as other's annotations), *aiia*: the average percentage intersection of other's annotations with reference annotation, *aiir*: the average percentage intersection of a reference annotation with other's annotations, *average confidence* (if the underlying extractors return such value), and *variance of the average confidence*. The last component in the training vector was the correct answer (i.e., the correct annotation type taken from manual annotation).

4.2 Model Training

Several types of classification models were considered, especially tree-models which allow the use of numerical and discrete attributes. Due to its large number of trees, Random Forests looked very advisable and reliable during the first round of testing. However, the increasing number of input attributes caused the performance of Random Forests to degrade. Therefore, we used a single decision tree generated by the C4.5 algorithm [5] as a simple alternative. The set of training vectors was preprocessed before the model training. Duplicate rows were removed from the training set and a randomize filter was applied to shuffle the training vectors. The preprocessed training set contained approximately $35,000$ vectors, each consisting of 105 attributes. The trained model was represented by a classification tree built by the J48 algorithm in Weka[5]. J48 is also known as an open-source implementation of the C4.5 algorithm with pruning. A Tenfold Fold Cross Validation was used. This model provided classification into five discrete classes (NULL, ORG, LOC, MISC, PER) for each record.

[5] http://www.cs.waikato.ac.nz/ml/weka/

4.3 Estimated Performance of the Model

To get an idea of our model performance, we have trained the model on an 80% split of the challenge training set cleaned from duplicate records and have evaluated it on the remaining 20% split. The evaluation results are displayed in Table 1. We included the results from the best individually performing tools for each entity type.

Table 1. Evaluation on the 20% training set split

	Illinois NER			Illinois Wikifier			Stanford NER			Miscinator			Annotowatch		
	P_S	R_S	F_{1S}	P_S	R_S	F_{1S}	P_S	R_S	F_{1S}	P_S	R_S	F_{1S}	P_S	R_S	F_{1S}
LOC	54%	56%	55%	36%	44%	40%	55%	54%	55%	-	-	-	57%	56%	57%
MISC	4%	7%	5%	10%	18%	13%	2%	2%	2%	87%	44%	59%	55%	58%	57%
ORG	31%	35%	33%	60%	41%	49%	23%	28%	25%	-	-	-	64%	49%	56%
PER	86%	84%	85%	89%	56%	69%	83%	78%	81%	-	-	-	85%	87%	86%
ALL	62%	66%	64%	63%	49%	55%	60%	60%	60%	87%	4%	7%	77%	75%	76%

5 Runs Submitted

Three runs were submitted for evaluation in the challenge. The first run was generated by the C4.5 algorithm trained model with parameter M denoting the minimum number of instances per leaf set to 2. The second run was generated by the model trained with parameter M set to 3. The third run was based on the first run and involved specific post-processing. If a micropost identical to one in the training set was annotated, we extended the detected concepts by those from manually annotated training data (affecting three microposts). A gazetteer built from a list of organizations found in the training set has been used to extend the ORG annotations of the model (affecting 69 microposts). The models producing the submission results were trained on a full challenge training set.

Acknowledgments. This work is supported by projects VEGA 2/0185/13, VENIS FP7-284984, CLAN APVV-0809-11 and ITMS: 26240220072.

References

1. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02). (2002)

2. Ratinov, L., Roth, D.: Design challenges and misconceptions in named entity recognition. In: Proceedings of the Thirteenth Conference on Computational Natural Language Learning. CoNLL '09, Stroudsburg, PA, USA, Association for Computational Linguistics (2009) 147–155
3. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to wikipedia. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1. HLT '11, Stroudsburg, PA, USA, Association for Computational Linguistics (2011) 1375–1384
4. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by gibbs sampling. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. ACL '05, Stroudsburg, PA, USA, Association for Computational Linguistics (2005) 363–370
5. Quinlan, J.R.: C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1993)

Learning with the Web: Spotting Named Entities on the intersection of NERD and Machine Learning

Marieke van Erp[1], Giuseppe Rizzo[2], Raphaël Troncy[2]

[1] VU University Amsterdam, The Netherlands
marieke.van.erp@vu.nl
[2] EURECOM, Sophia Antipolis, France,
giuseppe.rizzo@eurecom.fr, raphael.troncy@eurecom.fr

Abstract. Microposts shared on social platforms instantaneously report facts, opinions or emotions. In these posts, entities are often used but they are continuously changing depending on what is currently trending. In such a scenario, recognising these named entities is a challenging task, for which off-the-shelf approaches are not well equipped. We propose NERD-ML, an approach that unifies the benefits of a crowd entity recognizer through Web entity extractors combined with the linguistic strengths of a machine learning classifier.

Keywords: Named Entity Recognition, NERD, Machine Learning

1 Introduction

Microposts are a highly popular medium to share facts, opinions or emotions. They promise great potential for researchers and companies alike to tap into a vast wealth of a heterogeneous and instantaneous barometer of what is currently trending in the world. However, due to their brief and fleeting nature, microposts provide a challenging playground for text analysis tools that are oftentimes tuned to longer and more stable texts. We present an approach that attempts to leverage this problem by employing an hybrid approach that unifies the benefits of a crowd entity recogniser through Web entity extractors combined with the linguistic strengths of a machine learning classifier.

2 The NERD-ML System

In our approach, we combine a mix of NER systems in order to deal with the brief and fleeting nature of microposts. The three main modules of our approach are: NERD, Ritter et al.'s system, and Stanford NER. NERD [4] is used to spot entities using a variety of Web extractors. The strength of this approach lies in the fact that these systems have access to large knowledge bases of entities such as DBpedia[3] and Freebase[4]. Ritter et al. [3] propose a tailored approach for entity recognition based on a previously annotated Twitter stream; while Stanford NER [1] represents the state of the art in the entity recognition, providing off-the-shelf or customisable NER using a machine learning algorithm. While NERD and Ritter et al.'s approach are used as off-the-shelf extractors, Stanford NER is trained on the MSM training dataset. The outputs of these systems are used as features for NERD-ML's final machine learning module. We have also added extra features based on the token and the micropost format to further aid the system. The generated feature sets can be fed into any machine learning algorithm in order to learn the optimal extractor/feature combination. An overview of our system is shown in Figure 1. In the remainder of this section we explain the components.

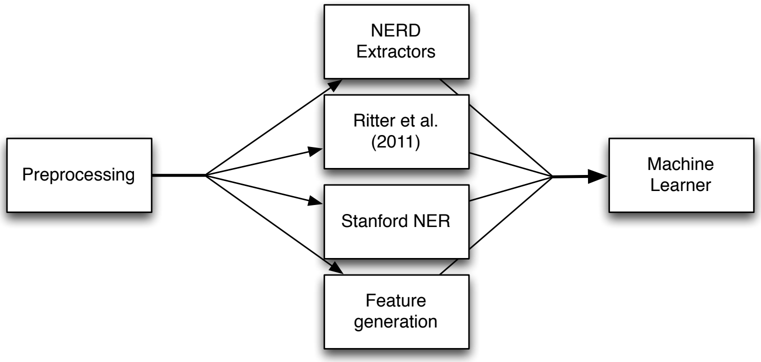


Fig. 1: Overview of the NERD-ML System

Preprocessing: In the preprocessing phase, the data is formatted to comply with the input format of our extractors. For ease of use, the dataset is converted to the CoNLL IOB format [5]. Furthermore, posts from the MSM2013 training data are divided randomly over 10 parts in order to a) be able to perform a 10-fold cross-validation experiment and b) comply with NERD filesize limitations.
NERD Extractors: Each of the data parts is sent to the NERD API to retrieve named entities from the following extractors: AlchemyAPI, DBpedia Spotlight (setting: *confidence=0*, *support=0*, *spotter=CoOccurrenceBasedSelector*), Extractiv, Lupedia, OpenCalais, Saplo, TextRazor, Wikimeta, Yahoo and Zemanta (setting: *markup_limit=10*). The NERD ontology consists of 75 classes, which are mapped to the four classes of the MSM2013 challenge.
Ritter et al. 2011: The off-the-shelf approach as described in [3] is taken both as baseline and input for the hybrid classifier. The 10 entity classes are mapped to the four classes of the MSM2013 challenge.

[3] `http://www.dbpedia.org`
[4] `http://www.freebase.com`

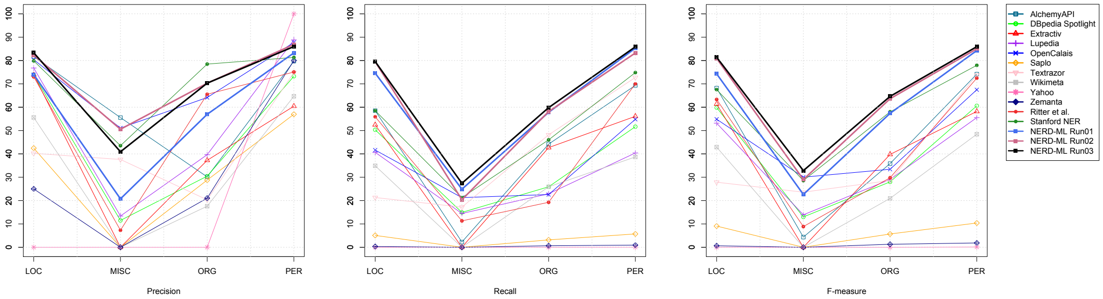


Fig. 2: Results of individual and combined extractors in 10-fold cross validation experiments

Stanford NER: The Stanford NER system (version 1.2.7) is retrained on the MSM2013 data challenge set, using parameters based on the properties file english.conll.4class.distsim.crf.ser.gz provided with the Stanford distribution. The Stanford results serve as a baseline, as well as input for the hybrid classifier.
Feature Generation: To aid the classifier in making sense of the structure of the microposts, we added 8 additional features to the dataset inspired by the features described in [3]. We implemented the following features: capitalisation information (initial capital, allcaps, proportion of tokens capitals in the micropost), prefix (first three letters of the token), suffix (last three letters of the token), whether the token is at the beginning or end of the micropost, and part-of-speech token using the TwitterNLP tool and POS-tagset from [2].
NERD-ML: The output generated by the NERD extractors, Ritter et al.'s system, Stanford NER system and the added features are used to create feature vectors. The feature vectors serve as input to a machine learning algorithm in order to find combinations of features and extractor outputs that improve the scores of the individual extractors. We experimented with several different algorithms and machine learning settings using WEKA-3.6.9[5].

3 Results

In Figure 2, the results of the individual NER components and the hybrid NERD-ML system are presented. The first run is a baseline run that includes the full feature set. The second run only includes the extractors and no extra features. The third run uses a smaller feature set that was compiled through automatic feature selection. The settings of the three runs of the hybrid NERD-ML system are:
Run 1: All features, k-NN, k=1, Euclidean distance, 10-fold cross validation

[5] `http://www.cs.waikato.ac.nz/ml/weka`

Run 2: AlchemyAPI, DBpedia Spotlight, Extractiv, Lupedia, OpenCalais, Saplo, Yahoo, Textrazor, Wikimeta, and Zemanta, Stanford NER, Ritter et al., SMO, standard parameters, 10-fold cross validation

Run 3: POS, Initial Capital, Suffix, Proportion of Capitals, AlchemyAPI, DBpedia Spotlight, Extractiv, Opencalais, Textrazor,Wikimeta, Stanford NER, Ritter et al., SMO, standard parameters, 10-fold cross validation

Results are computed using the conlleval script and plotted using R. All settings and scripts are publicly available[6].

4 Conclusions

Extracting named entities from microposts is a difficult task due to the ever-changing nature of the data, breadth of topics discussed and linguistic inconsistencies it contains. Our experiments with NERD-ML show that the combination of different NER systems outperforms off-the-shelf approaches, as well as the customised Stanford approach. Our results indicate that an hybrid system may be better equipped to deal with the task of identifying entities in microposts, but care must be taken in combining features and extractor outputs.

Acknowledgments

This work was partially supported by the European Union's 7th Framework Programme via the projects LinkedTV (GA 287911).

References

1. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by gibbs sampling. In: 43^{nd} Annual Meeting of the Association for Computational Linguistics (ACL'05). Ann Arbor, MI, USA (June 2005)
2. Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., Smith, N.A.: Improved part-of-speech tagging for online conversational text with word clusters. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2013). Atlanta, GA, USA (June 2013)
3. Ritter, A., Clark, S., Mausam, Etzioni, O.: Named entity recognition in tweets: An experimental study. In: Empirical Methods in Natural Language Processing (EMNLP'11). Edinburgh, UK (July 2011)
4. Rizzo, G., Troncy, R.: NERD: A Framework for Unifying Named Entity Recognition and Disambiguation Extraction Tools. In: 13^{th} Conference of the European Chapter of the Association for computational Linguistics (EACL'12). Avignon, France (April 2012)
5. Tjong Kim Sang, E.F.: Introduction to the conll-2002 shared task: Language-independent named entity recognition. In: Conference on Computational Natural Language Learning (CoNLL'02). Taipei, Taiwan (Aug-Sept 2002)

[6] https://github.com/giusepperizzo/nerdml

ACE: A Concept Extraction Approach using Linked Open Data

Keith Cortis

Digital Enterprise Research Institute, National University of Ireland, Galway, Ireland
keith.cortis@deri.org

Abstract. Given the increase in popularity of several social networks, numerous users tend to express themselves or reach out to their followers via online posts, normally in the form of microposts. Dealing with such data of short textual content can be quite intricate due to several factors such as misspellings, slang, emoticons, etc. In this paper we present an approach towards extracting several concepts from microposts, where the main challenge is to classify them into specific entity types. This will help in discovering knowledge from possible semi-structured/unstructured data after taking into account several factors. In our approach we extend a state-of-the-art information extraction system which we call ACE, and make use of a dataset that is part of the Linked Open Data cloud, in order to improve the named entity extraction process.

Keywords: Microposts, Natural Language Processing, Named Entity Recognition, Linked Open Data

1 Introduction

Dealing with online microposts which are made up of short textual content as posted on the Web such as Twitter status updates (up to 140 characters), Facebook tagged photos and Foursquare/Facebook check-ins, can be quite intricate due to several factors. Such factors amount from misspellings, incomplete content, slang, jargon and incorrect acronyms and/or abbreviations, to emoticons and content misinterpretation. Some of these issues can also be attributed to the nature of short textual content limit of a micropost, which at times forces a user to resort to using short words such as acronyms and slang, in order to make a statement. Therefore, the main challenge is that of extracting any possible concepts from micropost data, before classifying them into specific entity types e.g. Location, Organisation, Person. This will enable knowledge discovery from semi-structured/unstructured data, which can be modelled against specific standards and used for several tasks e.g. user modelling, user profiling techniques, social navigation and recommender systems. Besides microposts, concept extraction can also be applicable to other forms of short textual content, such as ebay selling item titles and customer reviews, which allow up to 80 characters.

· **#MSM2013 Workshop Concept Extraction Challenge Proceedings** ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

Our main focus was that of coming up with a novel solution by using a tool that can be extended together with Linked Open Data (LOD), in order to improve the entity concept (type and value) extraction from microposts. In ACE[1], we extend the ANNIE Information Extraction (IE) system [1], a plugin in the General Architecture for Text Engineering (GATE)[2] tool, since it can be customised according to a user's specific needs. ANNIE contains the following main processing resources for common NLP tasks: document reset, English tokeniser, gazetteer, sentence splitter, Part-of-Speech tagger, named entity (NE) transducer (semantic tagger) and orthomatcher. The DBPedia[3] dataset which is part of the LOD cloud[4] is also used, in order to generate or retrieve more concepts for some entities. The reason behind this choice is that the ANNIE gazetteers are limited to specific entity values and thus, it is beneficial that they are trained on manually annotated datasets, remote datasets, or both. Such an approach is expected to enhance the Named Entity Recognition (NER) techniques of the ANNIE IE system.

2 Concept Extraction Approach

Our concept extraction approach, involves three different process, as outlined in the sub-sections below.

2.1 Entity Concept Training

The first part of the approach involved the extraction of 3191 concepts (2103 without duplicates) from the 2815 microposts that made up the challenge training data. After the extraction was complete, we classified each unique concept to its respective entity and created a gazetteer for each of the four entity types of the challenge i.e. Person (PER), Location (LOC), Organisation (ORG) and Miscellaneous (MISC). These were added to the list of ANNIE gazetteers–and classified to their specific entity type (PER, LOC, ORG), while a new entity type was created for the MISC concepts–in order to enhance the system's training data for the NER process. The statistics for each extracted entity concept can be found within Table 1.

Table 1. Training data concept statistics

	PER	**LOC**	**ORG**	**MISC**
Total concepts	1721	621	618	231
Unique concepts	1199	360	351	193
Duplicate concepts	522	261	267	38

[1] *A*NNIE extension for *C*oncept *E*xtraction

[2] http://gate.ac.uk/

[3] http://dbpedia.org/

[4] http://lod-cloud.net/

2.2 ANNIE Extension

The ANNIE IE system was further extended with the six entity types that define the challenge MISC entity i.e. Film/Movie (F), Entertainment Award Event (EAE), Political Event (PE), Programming Language (PL), Sporting Event (SE) and TV Show (TVS). The existing Person and Location entities were also partly extended to recognise: multiple names/surnames and full person names with prefixes and suffixes (e.g., Dr. Joe Smith-Jones Jr.), for the former; and more postcodes for some major countries, and more complete street structures for the latter. The semantic tagger processing resource (PR) within the ANNIE pipeline—responsible for processing the outputs of any annotated entities—was extended through Java Annotation Patterns Engine (JAPE)[5] rules which are based on regular expressions. Several pattern/action rules were implemented for defining of the EAE, PE, SE and TVS named entities.

The pattern/action rules for the PE entity were based the Wikipedia Political Events structure[6], where the most common forms of events were highlighted from the existing subcategories. A gazetteer list of common political key terms such as general election, congress, debate, etc., was also added to the list of ANNIE gazetteers in order for the rules to be able to recognise the context around any key term that may be referring to a PE (e.g., South African general election, 6th congress of the Communist Party of China, Ireland Constitutional Convention 2012). Following some analysis, the EAE entity was also based on the most common and popular structure of the Entertainment award names within the Wikipedia Awards category[7].

A gazetteer list of common EAE key terms such as award, prize, festival, etc., was also added to the list of ANNIE gazetteers in order for the rules to identify the context around any key term that may be referring to an EAE (e.g., New York International Film Festival, Galway Prize 2012). The SE and TVS entities were also extended. A similar approach to the entities described above was adopted for both, together with a newly created gazetteer listing all kinds of sports for the former. Sporting Events (e.g., Galway Football cup 2012, John Doe tennis open) and TV Shows (e.g., The John Doe show, John Doe's program) will be recognised according to the implemented pattern/action rules.

DBPedia was used to retrieve more concepts for some entities. This dataset was chosen because it is constantly updated from Wikipedia, and is a reliable source for named entities. Several gazetteers were created from DBPedia in order to enhance the existing City (Ci), Country (Co), and Organisation (Org) ANNIE gazetteers, whereas new ones were created for the F and PL entities, together with the other four entities that were extended above. The mentioned gazetteers were populated directly from DBPedia through a SPARQL query by means of the Large KB Gazetteer[8], which is a PR within GATE that is used for loading a particular ontology from RDF. Every lookup annotation within each

[5] http://gate.ac.uk/sale/tao/splitch8.html#x12-2060008

[6] http://en.wikipedia.org/wiki/Category:Political_events

[7] http://en.wikipedia.org/wiki/List_of_prizes,_medals,_and_awards#Entertainment

[8] http://gate.ac.uk/sale/tao/splitch13.html#sec:gazetteers:lkb-gazetteer

imported gazetteer has a reference to the instance and its respective DBPedia class URI. Tests and analysis on both 'foaf:name' and 'rdfs:label' (English language) properties for each named entity were conducted prior to the gazetteer creation process, were the values of the property containing the most accurate and/or highest number of instances were extracted for each. For the known named entities, given that DBPedia contains 573,000 places, we decided to extract the 'Country' instances that do not have a dissolution year for the Co entity. On the other hand, the 'City' class was not chosen for the Ci entity, since it only contains instances of large urban settlements. Therefore, we opted for settlements having a population greater than 5599, due to the limit of triples per query that can be obtained from the DBPedia SPARQL endpoint. Similarly for the Org entity, DBPedia contains around 192,000 Organisations, therefore we extracted separate gazetteers for the most important types. The amount of DBPedia instances extracted for each named entity is recorded in Table 2.

Table 2. DBPedia entity concepts

Named Entity	DBPedia Class	#Instances
Co	Country	3910
Ci	Settlement	51796
Org	EducationalInstitution	48483
	PoliticalParty	5470
	TradeUnion	2144
	GovernmentAgency	3265
	MilitaryUnit	17397
	Company	48481
	Broadcaster	28412
	Non-ProfitOrganisation	3020
F	Film	52214
EAE	Award	1871
PE	Election	4556
PL	ProgrammingLanguage	491
SE	SportsEvent	6653
TVS	TelevisionShow	25114

2.3 Entity Concept Extraction

The entity concept extraction process is made up of two consecutive steps:

1. The challenge test data made up of 1526 microposts was cleaned from any common social media slang and emoticons, followed by
2. NER which is then performed on each cleaned micropost through the extended ANNIE IE system in order to find out all possible entity concepts.

All entity concepts that are either a stop word, number or single character, were not annotated due to precision reasons. Even though there might have been some true positives, we favoured a cautious approach, to lower the number of extracted false positives. We used the challenge training data to test ACE, where the average F_1 score achieved across the four entities was that of 0.743. All the results obtained for each entity can be seen within Table 3 below.

Table 3. Training data concept extraction results

	PER	**LOC**	**ORG**	**MISC**
Precision	0.886	0.891	0.723	0.218
Recall	0.918	0.923	0.94	0.883
F_1 *score*	0.901	0.907	0.817	0.35

References

1. H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, 2002.

Leveraging Existing Tools for Named Entity Recognition in Microposts

Fréderic Godin[†], Pedro Debevere[†], Erik Mannens[†],
Wesley De Neve[†*], and Rik Van de Walle[†]

[†] Multimedia Lab, Ghent University - iMinds, Ghent, Belgium
[*] Image and Video Systems Lab, KAIST, Daejeon, South Korea
{frederic.godin,pedro.debevere,erik.mannens,
wesley.deneve,rik.vandewalle}@ugent.be

Abstract. With the increasing popularity of microblogging services, new research challenges arise in the area of text processing. In this paper, we hypothesize that already existing services for Named Entity Recognition (NER), or a combination thereof, perform well on microposts, despite the fact that these NER services have been developed for processing long-form text documents that are well-structured and well-spelled. We test our hypothesis by applying four already existing NER services to the set of microposts of the MSM2013 IE Challenge.

Keywords: microposts, NER, text processing

1 Introduction

Research in the domain of text processing has traditionally focused on analyzing long-form text documents that are well-structured and well-spelled [1]. However, thanks to the high popularity of microblogging sites, research in the domain of text processing is increasingly paying attention to the analysis of microposts as well. Microposts are short-form text fragments that are typically noisy in nature, hereby lacking structure and often containing a substantial amount of slang and misspelled words, frequently in multiple languages. In this paper, we hypothesize that already existing services for Named Entity Recognition (NER), as often used for processing news corpora, perform well on microposts, even without preprocessing, and that future research efforts should regard these NER services as a strong baseline.

2 Evaluation of existing services

Current NER services are tailored to processing long-form text documents that are typically well-structured and well-spelled. Rizzo *et al.* [2] quantitatively evaluated six NER web services on three types of corpora: 5 TED talks, 1000 news articles of the New York Times, and 217 WWW conference abstracts. In this paper, we aim at complementing this evaluation by testing the effectiveness of these services on a fourth fundamentally different text corpus, namely the microposts of the MSM2013 IE Challenge. Because both Evri and Extractiv are no longer available, we had to limit ourselves to the testing of four services, namely AlchemyAPI[1], DBpedia Spotlight[2], OpenCalais[3], and Zemanta[4].

To test the effectiveness of the aforementioned services, we did not apply any type of preprocessing. Given the MSM2013 IE Challenge guidelines, we evaluated the recognition of four types of entities: persons, locations, organizations, and a set of miscellaneous entities. The miscellaneous category contains the following entities: movies, entertainment award events, political events, programming languages, sporting events, and TV shows.

Given that the services evaluated make use of ontologies that are much more elaborate, we mapped the service ontologies to the four entity types. We evaluated a total of 2813 microposts of the training set. We left out microposts 583 and 781 because OpenCalais could not handle them. Because we used an ontology mapping, our results can differ with other evaluations. We report our results in Table 1.

Table 1. Evaluation of four different services: AlchemyAPI (A), DBpedia Spotlight (S), OpenCalais (O), and Zemanta (Z). For DBpedia Spotlight, we evaluated two configurations: confidence=0.2 and confidence=0.5.

	PER			LOC			ORG			MISC		
	Pr	*Re*	F_1	*Pr*	*Re*	F_1	*Pr*	*Re*	F_1	*Pr*	*Re*	F_1
A	81.1%	75.6%	**78.2%**	81.2%	69.0%	**74.6%**	59.5%	50.2%	54.4%	54.2%	5.6%	10.2%
S (0.2)	54.6%	61.0%	57.6%	44.8%	48.1%	46.4%	16.1%	49.7%	24.4%	2.7%	40.7%	5.0%
S (0.5)	87.0%	20.3%	32.9%	54.5%	1.9%	3.7%	19.7%	3.9%	6.5%	5.8%	10.0%	7.3%
O	71.7%	67.2%	69.3%	81.8%	66.1%	73.1%	72.2%	45.5%	**55.8%**	46.2%	23.8%	**31.4%**
Z	91.0%	57.4%	70.4%	83.9%	52.1%	64.3%	71.9%	36.1%	48.1%	37.1%	24.2%	29.3%

Table 2. Evaluation of the Random Forest (RF)-based model for predicting entity types, using 10-fold cross validation. Dependent on the DBpedia Spotlight results obtained, we evaluated two configurations.

	PER			LOC			ORG			MISC		
	Pr	*Re*	F_1	*Pr*	*Re*	F_1	*Pr*	*Re*	F_1	*Pr*	*Re*	F_1
RF (0.2)	78.4%	86.3%	**82.2%**	80.9%	71.1%	**75.7%**	62.8%	58.1%	**60.4%**	62.0%	38.3%	**47.4%**
RF (0.5)	75.0%	89.5%	81.6%	81.7%	68.2%	74.3%	71.9%	50.6%	59.4%	62.2%	30.0%	40.5%

[1] http://www.alchemyapi.com/

[2] http://dbpedia.org/spotlight/

[3] http://www.opencalais.com/

[4] http://www.zemanta.com/

As highlighted in bold, AlchemyAPI outperforms the other three services in identifying persons and is a close first in recognizing locations. On the other hand, OpenCalais performs best in recognizing organizations and MISC entities. Although Zemanta never wins, this service is characterized by a high precision. DBpedia Spotlight performs poorly because it returns an extensive list of possible entity types that often adhere to all four categories, instead of returning a single entity type.

When zooming in on the individual results, we can notice that AlchemyAPI performs bad in recognizing exotic names, small villages and buildings (e.g., St. Georges Mill), and recognizing abbreviations of organizations (e.g., DFID and UKGov). Furthermore, AlchemyAPI performs poorly in recognizing well-known events and TV shows such as “Super Bowl” and “Baywatch”. Zemanta suffers from similar problems. However, Zemanta performs worse than AlchemyAPI because it is more dependent on the usage of capital letters (e.g., Uruguay - uruguay and URUGUAY). We can observe similar behavior for OpenCalais and AlchemyAPI, for recognizing locations and organizations. OpenCalais is also capable of recognizing well-known events like the Super Bowl. When the confidence is set high (0.5), a lot of well-known entities cannot be recognized by DBpedia Spotlight, such as “Katy Perry”. When the confidence is set low (0.2), “Katy Perry” is recognized but a lot of noise is recognized as a person too (e.g., love, follow, guy).

3 Combining existing services

To further improve the results of NER on the training set, we combined the outputs of the different services. E.g., one can imagine that it is more plausible that a word is an entity when multiple services claim this with high confidence than when only one service claims this with low confidence. For each of the recognized entities, we constructed a feature vector and classified it using the technique of Random Forest. The goal was to predict one of the four entity types. For each service, our feature vector contained an element referencing one of the four challenge entity types, the original entity type according to the service ontology used, and a confidence and/or relevance value. In the case of DBpedia Spotlight, we omitted the original entity type element because this element was too sparse. We created a negative set by making use of the entities that were recognized by the services, but that were not in the training set.

We evaluated our set of feature vectors by means of the Weka toolkit. We applied 10-fold cross validation. We made use of two sets: the first set contained the DBpedia Spotlight results when querying this service with a confidence of 0.2, whereas the second set contained the DBpedia Spotlight results when querying this service with a confidence of 0.5. We applied Random Forest with 20 trees and four attributes per tree. We report the results of our evaluation in Table 2.

We highlighted the best results of our Random Forest-based fusion approach in bold for categorizing entity types. When we make use of the entities recognized by DBpedia Spotlight with a low confidence as part of the feature vector, the use of Random Forest leads to better results than when making use of high-confidence DBpedia Spotlight results. Applying Random Forest on noisy data with low precision and recall values yields significant improvements. Especially in the MISC category where we obtained an improvement of almost 7%. (Note: The result in Table 1 and 2 cannot be compared directly because the evaluation was conducted in a different way. In Table 1, this was on a word-by-word basis. In Table 2, this was on a entity type-by-type basis.)

The next step is to make use of this categorization approach to decide whether we should trust the combined result of the different services for recognizing a certain named entity type. The final evaluation of the proposed algorithm is part of the Making Sense of Micropost Challenge 2013 and was conducted on the test set. The results were presented at the workshop itself and were therefore not available yet at the time of writing.

4 Conclusions

In this paper, we have shown that existing NER services can recognize named entities in microposts with high F_1 values, especially when aiming at the recognition of persons and locations. In addition, we have demonstrated how the results of several services can be combined with the goal of achieving a higher precision. We can conclude that already existing NER services make for a strong baseline when aiming at the design and testing of new NER algorithms for microposts.

5 Acknowledgments

The research activities in this paper were funded by Ghent University, iMinds, the Institute for Promotion of Innovation by Science and Technology in Flanders (IWT), the FWO-Flanders, and the European Union.

References

1. M. W. Berry and J. Kogan, editors. *Text Mining: Applications and Theory.* Wiley, Chichester, UK, 2010.
2. G. Rizzo, R. Troncy, S. Hellmann, and M. Bruemmer. NERD meets NIF: Lifting NLP extraction results to the linked data cloud. In *LDOW 2012, 5th Workshop on Linked Data on the Web*, 2012.

Memory-based Named Entity Recognition in Tweets

Antal van den Bosch[1] and Toine Bogers[2]

[1] Centre for Language Studies
Radboud University Nijmegen
NL-6200 HD Nijmegen, The Netherlands
a.vandenbosch@let.ru.nl
[2] Royal School of Library Information Science
Birketinget 6, DK-2300
Copenhagen, Denmark
tb@iva.dk

Abstract. We present a memory-based named entity recognition system that participated in the MSM-2013 Concept Extraction Challenge. The system expands the training set of annotated tweets with part-of-speech tags and seedlist information, and then generates a sequential memory-based tagger comprised of separate modules for known and unknown words. Two taggers are trained: one on the original capitalized data, and one on a lowercased version of the training data. The intersection of named entities in the predictions of the two taggers is kept as the final output.

1 Background

Named-entity recognition can be seen as a labeled chunking task, where all beginning and ending words of names of predefined entity categories should be correctly identified, and the category of the entity needs to be established. A well-known solution to this task is to cast it as a token-level tagging task using the IOB or BIO coding scheme [1]. Preferably, a structured learning approach is used which combines accurate token-level decisions with a more global notion of likely and syntactically correct output sequences.

Memory-based tagging [2] is a generic machine-learning-based solution to structured sequence processing that is applicable to IOB-coded chunking. The algorithm has been implemented in MBT, an open source software package.[3] MBT generates a sequential tagger that tags from left to right, taking its own previous tagging decisions into account when generating a next tag. MBT operates on two classifiers. First, the 'known words' tagger handles words in test data which it has already seen in training data, and of which it knows the potential tags. Second, the 'unknown words' tagger is invoked to tag words not seen

[3] MBT is available in Debian Science: Linguistics, http://blends.alioth.debian.org/science/tasks/linguistics and at http://ilk.uvt.nl/mbt. The software is documented in [3].

· **#MSM2013 Workshop Concept Extraction Challenge Proceedings** ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

during training. Instead of the word itself it takes into account character-based features of the word, such as the last three letters and whether it is capitalized or not [2].

Named entity recognition in social media microtexts such as Twitter messages, tweets, is generally approached with regular methods, but it is also generally acknowledged that language use in tweets deviates from average written language use in various aspects: it features more spelling and capitalization variants than usual, and it may mention a larger variety of people, places and organizations than, for instance, news. Most studies report relatively low scores because of these factors [4–6].

2 System Architecture

Figure 1 displays a schematic overview of the architecture of our system. A new incoming tweet is first enriched by seed list information, that for each token in the tweet checks whether it occurs as a geographical name, or as part of a person or organization name in gazetteer lists for these three types of entities. This produces a token-level code that is either empty (`-`) or any combination of letters representing occurrence in a person name list (`P`), a geographical name list (`G`), or an organizational name list (`O`). We provide details on the resources we used in our system in Section 3. The tweet is also part-of-speech tagged by a memory-based tagger trained on the Wall Street Journal part of the Penn Treebank [7], producing Penn Treebank part-of-speech tags for all tokens at an estimated accuracy of 95.9%.

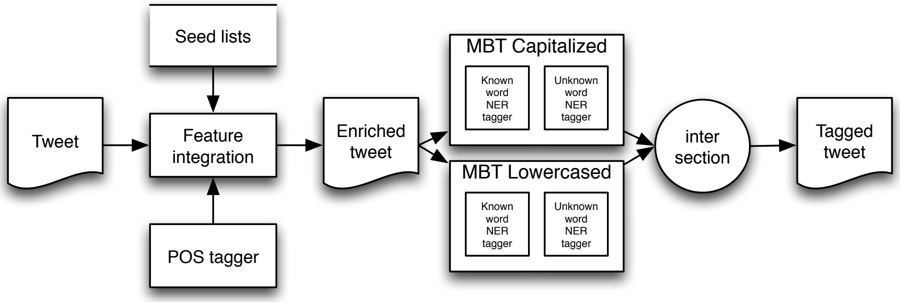


Fig. 1. The architecture of our system.

The enriched tweet is then processed by two MBT taggers. The first tagger is trained on the original training data with all capitalization information intact; the second tagger is trained on a lowercased version of the training set. The taggers both assign BIO-tags to the tokens constituting named-entity chunks [1].

The two MBT modules generate partly overlapping predictions. Only the named entity chunks that are fully identical in the output of the two modules, i.e. their intersection, are kept. The result is a tweet annotated with named entity chunks.

3 Resources

The MBT modules are trained on the official (version 1.5) training data provided for the MSM-2013 Concept Extraction Challenge.[4], complemented with the training and testing data of the CoNLL-2003 Shared Task [8] and the named-entity annotations in the ACE-2004 and ACE-2005 tasks.[5] The list of geographical names for the seedlist feature is taken from `geonames.org`;[6] Lists of person names and organization names are taken from the JRC Names corpus [9].[7].

4 Results

Table 1. Overall named entity recognition scores by the system and its components

Component	Precision	Recall	F-score
Capitalized	54.62	63.75	58.83
Lowercased	57.38	62.86	60.00
Intersection	65.82	57.21	61.21

Table 1 displays the overall scores of the final system, the intersection of the two MBT systems, together with the scores of the two systems separately. A test was run on a development set of 22,358 tokens containing 1,131 named entities extracted from the MSM-2013 training set. The capitalized MBT system attains the best recall, while the lowercased MBT attains the higher precision score. The intersection of the two predictably boosts precision at the cost of a lower recall, and attains the highest F-score of 61.21. If the gazetteer features are disabled, overall precision increases slightly from 65.8 to 66.1, but recall decreases from 57.2 to 54.9, leading to a lower F-score of 60.0. This is a predictable effect of gazetteers: they allow the recognition of more entities, but they import noise due to the context-insensitive matching of names in incorrect entity categories.

Table 2 lists the precision, recall, and F-scores on the four named entity types distinguished in the challenge. Person names are recognized more accurately than location and organization names; the miscellaneous category is hard to recognize.

[4] `http://oak.dcs.shef.ac.uk/msm2013/challenge.html`

[5] `http://projects.ldc.upenn.edu/ace/`

[6] `http://download.geonames.org/export/dump/allCountries.zip`

[7] `http://optima.jrc.it/data/entities.gzip`

Table 2. Overall named entity recognition scores on the four entity types

Named entity type	Precision	Recall	F-score
Person	75.90	69.52	72.57
Location	54.95	44.25	49.02
Organization	47.46	39.25	42.97
Miscellaneous	17.54	11.39	13.85

References

1. Tjong Kim Sang, E., Veenstra, J.: Representing text chunks. In: Proceedings of EACL'99, Bergen, Norway (1999) 173–179
2. Daelemans, W., Zavrel, J., Berck, P., Gillis, S.: MBT: A memory-based part of speech tagger generator. In Ejerhed, E., Dagan, I., eds.: Proceedings of the Fourth Workshop on Very Large Corpora, ACL SIGDAT (1996) 14–27
3. Daelemans, W., Zavrel, J., Van den Bosch, A., Van der Sloot, K.: MBT: Memory based tagger, version 3.0, reference guide. Technical Report ILK 07-04, ILK Research Group, Tilburg University (2007)
4. Ritter, A., Clark, S., Etzioni, O., et al.: Named entity recognition in tweets: an experimental study. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics (2011) 1524–1534
5. Li, C., Weng, J., He, Q., Yao, Y., Datta, A., Sun, A., Lee, B.S.: Twiner: Named entity recognition in targeted twitter stream. In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, ACM (2012) 721–730
6. Liu, X., Wei, F., Zhang, S., Zhou, M.: Named entity recognition for tweets. ACM Transactions on Intelligent Systems and Technology (TIST) **4**(1) (2013) 3
7. Marcus, M., Santorini, S., Marcinkiewicz, M.: Building a Large Annotated Corpus of English: the Penn Treebank. Computational Linguistics **19**(2) (1993) 313–330
8. Tjong Kim Sang, E., De Meulder, F.: Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In Daelemans, W., Osborne, M., eds.: Proceedings of CoNLL-2003, Edmonton, Canada (2003) 142–147
9. Steinberger, R., Pouliquen, B., Kabadjov, M., Belyaeva, J., van der Goot, E.: Jrcnames: A freely available, highly multilingual named entity resource. In: Proceedings of the 8th International Conference 'Recent Advances in Natural Language Processing. (2011) 104–110

Challenge Submissions – Section II:

Towards Concept Identification using a Knowledge-Intensive Approach

Óscar Muñoz-García[1], Andrés García-Silva[2], and Oscar Corcho[2]

[1] Havas Media Group, Madrid, Spain,
oscar.munoz@havasmg.com
http://www.havasmg.com

[2] Ontology Engineering Group, Departamento de Inteligencia Artificial
Universidad Politécnica de Madrid, Madrid, Spain,
hgarcia@fi.upm.es, ocorcho@fi.upm.es
http://www.oeg-upm.net

Abstract. This paper presents a method for identifying concepts in microposts and classifying them into a predefined set of categories. The method relies on the DBpedia knowledge base to identify the types of the concepts detected in the messages. For those concepts that are not classified in the ontology we infer their types via the ontology properties which characterise the type.

Keywords: concept identification, microposts, dbpedia

1 Introduction

In this paper we present an approach to identify concepts and their types in micro posts relying on the DBpedia knowledge base and ontology. Our approach consist first in carrying out a preprocessing task where messages are normalised. Then we attempt to identify candidate concepts leveraging part-of-speech tags and Wikipedia article titles. Next we associate the candidate concepts with DBpedia resources and tap into the ontology hierarchy of classes and resource properties to classify the resource in one of the following types: Person, Organization, Location, and Miscellaneous, which covers films, sport events, software, awards and television shows.

2 Spotting concepts

The concept spotting stage analyses the micropost for extracting the keywords that are candidates for being concepts, or can serve as context for disambiguating the concepts. The stage is executed in three steps, namely:

1. Text normalisation.
2. Part-of-speech tagging.
3. Keyword selection.

Next, each of the steps is described.

· **#MSM2013 Workshop Concept Extraction Challenge Proceedings** ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

2.1 Text Normalisation

The text normalisation step converts the text of the micropost, that often includes metalanguage elements, to a syntax more similar to the usual natural language. Previous results demonstrate that this normalisation step improves the accuracy of the part-of-speech tagger [3]. Specifically, we have implemented several rules for syntactic normalization of Twitter messages (some of them have been described in [6]). The rules executed are the following:

- Transform to lower-case the text completely written with upper-case characters.
- Delete the sequence of characters "RT" followed by a mention to a Twitter user (marked by the symbol "@") and, optionally, by a colon punctuation mark.
- Delete mentions to users that are not preceded by a coordinating or subordinating conjunction, a preposition, or a verb.
- Delete the word "via" followed by a mention to a user at the end of the tweet.
- Delete the hashtags found at the end of the tweet.
- Delete the "#" symbol from the hasthtags that are maintained.
- Delete the hyperlinks contained within the tweet.
- Delete ellipses that are at the end of the tweet, followed by a hyperlink.
- Delete characters that are repeated more than twice (e.g., "yeeeeeessss" is transformed to "yes").
- Transform underscores to blank spaces.
- Divide camel-cased words in multiple words (e.g., "AnalyticsTools" is converted to "Analytics Tools").

2.2 Part-of-speech Tagging

After normalising the micropost text, we execute the part-of-speech analysis of the normalised text. For doing so, we make use of Freeling [7].

2.3 Keyword Selection

Once the part-of-speech tagging is obtained, the keyword selection step is executed. For each sentence within the micropost text we extract all the possible n-grams. In this case a gram is a word in the sentence. After that we select only the n-grams that satisfy the following criteria:

- The n-gram contains at least one noun.
- The n-gram is not contained in a set of stop words.
- If the number of words included in the n-gram is greater than one, the n-gram is included in the set of Wikipedia article titles.
- The n-gram is not contained in another n-gram that has been added to the keyword set (longer n-grams prevail).

To speed-up the process of querying the millions of Wikipedia article titles we have uploaded the list of titles (available at [9]) to a Redis store [8].

3 Semantics of the Concepts

To identify the semantics of the keywords we tap into the DBpedia knowledge base [2] to elicit the types of the concepts to which the keywords correspond. DBpedia contains knowledge from Wikipedia for close to 3.5 million resources; 1.6 million resources are classified in a cross domain ontology containing 272 classes. DBpedia strengths include its large coverage and the fact that its data are exposed in RDF allowing to query them using SPARQL queries through the available endpoint [4].

Our process starts by identifying for each keyword the DBpedia resource which represents its intended meaning. Once we have the corresponding resource we query in DBpedia its classes, whenever they are available, or infer them through the identification of specific resource properties, so that we can identify the types defined in the challenge.

3.1 From keywords to DBpedia resources

First we query DBpedia for a resource with a label matching the keyword. We use exact string matching between the resource label and the keyword which has been previously modified to fit the style of article titles in Wikipedia. The output resource of this query represents the most frequent meaning of the keyword defined by Wikipedia editors. We call this resource default sense of a keyword. If the resource is not related to a disambiguation resource, we consider that the term is not ambiguous and therefore we use the default sense as the one representing the keyword meaning.

In case we do not find a match between the keyword and a resource label we use an spelling service that suggests similar titles of Wikipedia articles. This spelling service[3] compares the n-grams based on characters of both keywords and article titles, and takes into account the popularity of the articles in Wikipedia (*i.e.*, the times that an article has been linked from other articles) when producing the final ranking of suggestions. We use the most similar suggestion, above a given threshold, for searching for the DBpedia resource.

If the resource is related to a disambiguation resource, then we have to select the proper sense among the candidates. To do so we leverage the correspondence of DBpedia resources with Wikipedia articles to obtain textual descriptions of each resource. Thus, we calculate similarity between each resource and the term by comparing the resource textual description with the term context. The most similar resource is selected as the resource representing the term meaning. By context we mean the set of keywords identified in the same sentence.

To calculate similarity between the keyword context and the resource description we use a vector space model. The components of the vectors are the most frequent terms of the Wikipedia articles related to each candidate DBpedia resource. To populate the vectors representing resources we use term frequency

[3] The spelling service was built upon Lucene [1] spell checker.

and inverse document frequency (TF-IDF) as term weighting scheme. IDF is calculated using only the set of textual descriptions corresponding to the candidate resources. We calculate the cosine of the angle between the vector representing the keyword context with each of the vectors representing candidate resources. The candidate with the highest cosine is selected as the resource to represent the keywords. Details of this procedure can be found in [5].

In short for ambiguous keywords if there is not context and there is a default sense we select the DBpedia resource corresponding to the default sense. If there is context and default sense, and the context do not overlap with any of the candidate vectors we use the DBpedia resource corresponding to the default sense too. If there is overlap between the context and candidate vectors we use the most similar candidate.

3.2 Identifying concept types

We manually select the classes from DBpedia and linked ontologies that allow us to identify the types of the concepts defined in the challenge. For instance,

- *dbpedia-owl:Person* and *foaf:Person* are the classes for People;
- *dbpedia-owl:Place* is the class for Location;
- *dbpedia-owl:Organisation*, *dbpedia-owl:Company*, and *umbel:Organization* are the classes for Organizations;
- *dbpedia-owl:ProgrammingLanguage*, *umbel:SoftwareObject*, *dbpedia-owl:Film*, *dbpedia-owl:TelevisionShow*, *dbpedia-owl:Award*, and *dbpedia-owl:SportsEvent*, are the classes for the Miscellaneous type.

Therefore, for each DBpedia resource we obtain its class from the ontology and classify it according to the challenge types.

However, many DBpedia resources are not classified in the ontology. For those resources we infer its type from certain properties which are characteristic of the type. For instance, from a triple

```
<subject> dbpedia-owl:birthPlace <object>
```

we can infer that object is a location given that it is the birth place of the subject described in the triple. The same rationale can be used with predicates such as *dbpedia-owl:hometown* and *dbpedia-owl:location*. Similarly, from a triple

```
<subject> dbpprop:mvp <object>
```

we can infer that the subject is an sport event since it has a most valuable player. Other predicates used for identifying sport events include *dbpprop:menDraw*, *dbpprop:teams*, *dbpprop:sport*, and *dbpprop:referee*.

Finally, in case we cannot identify the concept type using DBpedia, we use a list of concepts and their types which have been collected from the training data set. From this list we take the first type associated with that concept.

We have not included evaluation results in this extended abstract since the only available source of annotated data for the evaluation in this challenge was the the training data set (the test data set was not annotated). Given that our approach uses a list of concepts gathered from the training set is not fair to report evaluation results on this data set.

Acknowledgements

This research is partially supported by the Spanish Centre for the Development of Industrial Technology under the CENIT program, project CEN-20101037, "Social Media" (http://www.cenitsocialmedia.es).

References

1. Apache Software Foundation: Apache Lucene. http://lucene.apache.org (2013), [Online; accessed 23-May-2013]
2. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - A crystallization point for the Web of Data. Journal of Web Semantic 7(3), 154–165 (2009)
3. Codina, J., Atserias, J.: What is the text of a tweet? In: Proceedings of @NLP can u tag #user_generated_content?! via lrec-conf.org. ELRA, Istanbul, Turkey (May 2012)
4. DBpedia: DBpedia SPARQL endpoint. http://dbpedia.org/sparql (2013), [Online; accessed 23-May-2013]
5. García-Silva, A., Cantador, I., Corcho, O.: Enabling folksonomies for knowledge extraction: A semantic grounding approach. International Journal on Semantic Web and Information Systems (IJSWIS) 8(3), 24–41 (2012)
6. Kaufmann, M., Jugal, K.: Syntactic normalization of twitter messages. In: Proceedings of the International Conference on Natural Language Processing (ICON-2010) (2010)
7. Padró, L., Stanilovsky, E.: Freeling 3.0: Towards wider multilinguality. In: Proceedings of the Language Resources and Evaluation Conference (LREC 2012). ELRA, Istanbul, Turkey (May 2012)
8. Sanfilippo, S.: Redis. http://redis.io (2009), [Online; accessed 23-May-2013]
9. Wikipedia: Wikipedia:Database download. http://en.wikipedia.org/wiki/Wikipedia:Database_download (2013), [Online; accessed 23-May-2013]

Classifying short messages using collaborative knowledge bases: Reading Wikipedia to understand Twitter

Yegin Genc, Winter Mason, Jeffrey V. Nickerson

Stevens Institute of Technology
{ygenc, wmason, jnickerson}@stevens.edu

1 Introduction

To detect concepts from tweets, we leverage the content of Wikipedia. This is a form of semantic transformation: ideas that emerge in short texts are mapped onto more extensive texts that contain additional structure. This additional structure is used to amplify the signal in the short text. This idea is rooted in our previous research [1, 2], as well as in the work of other authors pursuing similar goals [3-5].

Our method has two main stages. First, we recognize candidate concepts—parts-of-tweets—that may be valid entities in the tweet. These concepts are then classified into four categories: Locations, People, Organizations, and Miscellaneous. Candidate concepts are identified by mapping tweets to Wikipedia pages, and the networks of these concepts in Wikipedia are used for filtering and classification. We believe this technique can be applied more generally to the understanding of many forms of short messages, not just tweets, utilizing many forms of collaborative knowledge bases, not just Wikipedia.

2 Concept Recognition

Automatically determining whether a word in a tweet represents a concept is not trivial, because the words may be stop words or personal or idiosyncratic concept. Wikipedia titles, on the other hand, can be viewed as representing concepts. Moreover, Wikipedia pages are situated in a network, so that the semantics of a page title can be utilized to classify the concept. Thus, as a first step, we look for parts-of-tweets that match a Wikipedia title. Specifically, concept words are extracted and submitted as search criteria against the page titles of Wikipedia articles using the Wikipedia API. To this end, we segmented each tweet in two ways: First, using Natural Language Processing toolkits , we extracted sentences and then noun phrases from each sentence. Second, we removed punctuation and extracted n-grams (n up to 4) from the entire tweet using a sliding window. To meet Wikipedia's title conventions required for matching search results, we normalized the parts-of-tweets (noun phrases and n-grams) by capitalizing the first letter and changing the rest to lower case. For the parts-of-tweets that didn't match a Wikipedia title after normalization, we also searched for a match after capitalizing each word in the text. When a part-of-tweet

· #MSM2013 Workshop Concept Extraction Challenge Proceedings ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

landed on a Wikipedia title, we ignored all the other parts-of-tweets that are its subsets. For example, when ‘Sarah Palin’ occurs in a tweet, and maps to Wikipedia page containing ‘Sarah Palin’, ‘Sarah’ and ‘Palin’ are not processed.

3 Filtering And Classification

For classification and filtering, we utilized the concept network in Wikipedia, which consists of categories and category containers. Wikipedia pages are tagged with categories they belong to and these categories are linked to one another in a graph structure. Container-categories are special categories that contain only other categories and are not referenced by any page. They arguably serve as meta-level tags for the pages that belong to its sub-graph of categories. Moreover, their titles capture the mutual themes that run through the children categories. For example, Container Category: 21st Century people by their nationality holds categories that are used to tag pages, or other categories about people. Therefore, we labeled the container-categories with the entity labels from the contest (Locations, People, Organizations, Miscellaneous) using simple keyword searches. The keywords we selected for each label are shown in Table 1. Using this keyword search process, we labeled 1,560 of the 4,227 containers. Based on our tests, we later included 9 manually selected categories from Wikipedia to our list to improve our results. We provide more detail in section 3.

For the parts-of-tweets that match a Wikipedia page title, we traverse up the page’s category graph and count how many of the categories within 3 levels of the original page fall immediately under a labeled container-category. We label the Wikipedia page, and hence the part-of-tweet, with the container label that holds the maximum number of the categories from the page’s category graph. If the categories from the traversal of the page’s category graph don’t fall under any of the labeled containers, we ignore the concept.

4 Using The Training Set

One benefit to our method is that both the concept extraction and the classification are completely unsupervised. However, we found it was possible to improve our classification results for this contest by leveraging the training set to refine our category selection, as well as to decrease the run time. homogeneous as possible.

Table 1. Keywords used to label container categories

Locations	People	Orgs.	Misc.
cities provinces states countries continents facilities buildings counties	people men women doctors musicians government officials actors actresses champions officials athletes alumni rappers soccer- players sportspeople members comedian	organizations companies colleges businesses enterprises	films television series awards events

4.1 Category Selection

During our test runs, we realized that our method works well with entities that are explicit mentions of people or locations, e.g., Sarah Palin. However, for mentions of more generic entities—e.g., Louis, Clint, or Sue—despite successfully finding a matching Wikipedia page, they are dismissed during the classification process. We observe that for such ambiguous parts-of-tweets the matching Wikipedia pages tended to be lists of its many possible meanings; such pages are called disambiguation pages. Disambiguation pages are also categorized in a graph-like structure, however their classification scheme is distinct from the other category pages and serves only to organize disambiguation pages. Therefore, we labeled 5 of the top 26 disambiguation-categories and added them to our containers list. Finally, since the MISC category includes 'Programming Languages', we included 'Computer Languages' category to our list. These manually added containers are shown in Table 2.

Table 2. Additional Categories

Category	Label
Disambiguation pages with given-name-holder lists	PER
Disambiguation pages with surname-holder lists	PER
Human name disambiguation pages	PER
Place name disambiguation pages	LOC
Educational institution disambiguation pages	ORG
Computer Languages	MISC

5 Discussion And Concluding Thoughts

The approach to classification described here takes advantage of information that has been created and curated by many thousands of people. The contest task illustrated the complexity of classifying short messages. For example, a noun such as “Canada” might be classified as a place, or as an organization. It is far from obvious that people will agree on such a classification. Tests might be run to determine the consistency of human judgment on this and related short message classification tasks; we might learn from the diversity of human judgment when such tasks are ambiguous, and, with further research, how such ambiguity might modeled in machine classification tasks. More generally, the task of classifying entities is one that is not only context dependent, but also may admit to differing degrees of certainty. If our goal is to classify as humans do, we ideally should understand the distribution of human responses. Thus, we suggest two paths for future research: one that continues to study how classification can be improved by using collaborative data stores, and another that examines human performance on such tasks, so that we may further understand and augment the still-mysterious process of sense making.

References

[1] Genc, Y., Mason, W., and Nickerson, J.V. 2012. Semantic Transforms Using Collaborative Knowledge Bases, *Workshop on Information in Networks*

[2] Genc, Y., Sakamoto, Y., and Nickerson, J.V. Discovering context: Classifying tweets through a semantic transform based on Wikipedia, In D. Schmorrow and C. Fidopiastis (Eds). *Foundations of Augmented Cognition: Directing the Future of Adaptive Systems*, Lecture Notes in Computer Science, 6780 LNAI, Springer, Berlin, 2011, 484-492.

[3] M. Michelson and S. A. Macskassy, “Discovering users' topics of interest on twitter: a first look,” *Proceedings of the fourth workshop on Analytics for noisy unstructured text data*, pp. 73–80, 2010.

[4] E. Gabrilovich and S. Markovitch, “Wikipedia-based semantic interpretation for natural language processing,” *Journal of Artificial Intelligence Research*, vol. 34, no. 1, pp. 443–498, 2009.

[5] M. Strube and S. P. Ponzetto, “WikiRelate! Computing semantic relatedness using Wikipedia,” *Proceedings of the National Conference on Artificial Intelligence*, vol. 21, no. 2, pp. 1419–1424, 2006.

Unsupervised Information Extraction using BabelNet and DBpedia

Amir H. Jadidinejad

Islamic Azad University, Qazvin Branch,
Qazvin, Iran.
amir@jadidi.info

Abstract. Using linked data in real world applications is a hot topic in the field of Information Retrieval. In this paper we leveraged two valuable knowledge bases in the task of information extraction. BabelNet is used to automatically recognize and disambiguate concepts in a piece of unstructured text. After extracting all possible concepts, DBpedia is leveraged to reason about the type of each concept using SPARQL.

Keywords: Concept Extraction, Linked Data, BabelNet, DBpedia, SPARQL.

1 BABELNET

BabelNet[1] is a multilingual lexicalized semantic network and ontology. It was automatically created by linking the largest multilingual Web encyclopedia – i.e. Wikipedia[1] – to the most popular computational lexicon of the English language – i.e. WordNet[2]. It contains an API for programmatic access of 5.5 million concepts and a multilingual knowledge-rich Word Sense Disambiguation (WSD) [3]. With the aid of this API, we can extract all possible concepts in a piece of text. These concepts are linked to DBpedia, one of the more famous parts of the Linked Data project.

2 DBPEDIA

DBpedia[4] is a project aiming to extract structured content from the information created as part of the Wikipedia project. This structured information is made available on Semantic Web formats. DBpedia allows users to query relationships and properties associated with Wikipedia concepts. In this paper we used SPARQL to query DBpedia. It's possible to reason about the type of each concept (PER, LOC, ORG, MISC) with the aid of a classic deductive reasoning using classes and subclasses. For example, "Settlement" is defined as a subclass of "Place" (although maybe not directly). That means that all Things that are "Settlements" are also "Places". "Tehran" is a "Settlement", so it is also a "Place". Using the following query:

[1] http://www.wikipedia.org

· #MSM2013 Workshop Concept Extraction Challenge Proceedings ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

```
ASK {
  {
    ?thing a ?p .
    ?p rdfs:subClassOf dbpedia-owl:Place OPTION (transi-
tive).
  }
 UNION
  {
    ?thing a dbpedia-owl:Place .
  }
}
```

It's possible to reason about the type of every “?thing” such as: http://dbpedia.org/resource/Tehran. A similar query is used for LOCATION and ORGANIZATION.

3 IMPLEMENTATION DETAILS

Our proposed solution shows in Figure 1. The input text is passed to "Text2Concept" module. This module is used "BabelNet" and "Knowledge-rich WSD" algorithm to recognize a list of concepts. Finally, "Text Reasoner" module reason about the type of each concept with the aid of DBpedia using a simple deductive reasoning.

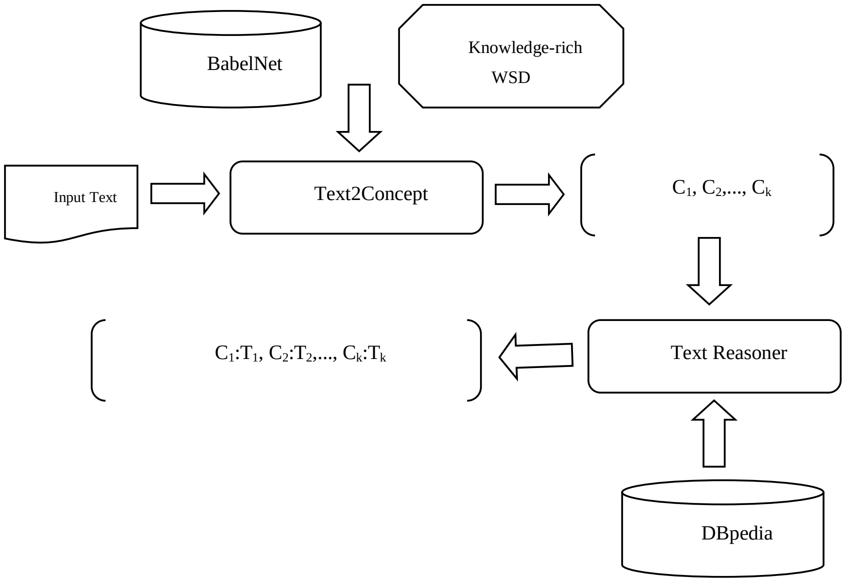
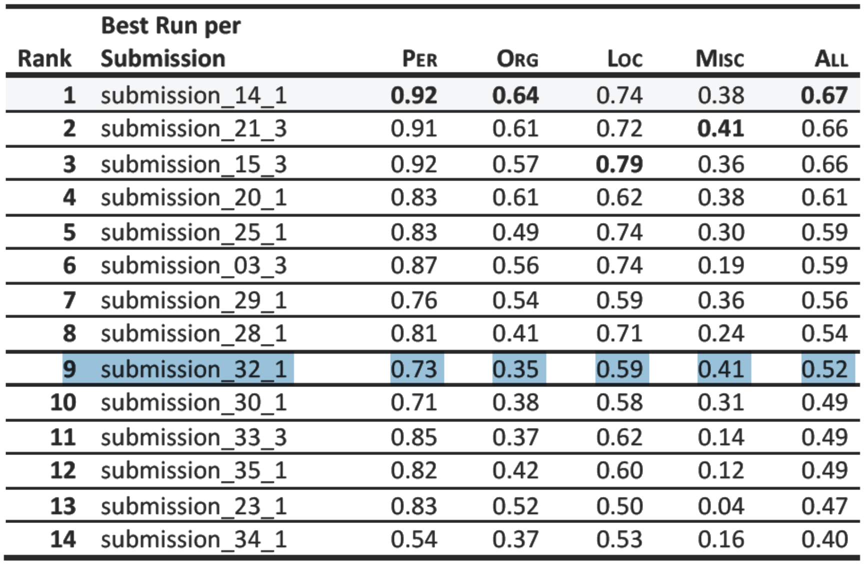


Fig. 1. Different parts of the proposed method.

Table 1 shows the impact of the proposed solution in on the training data set. Our proposed solution is achieved $F_1 = 0.50$ on training set and $F_1 = 0.52$ on testing set (See Fig. 2).

Table 1. Concept Extraction using the proposed solution on training data set

Data Set	Precision	Recall	F1
Train	0.5099	0.5003	0.5050



Rank	Best Run per Submission	PER	ORG	LOC	MISC	ALL
1	submission_14_1	**0.92**	**0.64**	0.74	0.38	**0.67**
2	submission_21_3	0.91	0.61	0.72	**0.41**	0.66
3	submission_15_3	0.92	0.57	**0.79**	0.36	0.66
4	submission_20_1	0.83	0.61	0.62	0.38	0.61
5	submission_25_1	0.83	0.49	0.74	0.30	0.59
6	submission_03_3	0.87	0.56	0.74	0.19	0.59
7	submission_29_1	0.76	0.54	0.59	0.36	0.56
8	submission_28_1	0.81	0.41	0.71	0.24	0.54
9	submission_32_1	0.73	0.35	0.59	0.41	0.52
10	submission_30_1	0.71	0.38	0.58	0.31	0.49
11	submission_33_3	0.85	0.37	0.62	0.14	0.49
12	submission_35_1	0.82	0.42	0.60	0.12	0.49
13	submission_23_1	0.83	0.52	0.50	0.04	0.47
14	submission_34_1	0.54	0.37	0.53	0.16	0.40

Fig. 2. Overall results between different participants.

4 REFERENCES

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

[2] George A. Miller (1995). WordNet: A Lexical Database for English. *Communications of the ACM*. 38, 11, 39-41.

[3] Navigli, R., Ponzetto, S. P. 2012. Multilingual WSD with Just a Few Lines of Code: the BabelNet API. In *Proc. of the 50th Annual Meeting of the Association for Computational Linguistics* (ACL 2012), Jeju, Korea, 67-72.

[4] Bizer, C., Lehmann, J., Kobilarov, G., Becker, C., Cyganiak, R., Hellmann, C. 2009. DBpedia – A Crystallization Point for the Web of Data. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 7, 154–165.

DBpedia Spotlight at the MSM2013 Challenge

Pablo N. Mendes[1], Dirk Weissenborn[2], and Chris Hokamp[3]

[1] Kno.e.sis Center, CSE Dept., Wright State University
[2] Dept. of Comp. Sci., Dresden Univ. of Tech.
[3] Lang. and Inf. Tech., Univ. of North Texas
pablo@knoesis.org,dirk.weissenborn@mailbox.tu-dresden.de
christopherhokamp@my.unt.edu

1 Introduction

DBpedia Spotlight [5] is an open source project developing a system for automatically annotating natural language text with entities and concepts from the DBpedia knowledge base. The input of the process is a portion of natural language text, and the output is a set of annotations associating entity or concept identifiers (DBpedia URIs) to particular positions in the input text. DBpedia Spotlight provides programmatic interfaces for phrase recognition and disambiguation (entity linking), including a Web API supporting various output formats (XML, JSON, RDF, etc.)

The annotations generated by DBpedia Spotlight may refer to any of 3.77 million things in DBpedia, out of which 2.35 million are classified according to a cross-domain ontology with 360 classes. Through identity links, DBpedia also provides links to entities in more than 100 other languages, and tens of other data sets. This paper describes our application of DBpedia Spotlight to the challenge of extracting Person (PER), Location (LOC), Organization (ORG) and Miscellaneous (MISC) entities from microposts (e.g. tweets) as part of the MSM2013 Challenge at WWW2013.

All of the code used in this submission is available as Open Source Software, and all of the data used is shared as Open Data. A description of the software, data sets and more detailed evaluations are available from our supporting material page at http://spotlight.dbpedia.org/research/msm2013/.

Table 1. Comparison between NER approaches on the MSM2013 Challenge Training Set.

Syst./NERType	PER	LOC	ORG	MISC	**Average**
	P / R / F1	P / R / F1	P / R / F1	P / R / F1	P / R / F1
Unsup. (1)	0.95 0.50 0.65	0.62 0.58 0.60	0.62 0.38 0.47	0.22 0.21 0.21	0.60 0.42 0.48
CRF (2)	0.86 0.66 0.75	0.82 0.7 0.76	0.73 0.56 0.63	0.49 0.29 0.36	0.72 0.53 0.61

· **#MSM2013 Workshop Concept Extraction Challenge Proceedings** ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

2 Datasets

DBpedia Spotlight's annotation model was constructed based on a number of datasets derived mainly from DBpedia and Wikipedia. First, for each DBpedia resource r, we extracted from Wikipedia all paragraphs containing wiki links with target on r's Wikipedia article. Second, from the collection of wiki links, disambiguation pages and redirects, we extracted a number of *lexicalization examples* – words that have been used to express a given DBpedia entity. Third, we use the community-maintained DBpedia Ontology mappings to collect a list of ontology classes (and superclasses) for each DBpedia resource. More details on this preliminary extraction process are available from Mendes et al., 2011 [5] and Mendes et al. 2012 [4].

To adapt this framework to the challenge, we also extended the coverage of known instance types by importing extra `rdf:type` statements between DBpedia and the DBpedia Ontology from Aprosio et al., 2013 [1], between DBpedia and Freebase[4] and between DBpedia and OpenCyc[5] by Pohl, 2012 [7].

Subsequently, we extended our lexicalization examples with a number of person and organization names based on 'naming' ontology properties such as `foaf:givenName`, `foaf:name`, `foaf:familyName`, etc.We further extended our lexicon with gazeteers from BALIE [6] including names for association, company designator, company, government, military, first name, last name, person title, celebrity, month, city, state province, country.

To allow our tool to output the target types of the challenge, we manually browsed through the ontology and created mapping from the types used in the MSM2013 Challenge, and the ontology types in the DBpedia Ontology, Freebase and OpenCyc. We refer to this set as "Manual Mappings."

Evaluation Corpus Pre-processing. The version of the MSM2013 Challenge corpus used in our evaluation contains a number of undesirable artifacts, presumably resulting from pre-processing parsing and tagging steps. The text was seemingly pre-tokenized, including spaces between tokens and punctuation, although not consistently so throughout the data set.

In our pre-processing, we attempted to reconstruct original sentences by adding extra markers as token separators (\/), as well as removing parsing artifacts (`-[LR]RB-`, `#B-ORG/`), Twitter markers (`RT`,`#\S+`), and other artifacts included in the training set for anonymization (`_URL_`, `_MENTION_`, `_Mention_`, `<NEWLINE>` and `_HASHTAG_`). For the sentence reconstruction, we also reverted the separation from the left-neighboring token of punctuation such as commas, apostrophes and exclamation marks. We will refer to this corpus as "reconstructed sentences".

[4] `http://downloads.dbpedia.org/3.8/links/freebase_links.nt.bz2`

[5] `http://opencyc.org`

3 Methodology

The Concept Extraction task proposed is very similar to the task performed by Named Entity Recognition (NER). The task can be broken down into two problems. First, a segmentation problem requires finding boundaries of entity names within sentences; and second, a classification problem requires correctly classifying the segment into one of the entity types. We have tested approaches that perform each task separately, as well as approaches that perform both tasks jointly.

First, we tested an unsupervised approach – i.e. one that does not use the training set provided in the challenge. It uses DBpedia Spotlight's phrase recognition and disambiguation to perform NER in a two-step process of segmentation and classification (dbpedia_spotlight_1.tsv). For this approach, the reconstructed sentences were sent through DBpedia Spotlight's lexicon-based recognition, and subsequently through the disambiguation algorithm. Based on the types of the entities extracted, we used our manual mappings to classify the names into one of the NER types.

Our joint segmentation/classification method is a supervised-machine learning approach enhanced by knowledge-based distant supervision from DBpedia. We use lexicalizations from DBpedia to indicate that a given token may be within an entity or concept name. This feature is intended to help with the segmentation task, particularly in cases where morphological characteristics of a word are not informative. Moreover, we use the ontology types for DBpedia resources to create a battery of features which further bias the classification task, according to the types predicted by DBpedia Spotlight.

We collected all our best features and created a Linear-Chain Conditional Random Fields (CRF) model to act as our NER (dbpedia_spotlight_2.tsv). We used Factorie [3] to implement our CRF. Our features include morphological (e.g. punctuation, word shape), context-based (e.g. surrounding tokens) and knowledge-based characteristics. Our knowledge-based features include the presence of a token within a name in our knowledge base, as well as the types predicted for this entity.

Given those features and the provided training corpus, the model is trained using stochastic gradient ascent. Gibbs sampling is used to estimate the posterior distribution for each label during training. We also added a small post-processing filter to remove whole entities that contain less than two letters or digits in them as well as entities with name ”the” and ”of”.

Finally, we included Stanford NER [2] as our third baseline (dbpedia_spotlight_3.tsv), since it is a well known NER implementation.

4 Evaluation and Discussion

Table 1 presents our evaluation results on the training set. Precision, recall and F1 on Table 1 were computed based on the overlap (using exact name and type matches) between the set of entities we extracted and the set of annotated entities. The scores shown for our supervised method are our averaged 10-fold cross-validation scores.

We also report token-based precision, recall and F1 averaged over a 10-fold cross-validation on the training set. For Stanford NER (Vanilla) (with default features), we obtain P: 0.77, R: 0.54 and F1: 0.638. For Stanford NER (Enhanced), after adding our knowledge-based features, we observe improvements to P: 0.806, R: 0.604 and F1: 0.689. The same evaluation approach applied to DBpedia Spotlight CRF yields P:0.91, R:0.72, F1:0.8.

We found the segmentation to be far harder than classification in this dataset. First, as expected in any task that requires agreement between human experts, some annotation decisions are debatable. Second, inconsistent tokenization was a big issue for our implementation.

In some cases, our model found annotations that were not included by the human-annotators, such as `ORG/twitter`, where “twitter account” could be (but was not) interpreted as an account within the `ORG` Twitter. In other cases, our model trusted the tokenization provided in the training set and predicted `MISC/Super Bowl-bound` while the human-generated annotation was `MISC/Super Bowl`.

However, in general, after guessing correctly the boundaries, the type classification seemed an easier task. Our manual mappings already obtain an average accuracy over 82%. After training, those numbers are improved even further.

However, in some cases, there seems to be some controversial issues in the classification task. Is “Mixed Martial Arts” a `Sport` or a `SportEvent`? Is “Hollywood” an organization or a location? Depending on the context, the difference can be subtle and may be missed even by the human annotators.

By far, the toughest case to classify is `MISC`. Perhaps, such a “catch all” category may be too fuzzy, even for human annotators. The annotations often contain human languages like `MISC/English;MISC/Dutch;` where the guidelines stated that only Programming languages would be annotated.

In future work we plan to carefully evaluate the contribution of each of our features, further expand our evaluations within the MISC type, and conduct a reannotation of the dataset to normalize some of the issues found.

References

1. A. P. Aprosio, C. Giuliano, and A. Lavelli. Automatic expansion of DBpedia exploiting Wikipedia cross-language information. In *ESWC'13*, Montpellier, France, 2013 (to appear).
2. J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *In ACL*, 2005.
3. A. McCallum, K. Schultz, and S. Singh. FACTORIE: Probabilistic programming via imperatively defined factor graphs. In *NIPS*, 2009.
4. P. N. Mendes, M. Jakob, and C. Bizer. DBpedia for NLP: A Multilingual Cross-domain Knowledge Base. In *LREC'12*, Istanbul, Turkey, 2012.
5. P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. DBpedia Spotlight: Shedding light on the web of documents. In *I-Semantics*, Graz, Austria, 2011.

6. D. Nadeau, P. Turney, and S. Matwin. Unsupervised Named-Entity Recognition: Generating Gazetteers and Resolving Ambiguity. *Advances in Artificial Intelligence*, 4013:266–277, 2006.
7. A. Pohl. Classifying the Wikipedia Articles into the OpenCyc Taxonomy. In *WoLE'12 at ISWC'12*, 2012.

NER from Tweets: SRI-JU System @MSM 2013

Amitava Das[1], Utsab Burman[2], Balamurali A R[3] and Sivaji Bandyopadhyay[4]

[1&3]Samsung Research India,
Bangalore, India.

[2&4]Department of Computer Science &
Engineering, Jadavpur University
Kolkata, India

{amitava.santu[1],utsab.barman.ju[2],balamurali.ar[3]}@gmail.com, sivaji_cse_ju@yahoo.com

Abstract. Now a day Twitter has become an interesting source of experiment for different NLP experiments like entity extraction, user opinion analysis and more. Due to the noisy nature of user generated content it is hard to run standard NLP tools to obtain a better result. The task of named entity extraction from tweets is one of them. Traditional NER approaches on tweets do not perform well. Tweets are usually informal in nature and short (up to 140 characters). They often contain grammatical errors, misspellings, and unreliable capitalization. These unreliable linguistic features cause traditional methods to perform poorly on tweets. This article reports the author's participation in the Concept Extraction Challenge, Making Sense of micro posts (#MSM2013). Three different systems runs have been submitted. The first run is the baseline, second run is with capitalization and syntactic feature and the last run is with dictionary features. The last run yielded than all other. The accuracy of the final run has been checked is 79.57 (precision), 71.00 (recall) and 74.79 (f-measure) respectively.

1 Introduction

Micro posts are the new form of communication in the web. Posts from different social networking sites and micro blogs reflect the present social, political and other events through user's text. Due to the limitation of message length (140 characters) and the noise of user generated content it is difficult to extract the concepts from them.

The different forms of user gen-erated noise makes Twitter text extreme noisy for standard NLP tasks. Such as -

a. Abbreviations and short forms of phonetic spelling (Examples: nite - "night", sayin -"saying"), inclusion of letter/number such as gr8-"great".

b. Acronyms (Examples: lol-"laugh out loud", iirc-"if I re-member correctly" etc).

c. Typing error/ misspelling in tweets. Examples: wouls-"would", ridiculous-"ridiculous".

d. Punctuation omission/error. (Examples: im -"I'm", dont-"don't").

e. Non-dictionary slang in tweets. This category includes word sense disambiguation (WSD) problems caused by slang uses of standard words, e.g. that was well mint ("that was very good"). It also includes specific cultural reference or group-memes.

· **#MSM2013 Workshop Concept Extraction Challenge Proceedings** ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

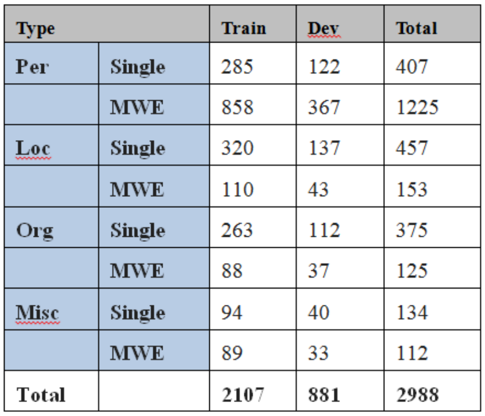
f. User's wordplay in tweets. This includes phonetic spelling and intentional misspelling for verbal effect e.g. that was soooooo great (“that was so great”).

g. Censor avoidance. This includes use of numbers or punctuation to disguise vulgarities, e.g. sh1t, f***, etc.

h. Presence of emoticons. While often recognized by a human reader, emoticons are not usually understood in NLP tasks such as Machine Translation and Information Retrieval. Examples: :) (Smiling face), <3 (heart).

2 Data

Table 1. NE Distribution of Training and Development Set



Type		Train	Dev	Total
Per	Single	285	122	407
	MWE	858	367	1225
Loc	Single	320	137	457
	MWE	110	43	153
Org	Single	263	112	375
	MWE	88	37	125
Misc	Single	94	40	134
	MWE	89	33	112
Total		2107	881	2988

The work has been done on MSM-2013 dataset. The datasets were available in 2 subsets as training and test datasets. No development set has been provided therefore the training data was divided into 2 further subsets (in 70%-30% ratio). The name entities are considered as two types - single word NE and multiword NE. The division of the available training data was made based on the presence of 4 different types of name entities with each type single and multiword. The statistics of the above process is elaborated in Table 1.

3 Experiment

Three different runs have been submitted. This is a CRF based system and the features are described below. Yamcha toolkit has been used for CRF implementation.

3.1 Baseline

Our baseline system incorporates the part of speech tags, stemmed tokens to train the baseline classifier. For POS tags of a micro post, we used CMU-POS tagger tool[1] which is specialized for tweets.

3.2 Capitalization

Capitalization of tokens is one of the key features to recognize the name entities in micro posts. It has been used as a binary feature in the classifier.

3.3 Predicate Rules

Generally the position of a name entity in a sentence is always close to the positions of functional words. For example in, of, near and etc. N-grams rules have been developed and used to train the classifier.

3.4 Out of Vocabulary Words

Most of the name entities are not the dictionary words. We used Samsad[2] & NICTA dictionary[3] in the experiment.

3.5 Gazetteers

For Location and MISC types two separate lists has been augmented. The LOC type consists of 220 country names and 100 popular city names. The MISC type has 110 NEs of different types. Mostly the error case in the Dev set.

We have experimented with series of features. Tweets are extremely noisy and therefore a concise set of named entity clue is very hard to finalize. Indeed person and organization categories are relatively naïve but location and miscellaneous category are very hard for a classifier.

4 Performance

The performance results on the Dev set is been reported in the Table 2. It should be noted the actual result on the test is yet to be evaluated by the organizer of MSM.

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

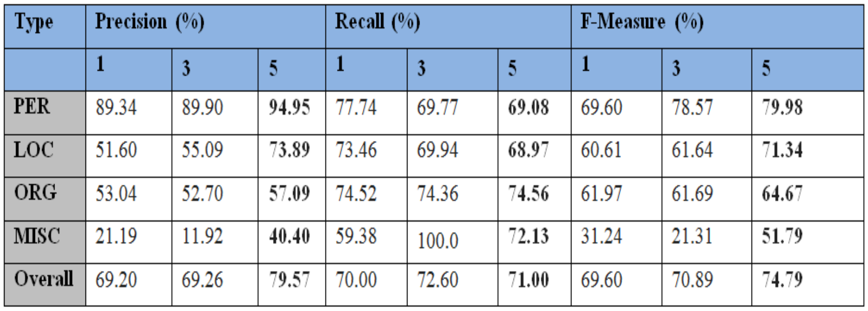
[2] http://dsal.uchicago.edu/dictionaries/biswas-bengali/

[3] http://www.csse.unimelb.edu.au/~tim/etc/emnlp2012-lexnorm.tgz

We run multiple iterations to reach the final accuracy. Broadly they could be categorized in 5 genres, as reported below. Among those iterations 3 best runs (1, 3 and 5) have been submitted. The details of the features used in each runs are as below and the scores are elaborated in Table 2.

1) Baseline: POS + Stem
2) 1 + Capitalization: Capitalization feature
3) 2 + N-Grams FW Predicates: in, of, or features
4) 3 + OOV
5) 4+Gazetters: LOC Dict + MISC Dict

Table 2. Experiment Results on Development Set



Type	Precision (%)			Recall (%)			F-Measure (%)		
	1	3	5	1	3	5	1	3	5
PER	89.34	89.90	**94.95**	77.74	69.77	**69.08**	69.60	78.57	**79.98**
LOC	51.60	55.09	**73.89**	73.46	69.94	**68.97**	60.61	61.64	**71.34**
ORG	53.04	52.70	**57.09**	74.52	74.36	**74.56**	61.97	61.69	**64.67**
MISC	21.19	11.92	**40.40**	59.38	100.0	**72.13**	31.24	21.31	**51.79**
Overall	69.20	69.26	**79.57**	70.00	72.60	**71.00**	69.60	70.89	**74.79**

5 Conclusion

In this paper we present a novel method for identification and classification of name entities based on the features. Though classifying named entities from twitter data is hard because of the noise and non-grammatical nature.

In this article we report our scores based on dev. set, we will incorporate the evaluation scores of #MSM2013 to support our evaluation framework.

Form the features that took part in our experiments, the gazetteer list, used in our experiment is small. We will try to include more in future.

We have observed that a-few Structural information can help to increase the results. For example - URL, Mention and Hash Tag. Our exploration is to find out more viable features that help to understand the semantics of micro post.

References

1. Gimpel, Kevin, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. Part-

of-speech tagging for twitter: Annotation, features, and experiments. CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, 2010.

2. Ritter, Alan, Sam Clark, and Oren Etzioni. "Named entity recognition in tweets: an experimental study." In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1524-1534. Association for Computational Linguistics, 2011.
3. Finin, Tim, et al. "Annotating named entities in Twitter data with crowdsourcing." Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, 2010.
4. Han, Bo, and Timothy Baldwin. "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, vol. 1, pp. 368-378. 2011.

MSM2013 IE Challenge NERTUW : Named Entity Recognition on tweets using Wikipedia

Sandhya Sachidanandan, Prathyush Sambaturu, and Kamalakar Karlapalem

IIIT-Hyderabad
{sandhya.s,prathyush.sambaturu}@research.iiit.ac.in
kamal@iiit.ac.in

Abstract. We propose an approach to recognize named entities in tweets, disambiguate and classify them into four categories namely person, organization, location and miscellaneous using Wikipedia. Our approach annotates the tweets on the fly, ie, it does not require any training data.

Keywords: named entity recognition, entity disambiguation, entity classification

1 Introduction

A significant amount of tweets generated each day, discusses about different types of popular entities which may be persons, locations, organizations etc. Most of the popular entities has a page in Wikipedia. Hence, Wikipedia can act as a useful source of information to recognize popular named entities in tweets. Moreover, Wikipedia contains huge number of names of different types of entities which will help us to recognize entities which does not have an explicit page in Wikipedia.

Tweets are of very short length. A tweet may or may not have enough context information to disambiguate the named entities in it. There would be a very small number of words in the tweet which supports the disambiguation of named entities which needs to be utilized efficiently. If the tweet do not have enough context to disambiguate the named entities in it, the popularity of each entity has to be leveraged in disambiguating it. Disambiguating an entity is essential to classify it correctly into location, person, organization or miscellaneous.

Our contributions are :- 1) An approach which utilizes the titles, anchors and infoboxes contained in Wikipedia and a little information from Wordnet and the context information in tweets to recognize, disambiguate and classify named entities in tweets. 2) Our approach does not require any training data and hence no human labelling effort is needed. 3) Along with the global information from Wikipedia, our approach utilizes the context information in the tweet by mapping them to their correct senses using a word sense disambiguation approach which is then used to disambiguate the named entities in the tweet. This will

· **#MSM2013 Workshop Concept Extraction Challenge Proceedings** ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

also help in disambiguating the words other than the named entities present in the tweet if any.

2 Approach

- Input tweet is split into ngrams. Link probability of each ngram is calculated as in [1], and those ngrams with link probability less than a threshold τ (experimentally set to 0.01) are discarded . Link probability of a phrase p is calculated as shown in Equation 1.

  $$LProb(p) = \frac{n_a(p, W)}{n(p, T)} \tag{1}$$

  where, $n_a(p, W)$ is the number of times a phrase p is used as an anchor text in Wikipedia W and $n(p, T)$ is the number of times the phrase occur as text in a corpus T of around one million tweets. Each concept associated with a phrase, will get the same link probability $LProb(p)$.
- For each ngram, a set of Wikipedia article titles are obtained based on their lexical match. The Wikipedia article titles mapped to the longest matching ngrams are then treated as candidate entities for disambiguation. For each ngram that matched to the title of a disambiguation page in Wikipedia, all the articles related to the ngram are added.
- The candidate entities are then passed on to a Syntax analyser, which uses YAGO's *type* relation to extract WordNet synsets mapped to the candidate entities. With the synsets mapped to the candidate entities and all the synsets of verbs and common nouns associated with the tweet as vertices, a syntax graph is generated using WordNet. The idea behind creating the syntax graph, is to identify the candidate entities which are supported by the syntax of the text. Since, this should be accompanied by disambiguation of words in the text, we found the approach proposed in [3] to be appropriate. In order to identify the candidate entities supported by the syntax of the tweet, we modify [3] by adding words from WordNet which are mapped to the candidate entities, to the syntax graph being generated. If a candidate entity is supported by the syntax of the tweet, the words from WordNet mapped to it get connected to the correct sense of the words added from the tweet in the Syntax graph. A portion of the syntax graph generated for a tweet is shown in Figure 1.
- Page Rank algorithm [2] is then applied on the syntax graph, setting high prior probabilities for synsets of common nouns and verbs added from the tweet. The average of the score of all synsets mapped to a candidate entity is treated as its syntax score.
- With the candidate entities as vertices, a semantic graph is created. The similarity between each pair of candidate entities is calculated and an edge is added with the similarity score as weight if the score is greater than an experimentally set threshold. This makes the most related candidate entities

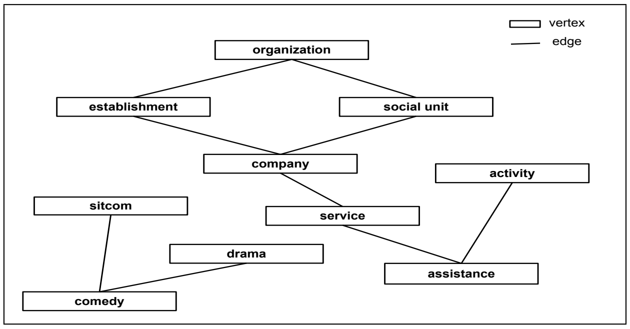


Fig. 1. A portion of the syntax graph created from the tweet - *How can #SMRT take in 161 million of profit and yet deliver sich a crapy service? why do we a company that puts....* The vertices include the words mapped to candidate entities by Yago along with all the senses of common nouns and verbs obtained from the tweet. The edges represents a relation between the vertices in WordNet.

connected in the resulting semantic graph, which may result in many connected components in the graph. An example of a so constructed semantic graph is shown in Figure 2.

- Weighted Page rank algorithm [4] is then applied on the semantic graph and the resulting scores assigned to the candidate entities is treated as the final score for ranking. The priors for each candidate entity is set as the linear combination of the following scores :-
  - Syntax score of each entity as calculated by the Syntax analyzer. This score represents the context information in the tweet.
  - Link probability of the ngram from which the candidate entity is generated.
  - Anchor probability of the candidate entity which is the number of times the entity is used as an anchor in Wikipedia. Both link probability and anchor probability represents the popularity of the candidate entity which plays a significant role in disambiguating the candidate entities in cases where a little or no context information is available in the tweet.
- ***Entity classification***: Each ngram which has a candidate entity in the semantic graph is considered as a named entity. For each ngram, the candidate entity with the highest page rank in the semantic graph is given to a named entity classifier, which uses the keywords present in the infobox of the Wikipedia page of the candidate entity to classify it as person, location, organization or miscellaneous. We extracted the unique keywords with maximum occurence, pertaining to each entity type provided in the training data to classify the named entities.

3 Error analysis and Discussion

- We use an automated and scalable approach to collect keywords from the infoboxes of Wikipedia pages to identify different entity types. Though it

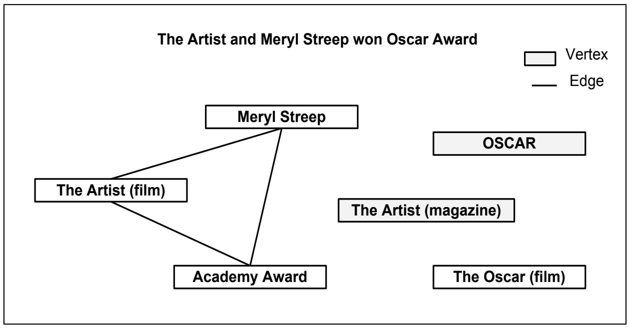


Fig. 2. A portion of semantic graph obtained from the tweet - *The Artist and Meryl Streep won oscar award*. The vertices represent the candidate entities, and edges represent their semantic relatedness.

is able to classify a significant number of entities correctly, it fails in cases where the articles do not contain infobox.

- Since not all entities are present in Wikipedia, we used a post processing step where we merge certain entities with the same type which occur adjacently in the tweet. More post processing can be done by merging adjacently located entities which are not of the same type and assign the most generic type to it which is not done.

References

1. E. Meij, W. Weerkamp, and M. de Rijke. Adding semantics to microblog posts. In *WSDM*, 2012.
2. R. Mihalcea, P. Tarau, and E. Figa. Pagerank on semantic networks, with application to word sense disambiguation. In *COLING*, 2004.
3. R. Navigli and M. Lapata. Graph connectivity measures for unsupervised word sense disambiguation. In *IJCAI*, pages 1683–1688, 2007.
4. W. Xing and A. A. Ghorbani. Weighted pagerank algorithm. In *CNSR*, pages 305–314, 2004.

Filter-Stream Named Entity Recognition: A Case Study at the MSM2013 Concept Extraction Challenge

Diego Marinho de Oliveira[1], Alberto H. F. Laender[1],
Adriano Veloso[1], Altigran S. da Silva[2]

[1] Universidade Federal de Minas Gerais, Departamento de Ciência da Computação,
Belo Horizonte, Brazil
{dmoliveira,laender,adrianov}@dcc.ufmg.br
[2] Universidade Federal do Amazonas, Instituto de Computação,
Manaus, Brazil
alti@icomp.ufam.edu.br

Abstract. Microblog platforms such as Twitter are being increasingly adopted by Web users, yielding an important source of data for web search and mining applications. Tasks such as Named Entity Recognition are at the core of many of these applications, but the effectiveness of existing tools is seriously compromised when applied to Twitter data, since messages are terse, poorly worded and posted in many different languages. In this paper, we briefly describe a novel NER approach, called FS-NER (Filter Stream Named Entity Recognition) to deal with Twitter data, and present the results of a preliminary performance evaluation conducted to assess it in the context of the Concept Extraction Challenge proposed by the 2013 Workshop on Making Sense of Microposts - MSM2013. FS-NER is characterized by the use of filters that process unlabeled Twitter messages, being much more practical than existing supervised CRF-based approaches. Such filters can be combined either in sequence or in parallel in a flexible way. Our results show that, despite the simplicity of the filters used, our approach outperformed the baseline with improvements of 4.9% on average, while being much faster.

Keywords: Twitter, Named Entity Recognition, FS-NER, CRF

1 Introduction

In this paper, we briefly describe a novel NER approach, called FS-NER (Filter Stream Named Entity Recognition), and present the results of a preliminary performance evaluation conducted to assess it in the context of the Concept Extraction Challenge proposed by the 2013 Workshop on Making Sense of Microposts - MSM2013[3]. Traditional approaches for Named Entity Recognition (NER) have demonstrated to be successful when applied to data obtained from typical Web documents, but they are ill suited to Twitter data [2, 3], since Twitter

[3] http://oak.dcs.shef.ac.uk/msm2013/challenge.html

· #MSM2013 Workshop Concept Extraction Challenge Proceedings ·
available online as CEUR Vol-1019, at: http://ceur-ws.org/Vol-1019
Making Sense of Microposts Workshop @ WWW'13, May 13th 2013, Rio de Janeiro, Brazil

messages are composed of few words and usually written in informal, sometimes cryptic style. FS-NER is an alternative NER approach better suited to deal with Twitter data [1]. In this approach, the NER process is viewed as a coarse grain Twitter message flow (i.e., a Twitter stream) controlled by a series of components, referred to as *filters*. A filter receives a Twitter message coming on the stream, performs specific processing in this message and returns information about possible entities in the message (i.e., each filter is responsible to recognize entities according to some specific criterion). Specifically, FS-NER employs five lightweight filters, exploiting nouns, terms, affixes, context and dictionaries. These filters are extremely fast and independent of grammar rules, and may be combined in sequence (emphasizing precision) or in parallel (emphasizing recall).

In our performance evaluation, we run a set of experiments using micropost data made available by the challenge organizers. Our aim in this challenge was, given a short message (i.e., a micropost), to recognize concepts generally defined as "abstract notions of things". Thus, for the purpose of the challenge our task was constrained to the extraction of entity concepts found in micropost data, characterised by a type and a value, and considering four entity types: *Person*, *Organization*, *Location* and *Miscellaneous*. We also employed a state-of-the-art CRF-based baseline. Our results show that, despite the simplicity of the filters used, our approach outperformed the baseline with improvements of 4.9% on average, while being much faster.

2 Proposed Approach

FS-NER adopts filters that allow the execution of the NER task by dividing it into several recognition processes in a distributed way. Furthermore, FS-NER adopts a simple yet effective probabilistic analysis to choose the most suitable label for the terms in the message being processed. Because of this lightweight structure, FS-NER is able to process large amounts of data in real-time. In what follows, we briefly describe the main FS-NER aspects involved. More details can be found in [1].

2.1 Structure and Design

Let $\mathcal{S} = < m_1, m_2, \ldots >$ be a stream of messages (i.e., tweets), where each m_j in $\mathcal{S}$ is expressed by a pair (X, Y), being X a list of terms $[x_1, x_2, \ldots x_n]$ that compound m_j and Y a list of labels $[y_1, y_2, \ldots, y_n]$, such that each label y_i is associated with the corresponding term x_i and assumes one of the values in the set {Beginning, Inside, Last, Outside, UnitToken}. While X is known in advance for all messages in $\mathcal{S}$, the values for the labels in Y are unknown and must be predicted. For example, the tweet "*RT: I love Mary*" could be represented by $([x_1 = \textit{RT:}, x_2 = I, x_3 = love, x_4 = Mary], [y_1 = Outside, y_2 = Outside, y_3 = Outside, y_4 = UnitToken])$.

To properly predict labels for Y, we need to provide representative data to generate a recognition model. In FS-NER, a filter is a processing component that estimates the probability of the labels associated with the terms of a message. A set of features is used to support the training of the filters (such features include information like the term itself, or if the first letter of the term is in uppercase). If a term in X satisfies one of these features, we say that the corresponding filter is activated by the term. Using the training set, we may count the number of times a filter is activated by a given term, and by inspecting the corresponding label we may calculate the likelihood of each pair $\{x_i, y_i\}$ for each filter as expressed by the equation

$$P(y_i = l | X \wedge F = k) \ = \theta_l \tag{1}$$

where F is a random variable indicating that a filter k is being used and θ_l is the probability of associating the label l with the term x_i. The probability θ_l is given by Equation 2, where TP is the number of true positive cases and FN is the number of false negative cases for the term x_i.

$$\theta_l = \frac{TP}{TP + FN} \tag{2}$$

Thus, after trained, a filter becomes able to recognize entities present in the upcoming messages. It is worth noting that each filter employs a different recognition strategy (i.e., a different feature), and thus different predictions are possible for different filters.

In sum, filters are simple abstract models that receive as input a list of terms X and a term $x_i \in X$, and provides as output a set of labels with the associated likelihood, denoted by $\{l, \theta_l\}$. Thus, a filter can be defined by

$$(X, x_i) \xrightarrow{input} F \xrightarrow{output} \{l, \theta_l\}.$$

During the recognition step, the set $\{l, \theta_l\}$ is used to choose the most likely label for the term x_i. However, if used in isolation, filters may not capture specific patterns that can be used for recognition. Fortunately, we may exploit filter combinations to boost recognition performance. Specifically, we may combine filters either in sequence (i.e., if we want to prioritize recognition precision), or in parallel (i.e., if we want to prioritize recognition recall). If combined in sequence, all filters must be activated by the input term, and the corresponding set $\{l, \theta_l\}$ is obtained by treating the combined filters as an atomic one using Equation 1. In this case, it is expected that filters when combined sequentially are able to capture more specific patterns. In contrast, if combined in parallel, the combined filters are not considered as an atomic one. Instead, they simply represent the average of the corresponding likelihoods, as expressed by the equation

$$\frac{1}{Z(\mathcal{F})} \sum_{k=1}^{K} P(y_i = l | X \wedge F = k) \tag{3}$$

where $Z(\mathcal{F})$ is a normalization function that receives as input a list of filters $\mathcal{F}$ and produces as output the number of filters activated by term x_i.

Once trained, the recognition models are used to select the most likely label for each term in the upcoming messages.

2.2 Filter Engineering

In FS-NER, features are encapsulated by five basic filters. They are the *term*, *context*, *affix*, *dictionary* and *noun* filters.

The *term filter* estimates the probability of a certain term being an entity. This estimation is based on the number of times a specific term has been assigned as an entity during the training step. The *context filter* is specially important since it is able to capture unknown entities. Hence, this filter analyzes only the terms around an observed term x_i considering a window of size n and infers whether it is an entity or not. The *affix filter* uses the fragments of an observation x_i to infer if it is an entity. Advantageously, this filter can recognize entities that have similar affix to the entities analyzed before. Thus, this filter makes use of the prefix, infix or suffix of the observation to infer its label y_i. The *dictionary filter* uses lists of names of correlated entities to infer whether the observed term is an entity. The dictionary is important to infer entities that do not appear in the training data. The *noun filter* only considers terms that have just the first letter capitalized to infer if the observed term is an entity.

3 Evaluation

We performed the preliminary evaluation of our approach with the training data made available for the MSM2013 Concept Extraction Challenge. This data includes microposts that refer to entities of types *Person* (PER), *Organization* (ORG), *Location* (LOC) and *Miscellaneous* (MISC). For this, we performed a 5-fold cross validation. To reduce noise, we applied simple preprocessing techniques like removing repeated letters and repeated adjacent terms within a micropost. We also used additional labeled Twitter data from [3] for improving recognition results for entities of types PER and LOC. The standard filter combination adopted for FS-NER was the generalized term filter combination that includes all five proposed filters and presented the best performance in [1]. In the *term* filter, the terms are case sensitive. The context filter, uses prefix and suffix contexts with a window of size three, which presented the best result for F_1 in all collections analyzed. The affix filter uses a prefix, infix and postfix size of 1 to 3. The dictionary filter, specifically, uses the same lists of names of correlated entities considered in [3] and others created from Wikipedia pages. The CRF-based framework used as baseline was the one available at `http://crf.sourceforge.net`, with features functionally similar to the FS-NER filters.

Table 1 presents the obtained results. The line *AVG-Diff* shows the average difference between the FS-NER and CRF-based framework results for all entity types. These results show that, on average, FS-NER outperformed the CRF-based framework by 4.9% for the F_1 metric.

Regarding the test dataset labeling, we followed the same procedure adopted in the preliminary experiment discussed above. In addition, we trained our approach for each entity type separately and then submitted all results together. In case of any intersection between distinct entity types, we chose the entity type that presented the most precise result among them (i.e., PER > LOC > ORG > MISC).

Entity Type	Approach	Precision	Recall	F_1
PER	FS-NER	0.7508	0.7546	0.7520
	CRF	0.7688	0.5350	0.6309
ORG	FS-NER	0.6924	0.4741	0.5612
	CRF	0.7188	0.4702	0.5685
LOC	FS-NER	0.6961	0.5400	0.6069
	CRF	0.7160	0.4656	0.5643
MISC	FS-NER	0.5734	0.3322	0.4185
	CRF	0.5610	0.2847	0.3777
AVG-Diff		-0.0130	0.0864	0.0493

Table 1: Results for FS-NER and the CRF-based framework on the challenge training dataset.

4 Conclusion

In this paper, we have briefly described a novel NER approach, called FS-NER (Filter Stream Named Entity Recognition), and presented the results of a performance evaluation conducted to assess it in the context of the Concept Extraction Challenge proposed by the 2013 Workshop on Making Sense of Microposts - MSM2013. In this challenge, our task was constrained to the extraction of entity concepts found in micropost data, characterised by a type and a value, and considering four entity types: Person, Organization, Location and Miscellaneous. We also employed a state-of-the-art CRF-based baseline. Following previous results [1], our approach outperformed the baseline with improvements of 4.9% on average, while being much faster.

Acknowledgments

This work was partially funded by InWeb - The Brazilian National Institute of Science and Technology for the Web (grant MCT/CNPq 573871/2008-6), and by the authors' individual grants from CNPq, FAPEMIG and FAPEAM.

References

1. D. M. de Oliveira, A. H. F. Laender, A. Veloso, and A. S. da Silva. FS-NER: A Lightweight Filter-Stream Approach to Named Entity Recognition on Twitter Data. In *Proceedings of the 22nd International World Wide Web Conference (Companion Volume)*, pages 597–604, 2013.
2. K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *Proceedings of the Association for Computational Linguistics (Short Papers)*, pages 42–47, 2011.
3. A. Ritter, S. Clark, Mausam, and O. Etzioni. Named Entity Recognition in Tweets: An Experimental Study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, 2011.

Appendices

A #MSM2013 Challenge Dataset Description

- 4341 manually annotated microposts
- 60% (training) / 40% (test data)

Anonymisation and Special Terms

To ensure anonymity all username mentions in the microposts were replaced with '_Mention_', and all URLs with '_URL_'

License

Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License[18]

Training Dataset

TSV data with indices specified as:

- *Element 1*: numeric ID of the micropost
- *Element 2*: concepts found within the micropost – as semi-colon separated entity type/instance pairs (e.g. PER/Obama;ORG/NASA)
- *Element 3*: micropost content – from which concepts had been detected and extracted

Sample snapshot (matching instances highlighted):

172 PER/OBAMA; _Mention_ CONGRESS OUTLAWS USE OF AUTOTUNES AFTER PRES. **OBAMA** INSISTS ON USING IT WHEN GIVING ALL HIS SPEECHES #LOCOPREDICTIONS2011
173 ORG/Amazon; _Mention_ there is no way one can explain this book that will sound reasonable and shame on **Amazon** !
...

1844 PER/Obama;PER/Andy Borowitz;ORG/White House; "Hu Presents **Obama** with Counterfeit DVD : Fake news by **Andy Borowitz** In a moving **White House** ceremony today , President ... _URL_"
...

1846 Hurricane simulator : pay $ 2 to stand in a glass booth and get wind blown on you . This is real .

[18] http://creativecommons.org/licenses/by-nc-sa/3.0/deed.en_US

Test Dataset

Unindexed TSV data:

- *Element 1*: numeric ID of the micropost
- *Element 2*: micropost content – from which concepts were to be detected and extracted

Sample snapshot:

2573 Politics is the art of preventing people from taking part in affairs which properly concern them . <NEWLINE> - Paul Valery

2574 "Pork chops , dirty rice , steamed vegetables , & Texas toast

B #MSM2013 Challenge Task Description

Concepts, especially with reference to ontologies, are defined as 'abstract notions of things'[19]. For the purposes of this challenge the task was constrained to the extraction of entity concepts in Micropost data, characterised by a type and a value. The classification of concepts was restricted to four entity types – where the Micropost contains a reference to:

1. a ***Person (PER)*** – full or partial person names

   Data sample:
   "*Obama responds to diversity criticism*"
   Extracted instance(s):
   PER/Obama;

2. a ***Location (LOC)*** – full or partial (geographical or physical) location names, including: cities, provinces or states, countries, continents and (physical) facilities

   Data sample:
   "*Finally on the train to London ahhhh*"
   Extracted instance(s):
   LOC/London;

3. a ***Organisation (ORG)*** – full or partial organisation names, including academic, state, governmental, military and business or enterprise organisations

   Data sample:
   "*NASA's Donated Spy Telescopes May Aid Dark Energy Search*"
   Extracted instance(s):
   ORG/NASA;

4. a ***Miscellaneous (MISC)*** – a concept not covered by any of the categories above, but limited to one of the entity types: film/movie, entertainment award event, political event, programming language, sporting event and TV show.

   Data sample:
   "*Okay, now this is getting seriously bizarre. Like a Monty Python script gone wrong.*"
   Extracted instance(s):
   MISC/Monty Python;

[19] See http://www.merriam-webster.com/dictionary/concept, http://en.wikipedia.org/wiki/Concept

Classification of Results

Results were to be returned for up to three runs, each in a TSV file, encoded as:

- *Element 1*: numeric ID of each micropost
- *Element 2*: concepts detected within the micropost – as semi-colon separated entity type/instance pairs (e.g. PER/Obama;ORG/NASA)

Sample submission:

For the dataset sample in Appendix A, a correctly classified submission is as below:

```
...

173 ORG/Amazon;
...

1844 PER/Obama;PER/Andy Borowitz;ORG/White House;
...

1846
```