Unsupervised Improvement of Named Entity Extraction in Short Informal Context Using Disambiguation Clues

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. Short context messages (like tweets and SMS's) are a potentially rich source of continuously and instantly updated information. Shortness and informality of such messages are challenges for Natural Language Processing tasks. Most efforts done in this direction rely on machine learning techniques which are expensive in terms of data collection and training.
In this paper we present an unsupervised Semantic Web-driven approach to improve the extraction process by using clues from the disambiguation process. For extraction we used a simple Knowledge-Base matching technique combined with a clustering-based approach for disambiguation. Experimental results on a self-collected set of tweets (as an example of short context messages) show improvement in extraction results when using unsupervised feedback from the disambiguation process.

1 Introduction

The rapid growth in IT in the last two decades has led to a growth in the amount of information available on the World Wide Web. A new style for exchanging and sharing information is short context. Examples for this style of text are tweets, social networks' statuses, SMS's, and chat messages.

In this paper we use twitter messages as a representative example of short informal context. Twitter is an important source for continuously and instantly updated information. The average number of tweets exceeds 140 million tweet per day sent by over 200 million users around the world. These numbers are growing exponentially [1]. This 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. IE systems analyze human language text in order to extract information about pre-specified types of events, entities, or relationships. Named entity *extraction* (NEE) (a.k.a. named entity recognition) 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. While named entity *disambiguation* (NED) is the task of exploring which correct person, place, event, etc. is referred to by a mention.

NEE & NED processes on short messages are basic steps of many SMS services such as [2] where users' communities can use mobile messages to share information. NLP tasks on short context messages are very challenging. The challenges come from

the nature of the messages. For example: (1) Some messages have limited length of 140 characters (like tweets and SMS's). (2) Users use acronyms for entire phrases (like LOL, OMG and b4). (3) Words are often misspelled, either accidentally or to shorten the length of the message. (4) Sentences follow no formal structure.

Few research efforts studied NEE on tweets [3–5]. Researchers either used off-the-shelf trained NLP tools known for formal text (like part of speech tagging and statistical methods of extraction) or retrained those techniques to suit informal text of tweets. Training such systems requires annotating large datasets which is an expensive task.

NEE and NED are highly dependent processes. In our previous work [6] we showed this interdependency in one kind of named entity (toponyms). We proved that the effectiveness of extraction influences the effectiveness of disambiguation, and reciprocally, the disambiguation results can be used to improve extraction. The idea is to have an extraction module which achieves a high recall; clues from the disambiguation process are then used to discover false positives. We called this behavior *the reinforcement effect*.

Contribution: In this paper we propose an unsupervised approach to prove the validity of the reinforcement effect on short informal text. Our approach uses Knowledge-Base (KB) lookup (here we use YAGO [7]) for entity mention extraction. This extraction approach achieves high recall and low precision due to many false positive matches. After extraction, we apply a cluster-based disambiguation algorithm to find coherent entities among all possible candidates. From the disambiguation results we find a set of isolated entities which are not coherent to any other candidates. We consider the mentions of those isolated entities as false positives and therewith improve the precision of extraction. Our approach is considered unsupervised as it doesn't require any training data for extration or disambiguation.

Furthermore, we propose an idea to solve the problem of lacking context needed for disambiguation by constructing profiles of messages with the same hashtag or messages sent by the same user. Figure 1 shows our approach on tweets as an example for short messages.

Assumptions: In our work we made the following assumptions:

(1) We consider the KB-based NEE process as a basic predecessor step for NED. This means that we are only concerned with named entities that can be disambiguated. NED cannot be done without a KB to lookup possible candidates of the extracted mentions. Thus, we focus on public and famous named entities like players, companies, celebrities, locations, etc.
(2) We assume the messages to be informative (i.e. contains some useful information about one or more named entities). Dealing with noisy messages is not within our scope.

2 Proposed Approach

In this work we use YAGO KB for extraction as well as disambiguation processes. YAGO is built on Wikipedia, WordNet, and GeoNames. It contains more than 447 million facts for 9.8 million entities. A fact is a tuple representing a relation between two entities. YAGO has about 100 relations, such as `hasWonPrize`, `isKnownFor`,

isLocatedIn and hasInternalWikipediaLinkTo. Furthermore, it contains relations connecting mentions to entities such as hasPreferredName, means, and isCalled. The means relation represents the relation between the entity and all possible mention representations in wikipedia. For example the mentions {"*Chris Ronaldo*", "*Christiano*", "*Golden Boy*", "*Cristiano Ronaldo dos Santos Aveiro*"} and many more are all related to the entity "*Christiano_Ronaldo*" through the means relation.

2.1 Named Entity Extraction

The list lookup strategy is an old method of performing NEE by scanning all possible n-grams of a document content against the mentions-entities table of a KB like YAGO or DBpedia [8]. Due to the short length of the messages and the informal nature of the used language, KB lookup is a suitable method for short context NEE.

The advantages of this extraction method are:

(1) It prevents the imperfection of the standard extraction techniques (like POS) which perform quite poorly when applied to Tweets [3].
(2) It can be applied on any language once the KB contains named entity (NE) representations for this language.
(3) It is able to cope with different representations for a NE. For example consider the tweet "*fact: dr. william moulton marston, the man who created wonder woman, also designed an early lie detector*", standard extractors might only be able to recognize either "*dr. william moulton marston*" or "*william moulton marston*" but not both (the one that maximizes the extraction probability). Extraction of only one representation may cause a problem for the disambiguation when matching the extracted mention against the KB which may contain a different representation for the same entity. We followed the longest match strategy for mentions extraction.
(4) It is able to find NEs regardless of their type. In the same example, other extractors may not be able to recognize and classify "*wonder woman*" as a NE, although it is the name of a comic character and helps to disambiguate the mention "*william moulton marston*".

On the other hand, the disadvantages of this method for NEE are:

(1) Not retrieving correct NEs which are misspelled or don't match any facts in the KB.
(2) Retrieving many false positives (n-grams that match facts in the KB but do not represent a real NE).

This results in a high recall and low precision for the extraction process. In this paper we suggest a solution for the second disadvantage by using feedback from NED in an unsupervised manner for detecting false positives.

As we are concerned with NED, it is inefficient to annotate all the n-grams space as named entities to achieve recall of 1. To do NED we still need a KB to lookup for the named entities.

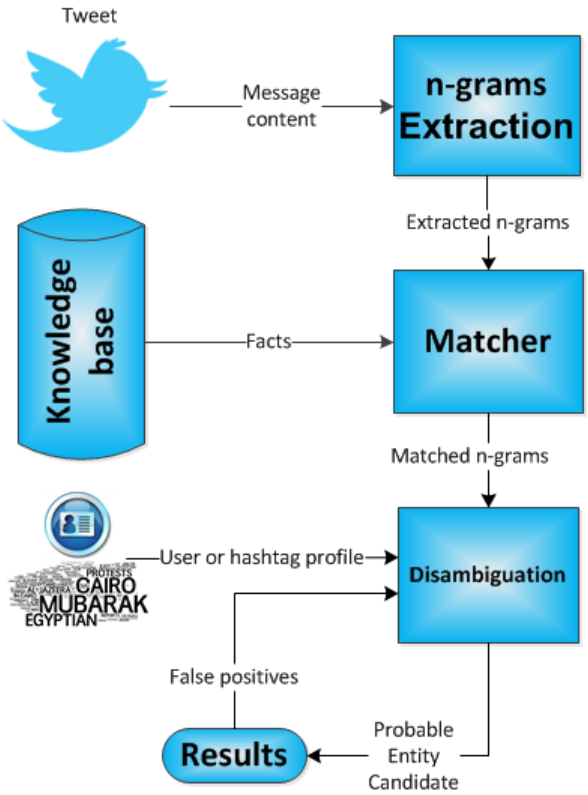


Fig. 1: Proposed Approach for Twitter NEE & NED.

2.2 Named Entity Disambiguation

NED is the process of establishing mappings between extracted mentions and the actual entities [9]. For this task comprehensive gazetteers such as GeoNames or KBs such as DBpedia, Freebase, or YAGO are required to find entity candidates for each mention.

To prove the feasibility of using the disambiguation results to enhance extraction precision, we developed a simple disambiguation algorithm (see Algorithm 1). This algorithm assumes that the correct entities for mentions appearing in the same message should be related to each other in YAGO KB graph.

The input of the algorithm is the set of all candidate entities $R(m_i)$ for the extracted mentions m_i. The algorithm finds all possible $permutations$ of the entities. Each permutation includes one candidate entity for each mention. For each permutation p_l we apply agglomerative clustering to obtain a set of clusters of related entities ($Clusters(p_l)$) according to YAGO KB. We determine $Clusters(p_l)$ having minimum size.

The agglomerative clustering starts with each candidate in p_l as a separate cluster. Then it merges clusters that contains related candidates. Clustering terminates when no more merging is possible.

Table 1: Examples of NED output (Real mentions and their correct entities are shown in Bold)

Tweet	rt @breakingnews: explosion reported at a coptic church in alexandria, egypt; several killed - bbc.com	wp opinion: mohamed elbaradei •egypt's real state of emergency is its repressed democracy
Extracted mentions	**coptic church**, church in, killed, **egypt**, **bbc.com** **alexandria**, explosion, reported	state of emergency, **egypt**, opinion, real, **mohamed elbaradei**, repressed, democracy
Groups of related candidate entities	{**Coptic_Orthodox_Church_of_Alexandria**, **Alexandria**, **Egypt**, **BBC_News**}, {Churches_of_Rome},{Killed_in_action}, {Space_Shuttle_Challenger_disaster}, {Reported}	{State_of_emergency},{**Mohamed_ElBaradei**, **Egypt**}, {Repressed}, {Democracy_(play)}, {Real_(L'Arc-en-Ciel_album)}

Two candidates for two different mentions are considered related if there exists a direct or indirect path from one to the other in YAGO KB graph. Direct paths are defined as follows: candidate e_{ij} is related to candidate e_{lk} if there exists a fact of the form $<e_{ij}$, `some relation`, $e_{lk}>$. For indirect relations, candidate e_{ij} is related to candidate e_{lk} if there exist two facts of the form $<e_{ij}$, `some relation`, $e_{xy}>$and a fact $<e_{xy}$, `some relation`, $e_{lk}>$. We refer to the direct and the indirect relation in the experimental results section with *"relations of depth 1"* and *"relations of depth 2"*.

We didn't go further than relations with length more than 2, because the time needed to build an entity graph grows exponentially with the increase in the number of levels. In addition, considering relations of a longer path is expected to group all the candidates in one cluster as they are likely to be related to each other through some intermediate entities.

Finding false positives: We select the winning $Clusters(p_l)$ as the one having minimum size. We expect to find one or more clusters that include almost all correct entities of all real mentions and other clusters each containing only one entity. Those clusters with size one contain most probably entities of false positive mentions.

Table 1 shows two examples for tweets along with the extracted mentions (using the KB lookup) and the clusters of related candidate entities. It can be observed that the correct candidate of real mentions are grouped in one cluster while false positives ended up alone in individual clusters.

Like the KB lookup extractor, this method of disambiguation can be applied on any language once the KB contains NE mentions for this language.

3 Experimental Results

Here we present some experimental results to show the effectiveness of using the disambiguation results to improve the extraction precision by discovery of false positives. We also discuss the weak points of our approach and give some suggestions for how to overcome them.

Algorithm 1: The disambiguation algorithm

input : $M = \{m_i\}$ set of extracted mentions, $R(m_i) = \{e_{ij} \in$ Knowledge base$\}$ set of candidate entities for m_i

output: $Clusters(p_l) = \{c_j\}$ set of clusters of related candidate entities for permutation p_l where $|Clusters(p_l)|$ is the minimum

$$Permutations = \{\{e_{1x}, \ldots, e_{nx}\} \mid \forall 1 \leq i \leq n \exists! x : e_{ix} \in R(m_i)\}$$

foreach *Permutation* $p_l \in Permutations$ **do**
  $Clusters(p_l) = Agglomerative\_Clustering\{p_l\}$;
end
Find $Clusters(p_l)$ with minimum size;

Table 2: Evaluation of NEE approaches

	Strict			**Lenient**			**Averag**		
	Pre.	**Rec.**	**F1**	**Pre.**	**Rec.**	**F1**	**Pre.**	**Rec.**	**F1**
Stanford	1.0000	0.0076	0.0150	1.0000	0.0076	0.0150	1.0000	0.0076	0.0150
Stanford_lower	0.7538	0.0928	0.1653	0.9091	0.1136	0.2020	0.8321	0.1032	0.1837
KB_lu	0.3839	0.8566	0.5302	0.4532	0.9713	0.6180	0.4178	0.9140	0.5735
KB_lu + rod_1	0.7951	0.4302	0.5583	0.8736	0.4627	0.6050	0.8339	0.4465	0.5816
KB_lu + rod_2	0.4795	0.7591	0.5877	0.5575	0.8528	0.6742	0.5178	0.8059	0.6305

3.1 Data Set

We selected and manually annotated a set of 162 tweets that are found to be rich with NEs. This set is collected by searching in an open collection of tweets[1] for named entities that belong to topics like politics, sports, movie stars, etc. Messages are selected randomly from the search results. The set contains 3.23 NE/tweet on average.

Capitalization is a key orthographic feature for extracting NEs. Unfortunately in informal short messages, capitalization is much less reliable than in edited texts [3]. To simulate the worst case of informality of the tweets, we turned the tweets into lower case before applying the extractors.

3.2 Experiment

In this experiment we evaluate a set of extraction techniques on our data set:

- **Stanford**: Stanford NER [10] trained on normal CoNLL collection.
- **Stanford_lower**: Stanford NER trained on CoNLL collection after converting all text into lower case.
- **KB_lu**: KB lookup.

[1] http://wis.ewi.tudelft.nl/umap2011/#dataset

Table 3: Examples some problematic cases

Case #	Message Content
1	rt @wsjindia: india tightens rules on cotton exports http://on.wsj.com/ev2ud9
2	rt @imdb: catherine hardwicke is in talks to direct 'maze runners', a film adaptation of james dashner's sci-fi trilogy. http://imdb.to/

- **KB_lu + rod_1**: KB lookup + considering feedback from disambiguation with *relations of depth 1*.
- **KB_lu + rod_2**: KB lookup + considering feedback from disambiguation with *relations of depth 2*.

The results are presented in table 2. The main observations are that the Stanford NER performs badly on our extraction task; and as expected the KB lookup extractor is able achieve high recall and low precision; and feedback from the disambiguation process improved overall extraction effectiveness (as indicated by the F1 measure) by improving precision at the expense of some recall.

3.3 Discussion

In this section we discuss in depth the results and causes.

Capitalization is a very important feature that NEE statistical approaches rely on. Even training Stanford CRF classifier on lower case version of CoNLL does not help to achieve reasonable results.

KB_lu extractor achieves a high recall with low precision due to many false positives. While **KB_lu + rod_1** achieves high precision as it looks only for direct related entities like "*Egypt*" and "*Alexandria*".

By increasing the scope of finding related entities to depth 2, **KB_lu + rod_2** finds more related entities and hence fails to discover some false positives. This leads to a drop in the recall and an enhancement in both precision and F1 measure (compared with **KB_lu**).

One major problem that harms recall is to have a message with an entity not related to any other NEs or to have only one NE within the message. Case 1 in table 3 shows a message with only one named entity (india) that ends up alone in a cluster and thus considered false positive. A suggestion to overcome such problem is to expand the context by also considering messages replied to this submission or messages having the same hashtag or messages sent by the same user. It is possible to get enough context needed for the disambiguation process using user or hashtag profiles. Figures 2(a), 2(b) and 2(c) show the word clouds generated for the hashtags "*Egypt*", "*Superbowl*" and for the user "*LizzieViolet*" respectively. Word clouds for hashtags are generated from the TREC 2011 Microblog Track collection of tweets [2]. This collection covers both the time period of the Egyptian revolution and the US Superbowl. The terms size in the

[2] http://trec.nist.gov/data/tweets/

word cloud proportionates the probability that the term is being mentioned in the profile tweets.

Another problem that harms precision are entities like the "*United_States*" that are related to many other entities. In case 2 of table 3, the mention "*talks*" is extracted as named entity. One of its entity candidates is "*Camp_David_Accords*" which is grouped with "*Catherine_Hardwicke*" as they both are related to the entity "*United_States*" (using **KB_lu + rod_2**). Both entities are related to "*United_States*" through relation of type "`hasInternalWikipediaLinkTo`". A suggestion to overcome this problem is to incorporate a weight representing the strength of the relation between two entities. This weight should be inversely proportional to the degree of the intermediate entity node in the KB graph. In our example the relation weight between "*Camp_David_Accords*" and "*Catherine_Hardwicke*" should be very low because they are related together through "*United_States*" which has a very high number of edges connected to its node in the KB graph.

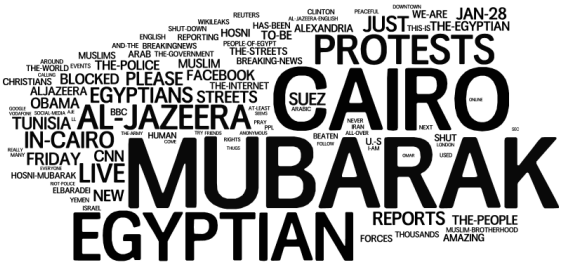
4 Conclusion and Future Work

In this paper we introduced an approach for unsupervised improvement of Named Entity Extraction (NEE) in short context using clues from Named Entity Disambiguation (NED). To show its effectiveness experimentally, we chose an approach for NEE based on knowledge base lookup. This method of extraction achieves high recall and low precision. Feedback from the disambiguation process is used to discover false positives and thereby improve the precision and F1 measure.

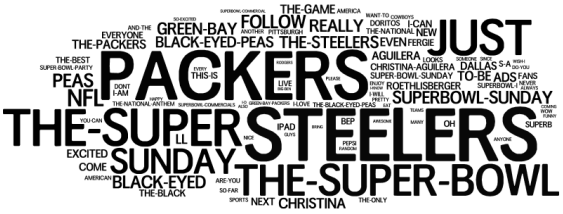
In our future work, we aim to enhance our results by considering a wider context than a single message for NED, applying relation weights for reducing the impact of non-distinguishing highly-connected entities, and to study the portability of our approach across multiple languages.

References

1. A. Gervai. Twitter statistics - updated stats for 2011. http://www.marketinggum.com/twitter-statistics-2011-updated-stats/, accessed 30-November-2011.
2. Mena B. Habib. Neogeography: The challenge of channelling large and ill-behaved data streams. In *Workshops proc. of ICDE 2011*, 2011.
3. Mausam A. Ritter, S. Clark and O. Etzioni. Named entity recognition in tweets: An experimental study. In *Proc. of EMNLP 2011*, 2011.
4. C. Doerhmann. Named entity extraction from the colloquial setting of twitter. In *Research Experiences for Undergraduates - Uni. of Colorado*, 2011.
5. A. S. Nugroho S. K. Endarnoto, S. Pradipta and J. Purnama. Traffic condition information extraction amp; visualization from social media twitter for android mobile application. In *Proc. of ICEEI 2011*, 2011.
6. Mena B. Habib and M. van Keulen. Named entity extraction and disambiguation: The reinforcement effect. In *Proc. of MUD 2011*, 2011.
7. K. Berberich E. L. Kelham G. de Melo J. Hoffart, F. M. Suchanek and G. Weikum. Yago2: Exploring and querying world knowledge in time, space, context, and many languages. In *Proc. of WWW 2011*, 2011.



(a) #Egypt



(b) #Superbowl

(c) user LizzieViolet

Fig. 2: Words clouds for some hashtags and user profiles

8. Peter D. Turney David Nadeau and Stan Matwin. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. In *Proc. of 19th Canadian Conference on Artificial Intelligence*, 2006.
9. I. Bordino H. Frstenau M. Pinkal M. Spaniol B. Taneva S. Thater J. Hoffart, M. A. Yosef and G. Weikum. Robust disambiguation of named entities in text. In *Proc. of EMNLP 2011*, 2011.
10. Trond Grenager Jenny Rose Finkel and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proc. of ACL 2005*, 2005.