Uncertainty Handling in Named Entity Extraction and Disambiguation for Informal Text

Maurice van Keulen and Mena B. Habib

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

Abstract. Social media content represents a large portion of all textual content appearing on the Internet. These streams of user generated content (UGC) provide an opportunity and challenge for media analysts to analyze huge amount of new data and use them to infer and reason with new information. A main challenge of natural language is its ambiguity and vagueness. To automatically resolve ambiguity, the grammatical structure of sentences is used. However, when we move to informal language widely used in social media, the language becomes more ambiguous and thus more challenging for automatic understanding.

Information Extraction (IE) is the research field that enables the use of unstructured text in a structured way. Named Entity Extraction (NEE) is a sub task of IE that aims to locate phrases (mentions) in the text that represent names of entities such as persons, organizations or locations regardless of their type. Named Entity Disambiguation (NED) is the task of determining which correct person, place, event, etc. is referred to by a mention.

The goal of this paper is to provide an overview on some approaches that mimic the human way of recognition and disambiguation of named entities especially for domains that lack formal sentence structure. The proposed methods open the doors for more sophisticated applications based on users' contributions on social media. We propose a robust combined framework for NEE and NED in semi-formal and informal text. The achieved robustness has been proven to be valid across languages and domains and to be independent of the selected extraction and disambiguation techniques. It is also shown to be robust against the informality of the used language. We have discovered a reinforcement effect and exploited it a technique that improves extraction quality by feeding back disambiguation results. We present a method of handling the uncertainty involved in extraction to improve the disambiguation results.

Keywords: Named Entity Extraction, Named Entity Disambiguation, Informal Text, Uncertainty Handling

1 Introduction

Computers cannot understand natural languages like humans do. Our ability to easily distinguish between multiple word meanings is developed in a lifetime of experience. Using the context in which a word is used, a fundamental understanding of syntax and logic, and a sense of the speaker's intention, we understand what another person is telling us or what we read. It is the aim of the Natural Language Processing (NLP) society to mimic the way humans understand natural languages. Although efforts spent for more than 50 years by linguists and computer scientists to get computers to understand human language, there is still long way to go to achieve this goal.

A main challenge of natural language is its ambiguity and vagueness. The basic definition of ambiguity, as generally used in natural language processing, is "*capable of being understood in more than one way*". Scientists try to resolve ambiguity, either semantic or syntactic, based on properties of the surrounding context. Examples include, Part Of Speech (POS) tagging, morphology analysis, Named Entity Recognition (NER), and relations (facts) extraction. To automatically resolve ambiguity, typically the grammatical structure of sentences is used, for instance, which groups of words go together (phrases) and which words are the subject or object of a verb. However, when we move to informal language widely used in social media, the language becomes more ambiguous and thus more challenging for automatic understanding.

The rapid growth in the IT in the last two decades leads to the growth in the amount of information available on the World Wide Web (WWW). Social media content represents a big part of all textual content appearing on the Internet. According to an eMarketer report [1], nearly one in four people worldwide will use social networks in 2013. The number of social network users around the world rose to 1.73 billion in 2013. By 2017, the global social network audience will total 2.55 billion. Twitter as an example of highly active social media network, has 140 million active users publishing over 400 million tweet every day[1].

These streams of user generated content (UGC) provide an opportunity and challenge for media analysts to analyze huge amount of new data and use them to infer and reason with new information. Making use of social media content requires measuring, analyzing and interpreting interactions and associations between people, topics and ideas. An example of a main sector for social media analysis is the area of customer feedback through social media. With so many feedback channels, organizations can mix and match them to best suit corporate needs and customer preferences.

Another beneficial sector is social security. Communications over social networks have helped to put entire nations to action. Social media played a key role in The Arab Spring that started in 2010 in Tunisia. The riots that broke out across England during the summer of 2011 also showed the power of social media. The growing criminality associated with social media has been an alarm to government security agencies. There is a growing demand to automatically monitor the discussions on social media as a source of intelligence. Nowadays, increasing numbers of people within investigative agencies are being deployed to monitor social media. Unfortunately, the existing tools and technologies used are limited because they are based on simple keyword selection and classification instead of reasoning with meaningful information. Furthermore, the processes followed are time and resources consuming. There is also a need for new tools and technologies that can deal with the informal language widely used in social media.

Information Extraction (IE) is the research field that enables the use of such a vast amount of unstructured distributed data in a structured way. IE systems analyze human language in order to extract information about different types of events, entities, or relationships. Named Entity Extraction (NEE) is a sub task of IE that aims to locate

[1] https://blog.twitter.com/2012/twitter-turns-six

phrases (mentions) in the text that represent names of persons, organizations or locations regardless of their type. It differs from the term Named Entity Recognition (NER) which involves both extraction and classification to one of the predefined set of classes. Named Entity Disambiguation (NED) is the task of exploring which correct person, place, event, etc. is referred to by a mention. NEE and NED have become a basic steps of many technologies like Information Retrieval (IR), Question Answering (QA).

Although state-of-the-art NER systems for English produce near-human performance [2], their performance drops when applied to informal text of UGC where the ambiguity increases. It this the aim of this paper to study the interdependency of NEE and NED on the domain of informal text, and to show how one could be used to improve the other and vice versa. We call this potential for mutual improvement, the *reinforcement effect*. It mimics the way humans understand natural language. Natural language processing (NLP) tasks are commonly split into a set of pipelined sub tasks. The residual error produced in any sub task propagates, adversely affecting the end objectives. This is why we believe that back propagation would help improving the overall system quality. We show the benefit of using this *reinforcement effect* on two domains: NEE and NED for toponyms in semi-formal text that represents advertisements for holiday properties; and for arbitrary entity types in informal short text in tweets. We proved that this mutual improvement makes NEE and NED robust across languages and domains. This improvement is also independent on what extractions and disambiguation techniques are used. Furthermore, we developed extraction methods that consider alternatives and uncertainties in text with less dependency on formal sentence structure. This leads to more reliability in cases of informal and noisy UGC text.

2 Examples of Application Domains

Information extraction has applications in a wide range of domains. There are many stakeholders that could benefit from UGC on social media. Here, we give some examples for applications of information extraction:

- Security agencies typically analyze large amounts of text manually to search for information about people involved in criminal or terrorism activities. Social media is a continuously instantly updated source of information. Football hooligans sometimes start their fight electronically on social media networks even before the sport event. Another real life example is the Project X Haren[2]. Project X Haren was an event that started out as a public invitation to a birthday party by a girl on Facebook, but ended up as a gathering of thousands of youths causing riots in the town of Haren, Groningen. Automatic monitoring and gathering of such information could be helpful to take actions to prevent such violent, and destructive behaviors. As an example for real application, we contribute to the TEC4SE project[3]. The aim of the project is to improve the operational decision-making within the security domain by gathering as much information available from different sources (like cameras, police officers on field, or social media posts). Then these information is linked

[2] http://en.wikipedia.org/wiki/Project_X_Haren

[3] http://www.tec4se.nl/

and relationships between different information streams are found. The result is a good overview of what is happening in the field of security in the region. Our contribution to this project is to the enrich Twitter stream messages by extracting named entities at run time. The amount and the nature of the flowing data is beyond the possibility of manually tracking. This is why we need new technologies that is capable of dealing with such huge noisy amounts of data.
- As users become more involved in creating contents in a virtual world, more and more data is generated in various aspects of life for studying user attitudes and behaviors. Social sciences study human behavior by studying their physical space and belongings. Now, it is possible to investigate users by studying their online activities, postings, and behavior in a virtual space. This method can be a replacement for traditional surveys and experiments [3]. Prediction and understanding of the attitudes and behaviors of individuals and groups based on the sentiment expressed within online virtual communities is a natural area of research in the Internet era. To reach this goal, social scientists are in dire need of stronger tools to provide them with the required data for their studies.
- Financial experts always look for specific information to help their decision making. Social media can be a very important source of information about the attitudes and behaviors of stakeholders. In general, if extracted and analyzed properly, the data on social media can lead to useful predictions of certain human related events. Such prediction has great benefits in many realms, such as finance, product marketing and politics [4]. For example, a finance company may want to know the stakeholders' reaction towards some political action. Automatically finding such information from user posts on social media requires special information extraction technologies to analyze the noisy social media streams and capture such information.
- With the fast growth of the Web, search engines have become an integral part of people's daily lives, and users search behaviors are much better understood now. Search based on bag-of-words representation of documents can no longer provide satisfactory results. More advanced information needs such as entity search, and question answering can provide users with better search experience. To facilitate these search capabilities, information extraction is often needed as a pre-processing step to enrich the document with information in structured form.

3 Challenges

NEE and NED in informal text are challenging. Here we summarize the challenges of NEE and NED for tweets as an example of informal text:

- The informal nature of tweets makes the extraction process more difficult. For example, in table 1 case 1, it is hard to extract the mentions (phrases that represent NEs) using traditional NEE methods because of the ill-formed sentence structure. Traditional NEE methods might extract '*Grampa*' as a mention because of it capitalization. Furthermore, it is hard to extract the mention '*Speechless*', which is a name of a song, as it requires further knowledge about '*Lady Gaga*' songs.

Table 1. Some challenging cases for NEE and NED in tweets (NE mentions are written in bold).

Case #	Tweet Content
1	– **Lady Gaga** - **Speechless** live @ **Helsinki** 10/13/2010 http://www.youtube.com/watch?v=yREociHyijk . . . @ladygaga also talks about her Grampa who died recently
2	**Qld** flood victims donate to **Vic** bushfire appeal
3	**Laelith Demonia** has just defeated **liwanu Hird**. Career wins is 575, career losses is 966.
4	Adding **Win7Beta**, **Win2008**, and **Vista** x64 and x86 images to munin. #wds
5	history should show that **bush jr** should be in jail or at least never should have been president
6	RT @BBCClick: Joy! **MS Office** now syncs with **Google Docs** (well, in beta anyway). We are soon to be one big happy (cont) http://tl.gd/73t94u
7	"Even Writers Can Help..An Appeal For **Australian** Bushfire Victims" http://cli.gs/Zs8zL2

- The limited length (140 characters) of tweets forces the senders to provide dense information. Users resort to acronyms to reserve space. Informal language is another way to express more information in less space. All of these problems make both the extraction and the disambiguation processes more complex. For example, in table 1 case 2 shows two abbreviations ('*Qld*' and '*Vic*'). It is hard to infer their entities without extra information.
- The limited coverage of a Knowledge Base (KB) is another challenge facing NED for tweets. According to [5], 5 million out of 15 million mentions on the web cannot be linked to Wikipedia. This means that relying only on a KB for NED leads to around 33% loss in disambiguated entities. This percentage is higher on Twitter because of its social nature where users discuss information about infamous entities. For example, table 1 case 3 contains two mentions for two users on the '*My Second Life*' social network. It is very unlikely that one could find their entities in a KB. However, their profile pages ('`https://my.secondlife.com/laelith.demonia`' and '`https://my.secondlife.com/liwanu.hird`') can be found easily by a search engine.
- Named entity (NE) representation in KB implies another NED challenge. YAGO KB [6] uses Wikipedia anchor text as possible mention representation for named entities. However, there might be more representations that do not appear in Wikipedia anchor text. Either because of misspelling or because of a new abbreviation of the entity. For example, in table 1 case 4, the mentions '*Win7Beta*' and '*Win2008*' do not appear in YAGO KB mention-entity look-up table, although they refer to the entities '`http://en.wikipedia.org/wiki/Windows_7`' and '`http://en.wikipedia.org/wiki/Windows_Server_2008`' respectively.
- The processes of NEE and NED involve degrees of uncertainty. For example, in table 1 case 5, it is uncertain whether the word *jr* should be part of the mention *bush* or not. Same for '*Office*' and '*Docs*' in case 6 which some extractors may miss. Another example, in case 7, it is hard to assess whether '*Australian*' should refer

2-room apartment 55 m2: living/dining room with 1 sofa bed and satellite-TV, exit to the balcony. 1 room with 2 beds (90 cm, length 190 cm). Open kitchen (4 hotplates, freezer). Bath/bidet/WC. Electric heating. Balcony 8 m2. Facilities: telephone, safe (extra). Terrace Club: Holiday complex, 3 storeys, built in 1995 2.5 km from the centre of **Armacao de Pera**, in a quiet position. For shared use: garden, swimming pool (25 x 12 m, 01.04.-30.09.), paddling pool, children's playground. In the house: reception, restaurant. Laundry (extra). Linen change weekly. Room cleaning 4 times per week. Public parking on the road. Railway station "**Alcantarilha**" 10 km. Please note: There are more similar properties for rent in this same residence. Reception is open 16 hours (0800-2400 hrs). Lounge and reading room, games room. Daily entertainment for adults and children. Bar-swimming pool open in summer. Restaurant with Take Away service. Breakfast buffet, lunch and dinner(to be paid for separately, on site). Trips arranged, entrance to water parks. Car hire. Electric cafetiere to be requested in adavance. Beach football pitch. IMPORTANT: access to the internet in the computer room (extra). The closest beach (350 m) is the "**Sehora da Rocha**", **Playa de Armacao de Pera** 2.5 km. Please note: the urbanisation comprises of eight 4 storey buildings, no lift, with a total of 185 apartments. Bus station in **Armacao de Pera** 4 km.

Fig. 1. Example of EuroCottage holiday home descriptions (toponyms in bold).

Table 2. Some challenging cases for toponyms extraction in semi-formal text (toponyms are written in bold).

Case #	Semi-formal Text Samples
1	**Bargecchia** 9 km from **Massarosa**.
2	**Olšova Vrata** 5 km from **Karlovy Vary**.
3	Bus station in **Armacao de Pera** 4 km.
4	Airport 1.5 km (2 planes/day).

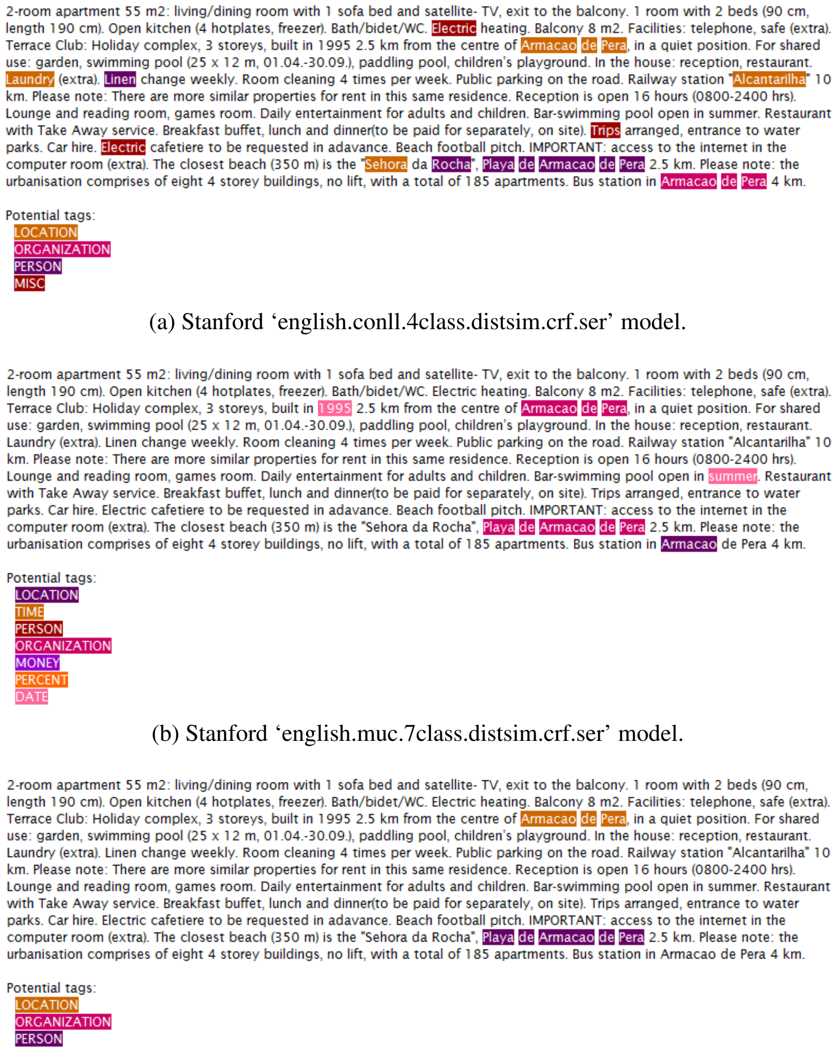
to '`http://en.wikipedia.org/wiki/Australia`' or '`http://en.wikipedia.org/wiki/Australian_people`'[4]. Both might be correct. This is why we believe that it is better to consider possible alternatives in the processes of NEE and NED.

- Another challenge is the freshness of the KBs. For example, the page of '*Barack Obama*' on Wikipedia was created on 18 March 2004. Before that date '*Barack Obama*' was a member of the Illinois Senate and you could find his profile page on '`http://www.ilga.gov/senate/Senator.asp?MemberID=747`'. It is very common on social networks that users talk about some infamous entity who might become later a public figure.
- Informal nature of language used in social media implies many different random representations of the same fact. This adds new challenges to machine learning approaches which need regular patterns for generalization. We need new methods that require less training data and generalize well at the same time.

Semi-formal text is text lacking the formal structure of the language but follows some pattern or format like product descriptions and advertisements. Although semi-formal text involves some regularity in representing information, this regularity implies some challenges.

In table 2, cases 1 and 2 show two examples for true toponyms included in a holiday description. Any machine learning approach uses cases 1 and 2 as training samples will

[4] Some NER datasets consider nationalities as NEs [7].



2-room apartment 55 m2: living/dining room with 1 sofa bed and satellite- TV, exit to the balcony. 1 room with 2 beds (90 cm, length 190 cm). Open kitchen (4 hotplates, freezer). Bath/bidet/WC. Electric heating. Balcony 8 m2. Facilities: telephone, safe (extra). Terrace Club: Holiday complex, 3 storeys, built in 1995 2.5 km from the centre of Armacao de Pera, in a quiet position. For shared use: garden, swimming pool (25 x 12 m, 01.04.-30.09.), paddling pool, children's playground. In the house: reception, restaurant. Laundry (extra). Linen change weekly. Room cleaning 4 times per week. Public parking on the road. Railway station "Alcantarilha" 10 km. Please note: There are more similar properties for rent in this same residence. Reception is open 16 hours (0800-2400 hrs). Lounge and reading room, games room. Daily entertainment for adults and children. Bar-swimming pool open in summer. Restaurant with Take Away service. Breakfast buffet, lunch and dinner(to be paid for separately, on site). Trips arranged, entrance to water parks. Car hire. Electric cafetiere to be requested in adavance. Beach football pitch. IMPORTANT: access to the internet in the computer room (extra). The closest beach (350 m) is the "Sehora da Rocha", Playa de Armacao de Pera 2.5 km. Please note: the urbanisation comprises of eight 4 storey buildings, no lift, with a total of 185 apartments. Bus station in Armacao de Pera 4 km.

Potential tags:
LOCATION
ORGANIZATION
PERSON
MISC

(a) Stanford 'english.conll.4class.distsim.crf.ser' model.

2-room apartment 55 m2: living/dining room with 1 sofa bed and satellite- TV, exit to the balcony. 1 room with 2 beds (90 cm, length 190 cm). Open kitchen (4 hotplates, freezer). Bath/bidet/WC. Electric heating. Balcony 8 m2. Facilities: telephone, safe (extra). Terrace Club: Holiday complex, 3 storeys, built in 1995 2.5 km from the centre of Armacao de Pera, in a quiet position. For shared use: garden, swimming pool (25 x 12 m, 01.04.-30.09.), paddling pool, children's playground. In the house: reception, restaurant. Laundry (extra). Linen change weekly. Room cleaning 4 times per week. Public parking on the road. Railway station "Alcantarilha" 10 km. Please note: There are more similar properties for rent in this same residence. Reception is open 16 hours (0800-2400 hrs). Lounge and reading room, games room. Daily entertainment for adults and children. Bar-swimming pool open in summer. Restaurant with Take Away service. Breakfast buffet, lunch and dinner(to be paid for separately, on site). Trips arranged, entrance to water parks. Car hire. Electric cafetiere to be requested in adavance. Beach football pitch. IMPORTANT: access to the internet in the computer room (extra). The closest beach (350 m) is the "Sehora da Rocha", Playa de Armacao de Pera 2.5 km. Please note: the urbanisation comprises of eight 4 storey buildings, no lift, with a total of 185 apartments. Bus station in Armacao de Pera 4 km.

Potential tags:
LOCATION
TIME
PERSON
ORGANIZATION
MONEY
PERCENT
DATE

(b) Stanford 'english.muc.7class.distsim.crf.ser' model.

2-room apartment 55 m2: living/dining room with 1 sofa bed and satellite- TV, exit to the balcony. 1 room with 2 beds (90 cm, length 190 cm). Open kitchen (4 hotplates, freezer). Bath/bidet/WC. Electric heating. Balcony 8 m2. Facilities: telephone, safe (extra). Terrace Club: Holiday complex, 3 storeys, built in 1995 2.5 km from the centre of Armacao de Pera, in a quiet position. For shared use: garden, swimming pool (25 x 12 m, 01.04.-30.09.), paddling pool, children's playground. In the house: reception, restaurant. Laundry (extra). Linen change weekly. Room cleaning 4 times per week. Public parking on the road. Railway station "Alcantarilha" 10 km. Please note: There are more similar properties for rent in this same residence. Reception is open 16 hours (0800-2400 hrs). Lounge and reading room, games room. Daily entertainment for adults and children. Bar-swimming pool open in summer. Restaurant with Take Away service. Breakfast buffet, lunch and dinner(to be paid for separately, on site). Trips arranged, entrance to water parks. Car hire. Electric cafetiere to be requested in adavance. Beach football pitch. IMPORTANT: access to the internet in the computer room (extra). The closest beach (350 m) is the "Sehora da Rocha", Playa de Armacao de Pera 2.5 km. Please note: the urbanisation comprises of eight 4 storey buildings, no lift, with a total of 185 apartments. Bus station in Armacao de Pera 4 km.

Potential tags:
LOCATION
ORGANIZATION
PERSON

(c) Stanford 'english.all.3class.distsim.crf.ser' model.

Fig. 2. Results of Stanford NER models applied on semi-formal text of holiday property description.

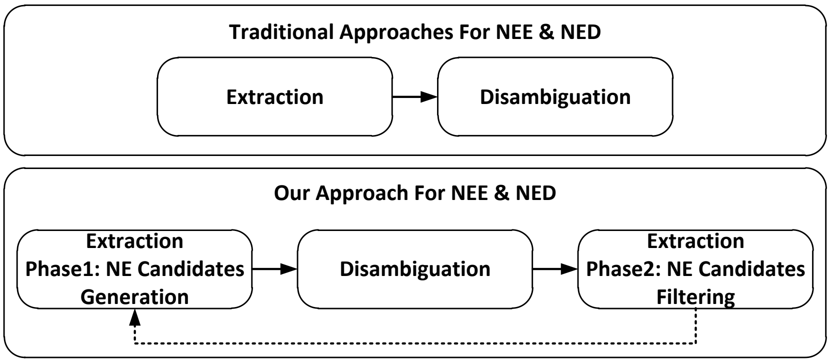

Traditional Approaches For NEE & NED
Extraction
Disambiguation
Our Approach For NEE & NED
Extraction Phase1: NE Candidates Generation
Disambiguation
Extraction Phase2: NE Candidates Filtering

Fig. 3. Traditional approaches versus our approach for NEE and NED.

annotate ‘*Airport*’ as a toponym following the same pattern of having a capitalized word followed by a number and the word ‘*km*’. Furthermore, the state-of-the-art approaches performs poorly on this type of text. Figure 2 shows the results of the application of three of the leading Stanford NER models[5] on a holiday property description text (see figure 1). Regardless of NE classification, even the extraction (determining if a phrase represents a NE or not) is performing poorly. Problems vary between *a*) extracting false positives (like ‘*Electric*’ and ‘*Trips*’ in figure 2a); or *b*) missing some true positives (like ‘*Sehora da Rocha*’ in figures 2b and 2c); or *c*) partially extracting the NE (like ‘*Sehora da Rocha*’ in figures 2a and ‘*Armacao de Pera*’ in figure 2b).

4 General Approach

Natural language processing (NLP) tasks are commonly composed of a set of chained sub tasks that form the processing pipeline. The residual error produced in these sub tasks propagates, affecting the final process results. In this paper we are concerned with NEE and NED which are two common processes in many NLP applications.

Let us first formalize the NEE and NED problems. Given a sequence of words (tokens) $\{w\} = \{w_1, w_2, ..w_n\}$, NEE is the process of identifying sub-lists of words that represents mentions of NEs where mention $\{m\} = \{w_i, w_{i+1}, ..w_j\}$. The process of NED is to assign m to one of its possible entities $\{e\} = \{e_1, e_2, ..e_n\}$. The final output of the two processes is list of pairs (m, e). Figure 4 shows the formalization of the two problems.

We claim that feedback derived from disambiguation would help in improving the extraction and hence the disambiguation. This is the same way we as humans understand text. The capability to successfully understand language requires one to acquire a range of skills including syntax, semantics, and an extensive vocabulary. We try to mimic a human’s way of reasoning to solve the NEE and NED problems. Consider the

[5] `http://nlp.stanford.edu:8080/ner/process`

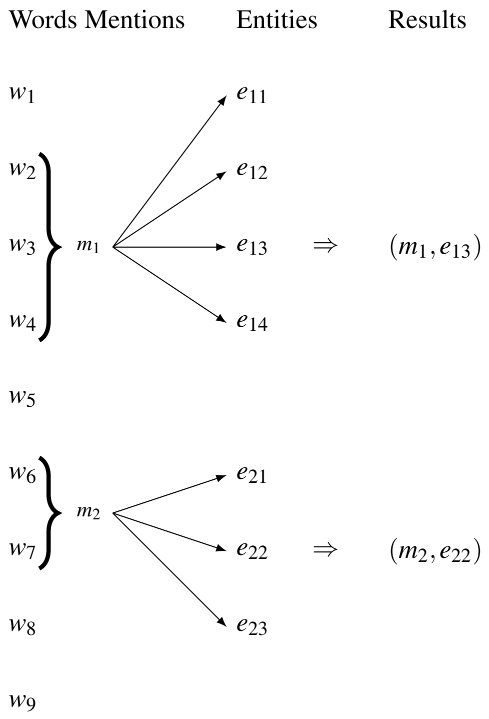


Fig. 4. Formalization of NEE and NED problems

tweet in table 1 case 1. One would use syntax knowledge to recognize ‘10/13/2010’ as a date. Furthermore, prior knowledge enables one to recognize ‘*Lady Gaga*’ and ‘*Helsinki*’ as a singer name and location name respectively or at least as names if one doesn’t know exactly what they refer to. However, the term ‘*Speechless*’ involves some ambiguity as it could be an adjective and also could be a name. A feedback clue from ‘*Lady Gaga*’ would increase one’s certainty that it refers to a song. Even without knowing that ‘*Speechless*’ is a song of ‘*Lady Gaga*’, there are sufficient clues to guess with quite high probability that it is a song. The pattern ‘live @’ in association with disambiguating ‘*Lady Gaga*’ as a singer name and ‘*Helsinki*’ as a location name, leads to infer ‘*Speechless*’ as a song.

Although the logical order for a traditional Information Extraction (IE) system is to complete the extraction process before commencing the disambiguation, we start with an initial phase of extraction which aims to achieve high recall (find as many reasonable mention candidates as possible) then we apply the disambiguation for all the extracted possible mentions. Finally we filter those extracted mention candidates

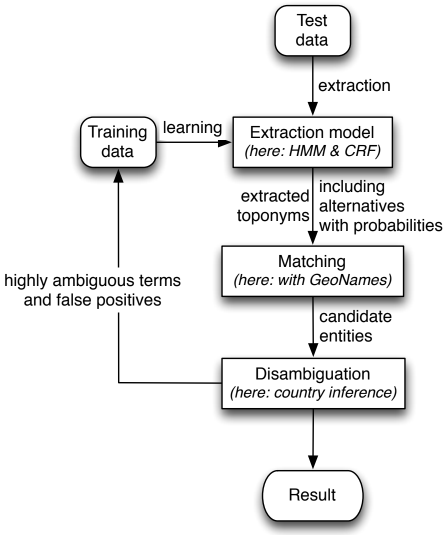


Fig. 5. Extraction and disambiguation approach

into true positives and false positives using features (clues) derived from the results of the disambiguation phase such as KB information and entity coherency. Figure 3 illustrates our general approach.

Unlike NER systems which extract entities mentions and assign them to one of the predefined categories (like location, person, organization), we focus first on extracting mentions regardless of their categories. We leave this classification to the disambiguation step which links the mention to its real entity.

The potential of this order is that the disambiguation step can give extra clues (such as entity-context similarity and entity-entity coherency) about each NE candidate. This information can help in the decision whether the candidate is a true NE or not.

The general principal we claim is that NED could be very helpful in improving the NEE process. For example, consider the tweet in case 1 in table 1. It is uncertain, even for humans, to recognize '*Speechless*' as a song name without having prior information about songs of '*Lady Gaga*'. Our approach is able to solve such problematic cases of named entities.

5 Case Study 1: Toponym Extraction and Disambiguation in Semi-formal Text

The task we focus on is to extract toponyms from EuroCottage holiday home descriptions[6] (an example is shown in figure 1) and use them to infer the country where the

[6] www.eurocottage.com

holiday property is located. We use this country inference task as a representative example of disambiguating extracted toponyms.

We propose an entity extraction and disambiguation approach based on uncertain annotations. The general approach illustrated in Figure 5 has the following steps:

1. Prepare training data by manually annotating named entities.
2. Use the training data to build a statistical extraction model.
3. Apply the extraction model on test data and training data.
4. Match the extracted named entities against one or more gazetteers.
5. Use the toponym entity candidates for the disambiguation process.
6. Evaluate the extraction and disambiguation results for the training data. Automatically find a list of highly ambiguous named entities and false positives that affect the disambiguation results and use it to re-train the extraction model.
7. The steps from 2 to 6 are repeated automatically until there is no improvement any more in either the extraction or the disambiguation.

5.1 Toponym Extraction

For toponym extraction, we developed two statistical named entity extraction modules[7], one based on Hidden Markov Models (HMM) and one based on Conditional Ramdom Fields (CRF).

The goal of HMM [8] is to find the optimal tag sequence (in our case, whether the word is assigned to toponym tag or not) $T = t_1, t_2, t_3, ..., t_n$ for a given word sequence $W = w_1, w_2, w_3 ..., w_n$ that maximizes $P(T \mid W)$.

Conditional Random Fields (CRF) can model overlapping, non-independent features [9]. Here we used a linear chain CRF, the simplest model of CRF.

5.2 Extraction Modes of Operation

We used the extraction models to retrieve sets of annotations in two ways:

- **First-Best:** In this method, we only consider the first most likely set of annotations that maximize the probability $P(T \mid W)$ for the whole text. This method does not assign a probability for each individual annotation, but only to the whole retrieved set of annotations.
- **N-Best:** This method returns a top-25 of possible alternative hypotheses for terms annotations in order of their estimated likelihoods $p(t_i|w_i)$. The confidence scores are assumed to be conditional probabilities of the annotation given an input token.

5.3 Toponym Disambiguation

For the toponym disambiguation task, we only select those toponyms annotated by the extraction models that match a reference in GeoNames. We furthermore use an adapted version of the clustering approach of [10] to disambiguate to which entity an extracted toponym actually refers.

[7] We made use of the *lingpipe* toolkit for development: `http://alias-i.com/lingpipe`

Table 3. Effectiveness of the disambiguation process for First-Best and N-Best methods in the extraction phase.

	HMM	CRF
First-Best	62.59%	62.84%
N-Best	68.95%	68.19%

Table 4. Effectiveness of the disambiguation after iteration of refinement.

	HMM	CRF
No Filtering	68.95%	68.19%
1st Iteration	73.28%	68.44%

5.4 Handling Uncertainty of Annotations

Instead of giving equal contribution to all toponyms, we take the uncertainty in the extraction process into account to include the confidence of the extracted toponyms. In this way terms which are more likely to be toponyms have a higher contribution in determining the country of the document than less likely ones.

5.5 Improving Certainty of Extraction

In despite of the abovementioned improvement, the extraction probabilities are not accurate and reliable all the time. Some extraction models retrieve some false positive toponyms with high confidence probabilities. This is where we take advantage of the reinforcement effect. To be more precise. We introduce another class in the extraction model called ‘highly ambiguous’ and annotate those terms in the training set with this class that the disambiguation process finds more than τ countries for documents that contain this term.

The extraction model is subsequently re-trained and the whole process is repeated without any human interference as long as there is improvement in extraction and disambiguation process for the training set. The intention is that the extraction model learns to avoid prediction of terms to be toponyms when they appear to confuse the disambiguation process.

5.6 Experimental Results

Here we present the results of experiments with the presented methods of extraction and disambiguation applied to a collection of holiday properties descriptions. The data set consists of 1579 property descriptions for which we constructed a ground truth by manually annotating all toponyms.

Experiment 1: Effect of Extraction with Confidence Probabilities Table 3 shows the percentage of holiday home descriptions for which the correct country was successfully inferred. We can see that the **N-Best** method outperforms the **First-Best** method for both HMM and CRF models. This supports our claim that dealing with alternatives along with their confidences yields better results.

Table 5. Effectiveness of the extraction process after iteration of refinement.

	HMM				CRF		
	Pre.	**Rec.**	**F1**		**Pre.**	**Rec.**	**F1**
No Filtering	0.3584	0.8517	0.5045	**No Filtering**	0.6969	0.7136	0.7051
1st Iteration	0.7667	0.5987	0.6724	**1st Iteration**	0.6989	0.7131	0.7059

Experiment 2: Effect of Extraction Certainty Enhancement Tables 4 and 5 show the effectiveness of the disambiguation and the extraction processes respectively before and after one iteration of refinement. We can see an improvement in HMM extraction and disambiguation results. The initial HMM results showed a high recall rate with a low precision. In spite of this, our approach managed to improve precision through iteration of refinement. The refinement process is based on removing highly ambiguous toponyms resulting in a slight decrease in recall and an increase in precision. In contrast, CRF started with high precision which could not be improved by the refinement process.

6 Case Study 2: Named Entity Extraction and Disambiguation Approach for Tweets

In this case study, we present a combined approach for NEE and NEL for tweets with an application on #Microposts 2014 challenge [11]. Although the logical order for such system is to do extraction first then the disambiguation, we start with an extraction phase which aims to achieve high recall (find as much NE candidates as possible). Then we apply disambiguation for all the extracted mentions. Finally, we filter those extracted NE candidates into true positives and false positives using features derived from the disambiguation phase in addition to other word shape and KB features. The potential of this order is that the disambiguation step gives extra information about each NE candidate that may help in the decision whether or not this candidate is a true NE. Figure 3 shows our system architecture versus traditional one.

6.1 NE Candidates Generation

For this task, we unionize the output of the following candidates generation methods:

- **Tweet Segmentation**: Tweet text is segmented using the segmentation algorithm described in [12]. Each segment is considered a NE candidate.
- **KB Lookup**: We scan all possible n-grams of the tweet against the mentions-entities table of DBpedia. N-grams that matches a DBpedia mention are considered NE candidates.
- **Regular Expressions**: We used regular expressions to extract numbers, dates and URLs from the tweet text.

6.2 NE Linking

Our NEL approach is composed of three steps; matcher, feature extractor, and SVM ranker.

- **Matcher**: This module takes each extracted mention candidate and looks for its Wikipedia reference candidates on DBpedia. Furthmore, for those mention candidates which don't have reference candidates in DBpedia, we use Google Search API to find possible Wikipedia pages for these mentions. This search helps to find references for misspelled or concatenated mentions like '*justinbieber*' and '*106andpark*'.
- **Feature Extractor**: This module is responsible for extracting a set of contextual and URL features for each candidate Wikipedia page as described in [13]. These features give indicators on how likely the candidate Wikipedia page could be a representative to the mention.
- **SVM Ranker**: After extracting the aforementioned set of features, SVM classifier is trained to rank candidate Wikipedia pages of a mention. For the challenge, we pick the page on the 1st order as a reference for the mention. The DBpedia URI is then generated from the selected Wikipedia URL.

6.3 NE Candidates Filtering

After generating the candidates list of NE, we apply our NE linking approach to disambiguate each extracted NE candidate. After the linking phase, we use SVM classifier to predict which candidates are true positives and which ones are not. We use the following set of features for each NE candidate to train the SVM:

- **Shape Features**: If the NE candidate is initially or fully capitalized and if it contains digits.
- **Probabilistic Features**:
  - The joint and the conditional probability of the candidate obtained from Microsoft Web N-Gram services.
  - The stickiness of the candidate as described in [12].
  - The candidate's frequency over around 5 million tweets[8].
- **KB Features**:
  - If the candidate appears in WordNet.
  - If the candidate appears as a mention in DBpedia KB.
- **Disambiguation Features**:
  - All the features used in the linking phase as described in [13]. We used only the feature set for the first top ranked entity page selected for the given NE candidate.

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

Table 6. Linking Results

	Percentage
Our Linking Approach	**70.98%**
AIDA Cocktail	56.16%
AIDA Prior	55.63%

Table 7. Extraction Results

	Pre.	Rec.	F1
Candidates Generation	0.120	**0.945**	0.214
Candidates Filtering (SVM)	**0.722**	0.544	0.621
CRF	0.660	0.568	0.611
Final Set Generation	0.709	0.706	**0.708**
Stanford NER	0.716	0.392	0.507

Table 8. Extraction and Linking Results

	Pre.	Rec.	F1
Extraction + Linking	**0.533**	**0.534**	**0.534**
Stanford + AIDA	0.509	0.279	0.360

6.4 Final NE Set Generation

Beside the SVM, we also train a CRF model for NEE. We used the CRF model described in [14]. To generate the final NE set, we take the union of the CRF annotation set and SVM results, after removing duplicate extractions, to get the final set of annotations. We tried two methods to resolve overlapped mentions. In the first method (used in UTwente_Run1.tsv), we select the mention that appears in Yago KB [6]. If both mentions appear in Yago or both don't, we select the one with the longer length. In the second method (used in UTwente_Run2.tsv), we select only the mention with the longer length among the two overlapped mentions. The results shown in the next section are the results of the first method.

The idea behind this unionization is that SVM and CRF work in a different way. The former is a distance based classifier that uses numeric features for classification which CRF can not handle, while the latter is a probabilistic model that can naturally consider state-to-state dependencies and feature-to-state dependencies. On the other hand, SVM does not consider such dependencies. The hybrid approach of both makes use of the strength of each.

6.5 Experimental Results

In this section we show our experimental results of the proposed approaches on the challenge training data [11] in contrast with other competitors. All our experiments are done through a 4-fold cross validation approach for training and testing. Table 6 shows the results of '**Our Linking Approach**' presented in section 6.2, in comparison with two modes of operation of AIDA [15]. The first mode is '**AIDA Cocktail**' which makes use of several ingredients: the prior probability of an entity being mentioned, the similarity between the context of the mention in the text and an entity, as well as the coherence among the entities. While the second mode is '**AIDA Prior**' which makes use only of the prior probability. The results show the percentage of finding the correct entity of the ground truth mentions. Table 7 shows the NEE results along the extraction process phases in contrast with '**Stanford NER**' [16]. Finally, table 8 shows our final results of both extraction and entity linking in comparison with our competitor

(‘**Stanford + AIDA**’) where ‘**Stanford NER**’ is used for NEE and ‘**AIDA Cocktail**’ is used for NEL.

7 Future Research Directions

Although many machine learning and fuzzy techniques abound, some aspects often remain absolute: extraction rules absolutely recognize and annotate a phrase or not, only a top item from a ranking is chosen for a next phase, etc. We envision an approach that *fundamentally* treats annotations and extracted information as uncertain throughout the process. We humans happily deal with doubt and misinterpretation every day, why shouldn’t computers?

We envision developing information extractors ‘Sherlock Holmes style’ — *“when you have eliminated the impossible, whatever remains, however improbable, must be the truth”* — by adopting the principles and requirements below.

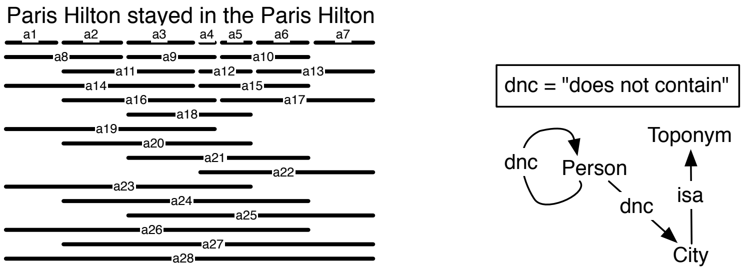
- Annotations are uncertain, hence we process both annotations as well as information about the uncertainty surrounding them.
- We have an unconventional conceptual starting point, namely not “no annotations” but “there is no knowledge hence anything is possible”. Fig.6a shows all possible annotations for an example sentence for one entity type.
- A developer gradually and interactively defines an ontology with positive and negative knowledge about the correctness of certain (combinations of) annotations. At each iteration, added knowledge is immediately applied improving the extraction result until the result is good enough (see also [17]).
- Storage, querying and manipulation of annotations should be scalable. Probabilistic databases are an attractive technology for this.

Basic forms of knowledge are the entity types one is interested in and declarations like τ_1 —*dnc*— τ_2 (no subphrase of a τ_1-phrase should be interpreted as τ_2, e.g, Person —*dnc*— City). See Fig.6b for a small example. We also envision application of background probability distributions, uncertain rules, etc. We hope these principles and forms of knowledge also allow for more effective handling of common problems (e.g., “you” is also the name of a place; should “Lake Como” or “Como” be annotated as a toponym).

7.1 Uncertain annotation model

An *annotation* $a = (b, e, \tau)$ declares a phrase φ_e^b from b to e to be interpreted as entity type τ. For example, a_8 in Fig. 6a declares $\varphi =$ “Paris Hilton” from $b = 1$ to $e = 2$ to be interpreted as type $\tau =$ Person. An *interpretation* $I = (A, \mathcal{U})$ of a sentence s consists of an annotation set A and a structure $\mathcal{U}$ representing the uncertainty among the annotations. In the sequel, we discuss what $\mathcal{U}$ should be, but for now view it as a set of random variables (RVs) R with their dependencies.

Rather unconventionally, we don’t start with an empty A, but with a ‘no knowledge’ point-of-view where any phrase can have any interpretation. So our initial A is $\{a \mid a = (b, e, \tau) \wedge \tau \in T \wedge \varphi_e^b \text{ is a phrase of } s\}$ where T is the set of possible types.



(a) All possible annotations for the example sentence (b) Small example ontology

Fig. 6. Example sentence and NEE ontology

With T finite, A is also finite. More importantly, $|A| = O(klt)$ where $k = |s|$ is the length of s, l is the maximum length phrases considered, and $t = |T|$. Hence, A grows linearly in size with each. In the example of Fig.6a, $T = \{\mathsf{Person}, \mathsf{Toponym}, \mathsf{City}\}$ and we have $28 \cdot |T| = 84$ annotations. Even though we envision a more ingenious implementation, no probabilistic database would be severely challenged by a complete annotation set for a typical text field.

7.2 Knowledge application is conditioning

We explain how to 'apply knowledge' in our approach by means of the example of Fig.6, i.e., with our A with 84 (possible) annotations and an ontology only containing Person, Toponym, and City. Suppose we like to add the knowledge Person—*dnc*—City. The effect should be the removal of some annotations and adjustment of the probabilities of the remaining ones.

An initial promising idea is to store the annotations in an uncertain relation in a probabilistic database, such as MayBMS [18]. In MayBMS, the existence of each tuple is determined by an associated world set descriptor (wsd) containing a set of RV assignments from a world set table (see Fig.7). RVs are assumed independent. For example, the 3rd annotation tuple only exists when $x_8^1 = 1$ which is the case with a probability of 0.8. Each annotation can be seen as a probabilistic event, which are all independent in our starting point. Hence, we can store A by associating each annotation tuple a_i^j with one boolean RV x_i^j. Consequently, the database size is linear with $|A|$.

annotations

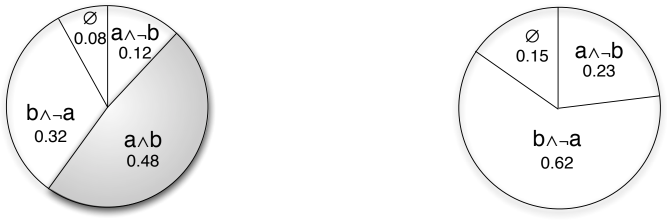
	b	e	type	...	wsd
a_1^1	1	1	Person	...	$\{x_1^1 = 1\}$
a_1^2	1	1	City	...	$\{x_1^2 = 1\}$
a_8^1	1	2	Person	...	$\{x_8^1 = 1\}$
	...	...	...	...	

world_set

x	v	P
x_1^1	0	0.4
x_1^1	1	0.6
x_1^2	0	0.7
x_1^2	1	0.3
x_8^1	0	0.2
x_8^1	1	0.8
...	...	...

Fig. 7. Initial annotation set stored in a probabilistic database (MayBMS-style)

Adding knowledge such as Person —*dnc*— City means that certain RVs become dependent and that certain combinations of RV assignments become impossible. Let us focus on two individual annotations a_1^2 ('"Paris" is a City) and a_8^1 ("Paris Hilton" is a Person). These two annotations become mutually exclusive. The process of adjusting the probabilities is called *conditioning* [19]. It boils down to redistributing the



(a) Annotations a and b independent with probabilities $P(a) = 0.6$ and $P(b) = 0.8$

(b) a and b conditioned to be mutually exclusive ($a \wedge b$ not possible)

Fig. 8. Defining a and b to be mutually exclusive means conditioning the probabilities.

remaining probability mass. Fig.8 illustrates this for $a = a_1^2$ and $b = a_8^1$. The remaining probability mass is $1 - 0.48 = 0.52$. Hence, the distribution of this mass over the remaining possibilities is $P(a \wedge \neg b) = \frac{0.12}{0.52} \approx 0.23$, $P(b \wedge \neg a) = \frac{0.32}{0.52} \approx 0.62$, and $P(\emptyset) = P(\neg a \wedge \neg b) = \frac{0.08}{0.52} \approx 0.15$.

A first attempt is to replace x_1^2 and x_8^1 with one fresh three-valued RV x' with the probabilities just calculated, i.e., $\mathrm{wsd}(a_1^2) = \{x' = 1\}$ and $\mathrm{wsd}(a_8^1) = \{x' = 2\}$ with $P(x' = 0) = 0.15$, $P(x' = 1) = 0.23$, and $P(x' = 2) = 0.62$. Unfortunately, since annotations massively overlap, we face a combinatorial explosion. For this rule, we end up with one RV with up to $2^{2 \cdot 28} = 2^{56} \approx 7 \cdot 10^{16}$ cases.

Solution directions What we are looking for in this paper is a structure that is expressive enough to capture all dependencies between RVs and at the same time allowing for scalable processing of conditioning operations. The work of [19] represents dependencies resulting from queries with a tree of RV assignments. We are also investigating the shared correlations work of [20].

References

1. Social networking reaches nearly one in four around the world.
2. Nancy A. Chinchor. Proceedings of the Seventh Message Understanding Conference (MUC-7) named entity task definition. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, page 21 pages, Fairfax, VA, April 1998. version 3.5, http://www.itl.nist.gov/iaui/894.02/related_projects/muc/.
3. Mohammad-Ali Abbasi, Sun-Ki Chai, Huan Liu, and Kiran Sagoo. Real-world behavior analysis through a social media lens. In *Proceedings of the 5th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction*, SBP'12, pages 18–26, Berlin, Heidelberg, 2012. Springer-Verlag.
4. Sheng Yu and Subhash Kak. A survey of prediction using social media. *CoRR*, abs/1203.1647, 2012.
5. Thomas Lin, Mausam, and Oren Etzioni. Entity linking at web scale. In *Proc. of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 84–88, 2012.

6. J. Hoffart, F. Suchanek, K. Berberich, E. 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*, pages 229–232, 2011.
7. Amparo Elizabeth Cano Basave, Andrea Varga, Matthew Rowe, Milan Stankovic, and Aba-Sah Dadzie. Making sense of microposts (#msm2013) concept extraction challenge. In *Making Sense of Microposts (#MSM2013) Concept Extraction Challenge*, pages 1–15, 2013.
8. Asif Ekbal and Sivaji Bandyopadhyay. A hidden markov model based named entity recognition system: Bengali and hindi as case studies. In *Pattern Recognition and Machine Intelligence*, volume 4815, pages 545–552. 2007.
9. Hanna Wallach. Conditional random fields: An introduction. Technical Report MS-CIS-04-21, Department of Computer and Information Science, University of Pennsylvania, 2004.
10. M. B. Habib and M. van Keulen. Named entity extraction and disambiguation: The reinforcement effect. In *Proc. of MUD 2011, Seatle, USA*, pages 9–16, 2011.
11. Amparo Elizabeth Cano, Giuseppe Rizzo, Andrea Varga, Matthew Rowe, Milan Stankovic, and Aba-Sah Dadzie. #microposts2014 neel challenge: Measuring the performance of entity linking systems in social streams. In *Proc. of the #Microposts2014 NEEL Challenge*, 2014.
12. Chenliang Li, Jianshu Weng, Qi He, Yuxia Yao, Anwitaman Datta, Aixin Sun, and Bu-Sung Lee. Twiner: named entity recognition in targeted twitter stream. In *SIGIR*, pages 721–730, 2012.
13. Mena B. Habib and Maurice van Keulen. A generic open world named entity disambiguation approach for tweets. In *Proceedings of the 5th International Conference on Knowledge Discovery and Information Retrieval, KDIR 2013, Vilamoura, Portugal*, pages 267–276, Portugal, September 2013. SciTePress.
14. Mena Habib, Maurice Van Keulen, and Zhemin Zhu. Concept extraction challenge: University of Twente at #msm2013. In *Making Sense of Microposts (#MSM2013) Concept Extraction Challenge*, pages 17–20, 2013.
15. Mohamed Amir Yosef, Johannes Hoffart, Ilaria Bordino, Marc Spaniol, and Gerhard Weikum. Aida: An online tool for accurate disambiguation of named entities in text and tables. *PVLDB*, pages 1450–1453, 2011.
16. Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL*, pages 363–370, 2005.
17. M. van Keulen and A. de Keijzer. Qualitative effects of knowledge rules and user feedback in probabilistic data integration. *The VLDB Journal*, 18(5):1191–1217, 2009.
18. J. Huang, L. Antova, C. Koch, and D. Olteanu. MayBMS: a probabilistic database management system. In *Proc. of the 35th SIGMOD Int'l Conf. on Management Of Data, Providence, Rhode Island*, pages 1071–1074, 2009.
19. C. Koch and D. Olteanu. Conditioning probabilistic databases. *Proc. VLDB Endow.*, 1(1):313–325, 2008.
20. P. Sen, A. Deshpande, and L. Getoor. Exploiting shared correlations in probabilistic databases. *Proc. VLDB Endow.*, 1(1):809–820, 2008.