Using Linked Open *Data sources* for Entity Disambiguation

Esther Villar Rodríguez
OPTIMA Unit
TECNALIA
esther.villar@tecnalia.com

Ana I. Torre Bastida
OPTIMA Unit
TECNALIA
isabel.torre@tecnalia.com

Ana García Serrano
ETSI Informática
UNED
agarcia@lsi.uned.es

Marta González Rodríguez
OPTIMA Unit
TECNALIA
marta.gonzalez@tecnalia.com

Abstract. Within the framework of RepLab 2013, the filtering task try to discover if one tweet is related to one certain entity or not. Our work tries to take advantages of the Web of Data in order to create a context for every entity, extracted from the available Linked Data Sources. The context in Natural Language Processing (NLP) is the outstanding issue able to distinguish the contained semantics in a message by analyzing the frame in which the words are embedded.

1 Introduction

Reputation management is used by companies (or individuals) to monitor the public opinions aiming at maintaining a good brand image. The first step is to establish the correct relation between the opinion (text) and the entity with some grade of confidence. This is the objective of the filtering task in RepLab, an initiative promoted by the EU project Limosine focused on the ability to process and understand which the strengths and weaknesses of one entity are, based on users opinions (http://www.limosine-project.eu/events/replab2013).

Nowadays there is a large amount of available information on the web, such as web pages, social media data (tweets, facebook and others) or blogs. All of them mention different entities, such as locations, characters, organizations ... The problem appears when a name refers to an entity that have several meanings, for example the song “Munich” of the music group “Editors” and the German city of the same name.

For this filtering task, our system uses an approach based on the semantic context of an entity. The goal of this work is to create a description of an entity that will help to achieve a, enough complex, semantic context to execute a successful disambiguation. The data sources from where entity descriptions are extracted make up the Web of Data, specifically the Linked Open Data Cloud.

In this respect, our research has been developed in the frame of Linked Open Data paradigm that is a recommended best practice for exposing, sharing, and connecting pieces of data, information, and knowledge on the Semantic Web using URIs and RDF. Due to activities like this, the volume of data in RDF format is continuously growing, building the known “Web of Data”, which today is the largest free available knowledge base. Its size, open access, semantic character and continued growth led us

to choose it as our information provider for context generation during the filtering task. This process is carried out using different semantic technologies: such as the SPARQL query language, the ontology definition languages, like RDF, RDFS or OWL and RDF repositories (SPARQL endpoints).

The main contribution of this paper is the definition of a system that achieves high precision/sensitivity in the tasks of filtering by entities, using semantic technologies to extract context information from the Linked Open Data Sources, as are going to be presented in the following.

2 Proposed Approach

In this section, we introduce our approach for filtering entities on tweets. Our procedure uses the semantic context of the analyzed entities, and compares it versus the terms contained in the tweet.

First of all the tweets are preprocessed, extracting the terms involved on them. These terms are the input for a second phase, where equivalent available forms for the concepts are obtained by the Stylus[1] tool. When all the possible forms of a term are calculated, the last step consists on generating a semantic context by querying different data sources (modeled by a set of ontologies) that the Linked Open Data Cloud provides to us.

The section is divided into four parts. First we introduce the motivation to use a semantic context for entities filtering, later we explain the preprocessing phase of the system. In the third subsection we include a description of the generation of the context and finally we resume our filtering algorithm.

2.1 Motivation

The main reasons to utilize a semantic context for discovering the relatedness of tweets with the different entities processed are the next two ones:

- **Powerful modeling and rendering capabilities offered by ontologies**. The ontologies allow us to capture the concepts and properties of a particular domain. It is possible to draw a conceptual structure as detailed and complete as necessary. Furthermore, the process of describing ontologies is simple and straightforward, generating an independent and autonomous model.
- **The amount of free available semantically represented data (RDF, RDFS, and OWL) into the linked Open Data Cloud.** Nowadays the amount of information available in RDF format is huge. The Linked Data paradigm has promoted the Web of Data, formed by the Linked Datasets. This makes possible that any user can obtain information about heterogeneous domains. Consulting these datasets through technologies such as SPARQL or RDF Dumps, the user can get semantic information about concepts or entities using modeling ontologies of different repositories.

[1] http://www.daedalus.es/productos/

To illustrate the benefits which can give us a semantic context, there is an example here: In the field of reputation analysis of music groups, we consider the following tweet and the studied entity is the group **U2***:*

*"Enjoy a lot in the last concert of the singer **Bono**"*

At first there is nothing in the tweet that can help us to relate it in a syntactically manner with "U2". But using the semantic context generated for this group of music, we know that among the members of the group, their vocalist is Paul David Hewson, better known by its artistic name "Bono".

The semantic context allows us to build relationships that lead to U2 from Bono and in this way we deduce that a tweet talking about Bono entity, it also does indirectly about the U2 entity and therefore the tweet and the second entity are related.

For the extraction of the necessary information for the generation of context, we have considered the data sources and ontologies shown in the table 1:

Dataset Name	Domain	Sparql Endpoint
DBPEDIA	General domain.	http://dbpedia.org/sparql
MusicBranz	Music domain	http://dbtune.org/musicbrainz/sparql
EventMedia	Media domain	http://eventmedia.eurecom.fr/sparql
ZBW Economics	Economic domain	http://zbw.eu/beta/sparql/
Swetodblp	University bibliography domain	http://datahub.io/dataset/sweto-dblp
DBLP	University bibliography domain	http://dblp.rkbexplorer.com/sparql/

Table 1. Available datasets and ontologies into Linked Open Data cloud for the studied domains

The table shows datasets of the four domains used in the task of filtering: music, university, banks and automobile.

2.2 Preprocessing of tweets

This task is in charge of extracting the terms contained in the tweet. Before that the terms are compared with the entities, they need to be pre-processed to remove the typical characteristics of the tweets (#) which can affect the precision.

This preprocessing has three main tasks (fig 1):

1. Removing URL. URLs in this approach are eliminated, because they do not provide value for the comparison in a the semantic context. In future work, we will try to replace the URLs by entities that represent them, and we could even consider the various relationships / links inside the web page that are identified by the URL under study.
2. Removing mentions. For our task, the mentions are not interesting at this moment, because the relationship between them and the content of the tweet is irrelevant.

3. Transforming hashtags. Hashtags are topics with relevance that somehow summarize the content of the tweet; therefore we parse their terms so they can be treated by subsequent processes.

Fig. 1. Preprocessing tweets

2.3 Context generation

The context represents the related concepts/entities and the kind of relationships between them. In our approach, we generate a context for each needed entity.

The information to build the context is obtained from the datasets shown in Table 1. Depending on the type of entity, we perform different types of questions to a specific domain (music, banks, automobiles, universities). These queries are constructed from the different forms or variants that represent the entity.

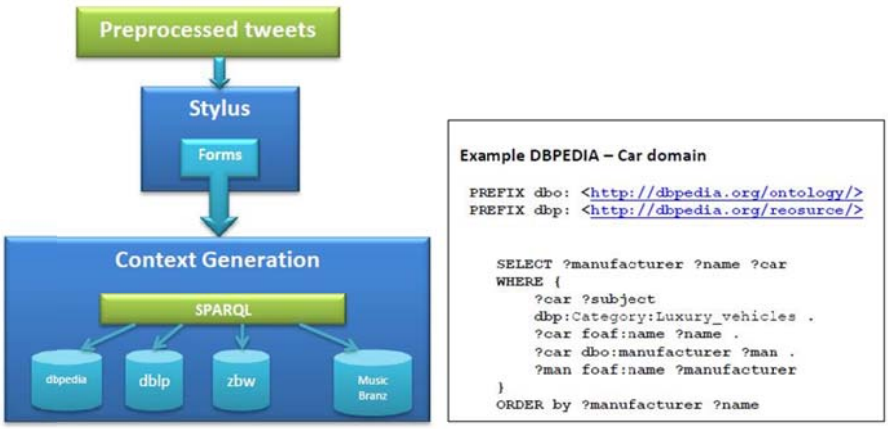

Fig. 2. Process of context generation

Thus, the context generation process consists in two sub-processes, (figure 2):

- **Extraction of the forms of an entity.** Using the API of Stylus[2] (Daedalus) the entity forms have been extracted to try to avoid misspelled or ambiguous names.
  IBM Software IBM Software Group IBM System

[2] http://www.daedalus.es/productos/stilus/stilus-sem/

- **Extraction of the concepts/entities and relationships from the previous forms**. This task is performed by consulting different datasets through their corresponding SPARQL endpoint, using SPARQL query language and following the ontologies of each dataset, that depends on the type of entity. An example of a SPARQL query type is described in figure 2.

2.4 Filtering Algorithm

In this section, the final version of the complete algorithm is provided.

```
//Preprocesed the tweets
PREPROCESS TWEET(tweet list) return preocessedTweet list
BEGIN
FOR EACH tweet IN tweet_list
{
processedTweet =RemoveURL(tweet);
processedTweet =RemoveMentions(processedTweet );
processedTweet =TransforHashTag(processedTweet );
processedTweet list.add(processedTweet);
}
return processedTweet_list;
END
//
CONTEXT_GENERATION (entitites_form) return context_list
BEGIN
FOR EACH entity IN entity form
{
   queries=Select TypeQuery(entity);
   results=ExecuteSparql(queries);
   context_list.put(entity, results);
}
return context list;
END;

// Main Program
MAIN (t_files, e_files)
BEGIN
   tweet list=readTweets(t files);
   entity_list=readEntities(e_files);
   tweet_terms=PREPROCESS_TWEETS(tweet_list);
   //Obtain all the forms for each entity with Stylus API
   ent forms=OBTAIN FORMS(entity  list);
   //Obtain Context for each entity
   context list=CONTEXT GENERATION (ent forms);
   //Compare tweets versus entities
   relatedness_list=COMPARE (contex_lists, tweets_terms);
   // Print on a file the filtering task results
   writeFilteringOutput(relatedness list);
END;
```

3 Experiments and Results

The corpus of RepLab 2013 uses Twitter data in English and Spanish. The corpus consists of a collection of tweets (at least 2,200 for each entity) potentially associated to 61 entities from four domains: automotive, banking, universities and music/artists. As result measures, reliability and sensitivity are used [8]. For a better and deep understanding, the outcomes for this typical binary classification problem (true positives, false positives, true negatives and false negatives) are showed:

Table 1.

Predicated Class	*Actual Class*		
		Related	Unrelated
	Related	**TP = 22926**	**FP = 1358**
	Unrelated	**FN = 47135**	**TN = 18495**

- **Sensitivity** (also called the true positive rate) measures the proportion of actual positives which are correctly classified.

$$\text{Sensitivity} = \text{TP} / \text{Actual P} = 0{,}944078 \quad (1)$$

- **Reliability** (also called precision) measures the fraction of retrieved instances which belong to positive class.

$$\text{Reliability} = \text{TP} / \text{Predicted P} = 0{,}327229 \quad (2)$$

Table 2. Filtering Results

Run	Reliability	Sensitivity	F(R,S)
BEST_APPROACH	**0,7288**	**0,4507**	**0,4885**
UNEDTECNALIA_filtering_1	0,2760	0.2862	0.1799

4 Related work

The disambiguation task has become essential when trying to mining the web in the search of opinions. Brands or individuals such as Apple or Bush usually lead to confusions due to their ambiguity and need to be disambiguated as a related or unrelated reference. In many approaches Wikipedia has been used to tackle this challenge by co-reference resolution methods (measuring context similarity through vectors or another kind of metric) [1, 2]. Research has been used to focus on the appearance of a pair of named entities in both texts to come to a conclusion about their interrelation. The problem with Twitter is the shortness of its messages which makes more difficult the comparison (overall considering the usual lack of two co-appearing entities).

Some works are carried out by mapping name entities to Wikipedia articles and overlapping the context surrounding the entity in the text (the string which is wanted to be disambiguated) [3]. The systems return the entity which best matches the context.

This approach, instead, tries to take advantage of Linked Open Data Cloud. A huge open data base where to ask and recover data in a straight way. This avoids scanning unstructured pages and obtaining wrong or disconnected information. Other paper that has been used as corpus, the data extracted from Linked Open Data sources is presented by Hogan et al. [4].

In Natural Language Processing, the Recognizing named entities (NER) is a extensively research field. Typically, the approaches used Wikipedia for explicit disambiguation [5], but there are also some examples of how semantics can be used for this task [6, 7]. Both works are based on defining a similarity measure based on the semantic relatedness.

Hoffart et al. [6]] is the most closed approach to our work, because the knowledge base used on their works are Linked Data sources, like DBpedia and YAGO and in our research we also use DBpedia (among others). The main difference is that on our approach we generate a context on which we place the entities in study. Afterwords we check if the text has any relationship with the generated context, instead of using a measure of semantic relatedness.

5 Conclusions

The results reveal a high value for sensitivity which comes along with a low value for false negatives. This indicates that the system does not usually get wrong classifications and if it concludes that one example is related to one entity it is almost sure that it is correct.

The reliability, however, is quite poor due to the fact that the context is very enclosed and thus there are a lot of not found examples (false negative rate). This leads us to think that the context should be widely enriched and this could enlarge the well classified group. So to filter with better precision, the context should contain not only semantic information from Linked Data sources, but also domain concepts such as verbs, idioms or any kind of expressions prone to be good indicators.

```
Entity: Led Zeppelinq
Tweet: Listening to Led Zeppelin
Context: [music, band, concert, instrument,… listen,…]

Result: TRUE
```

These clues could be extracted and treated by means of PLN and IR (Information Retrieval) algorithms. The first ones to preprocess the words (including stemming and disambiguation treatment) and the former in order to find a similarity-based structure for the data so the filtering can be carried out by measuring the distance between the query (actually the relationship related/unrelated) and the tweet according to the clues. Commonly, a data mining process would need to learn from training examples or on the other hand to use some statistical method as the *tf-idf* scheme or LSI (Latent Semantic Indexing) able to categorize and clustering the concepts most associated to a certain subject

For context generation, we will also analyze more refined techniques in the same research line:

- Improving the semantic context, using a larger number of Linked Datasets, and refining the questions to be sent. In order to improve the questions we

plan to delve deeper into the ontologies and thereby expand the scope of the context.
- Using other disambiguation techniques that can be combined with our approach, as the information extraction from web pages, cited in the text, the study of hash tags and mentions or using other non-semantic corpuses.

The combination of all these techniques would allow creating a huge semantic-pragmatic context with the valuable distinct feature of not being static, but an increasing and open context fed by Linked Data.

Acknowledgments. This work has been partially supported by the Regional Government of Madrid under Research Network MA2VIRMR (S2009/TIC-1542), and by HOLOPEDIA (TIN 2010-21128-C02). Special thanks to Daedalus for the free licencing to the utilization of Stilus Core. Hereby the authors would like to thank Fundación Centros Tecnológicos Iñaki Goenaga (País Vasco) for the awarded doctoral grant to the first author.

References

1. Ravin, Y. and Z. Kazi. 1999. Is Hillary Rodham Clinton the President? In ACL Workshop on Coreference and its Applications.
2. Wacholder, N., Y. Ravin, and M. Choi. 1997. Disambiguation of proper names in text. In Proceedings of ANLP, 202-208.
3. Bunescu and Pasca. 2006. Razvan C. Bunescu and Marius Pasca. Using encyclopedic knowledge for named entity disambiguation. In EACL. The Association for Computer Linguistics, 2006.
4. HOGAN, Aidan, et al. Scalable and distributed methods for entity matching, consolidation and disambiguation over linked data corpora. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2012, vol. 10, p. 76-110.
5. HAN, Xianpei; ZHAO, Jun. Named entity disambiguation by leveraging wikipedia semantic knowledge. En *Proceedings of the 18th ACM conference on Information and knowledge management*. ACM, 2009. p. 215-224.
6. HOFFART, Johannes, et al. Robust disambiguation of named entities in text. En *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011. p. 782-792.
7. HAN, Xianpei; ZHAO, Jun. Structural semantic relatedness: a knowledge-based method to named entity disambiguation. En *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010. p. 50-59.
8. Amigo, Enrique and Gonzalo, Julio and Verdejo, Felisa. A General Evaluation Measure for Document Organization Tasks. Proceedings SIGIR 2013.