Semantic Web and Information Extraction

SWAIE 2012

Workshop in conjunction with the
18th International Conference on
Knowledge Engineering and Knowledge Management
Galway City, Ireland, 9 October 2012

Edited by:
Diana Maynard
Marieke van Erp
Brian Davis

Preface

There is a vast wealth of information available in textual format that the Semantic Web cannot yet tap into: 80% of data on the Web and on internal corporate intranets is unstructured, hence analysing and structuring the data - social analytics and next generation analytics - is a large and growing endeavour. The goal of the 1st workshop on Semantic Web and Information Extraction was to bring researchers from the fields of Information Extraction and the Semantic Web together to foster inter-domain collaboration. To make sense of the large amounts of textual data now available, we need help from both the Information Extraction and Semantic Web communities. The Information Extraction community specialises in mining the nuggets of information from text: such techniques could, however, be enhanced by annotated data or domain-specific resources. The Semantic Web community has already taken great strides in making these resources available through the Linked Open Data cloud, which are now ready for uptake by the Information Extraction community. The workshop invited contributions around three particular topics: 1) Semantic Web-driven Information Extraction, 2) Information Extraction for the Semantic Web, and 3) applications and architectures on the intersection of Semantic Web and Information Extraction.

SWAIE 2012 had a number of high-quality submissions. From these, the 6 best papers were chosen for the two paper sessions of the programme: 4 long paper presentations and 2 short ones. Additionally, we held a lightning talks session where attendees could present brief 3-minute talks about late-breaking work or demos around the workshop themes, and a panel session involving some general discussion about these themes. To initiate the workshop, a keynote talk was provided by D.J. McCloskey, NLP Architect in IBM's new Watson Solutions division. The keynote presented the post-Watson role of Information Extraction and its intersection with the Semantic Web.

We would like to thank the many people who helped make SWAIE 2012 such a success: the Programme Committee, the paper contributors, the invited speaker and panellists, and all the participants present at the workshop who engaged in lively debate.

Diana Maynard, University of Sheffield
Marieke van Erp, VU University Amsterdam
Brian Davis, DERI Galway

Programme Committee

- Georgeta Bordea, DERI Galway, Ireland
- Matje van de Camp, Tilburg University, The Netherlands
- Christian Chiarcos, Information Sciences Institute, USA
- Hamish Cunningham, University of Sheffield, UK
- Thierry DeClerck, DFKI, Germany
- Robert Engels, Western Norwegian Research Institute, Norway
- Phil Gooch, City University London, UK
- Seth Grimes, Alta Plana Corporation
- Siegfried Handschuh, DERI, Ireland
- Dirk Hovy, Information Sciences Institute, USA
- Laurette Pretorius, University of South Africa, South Africa
- Birgit Proell, Johannes Kepler University of Linz, Austria
- Giusseppe Rizzo, EURECOM, France
- Piek Vossen, VU University, The Netherlands
- Marie Wallace, IBM Dublin, Ireland
- René Witte, Concordia University, Montreal, Canada

Contents

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.

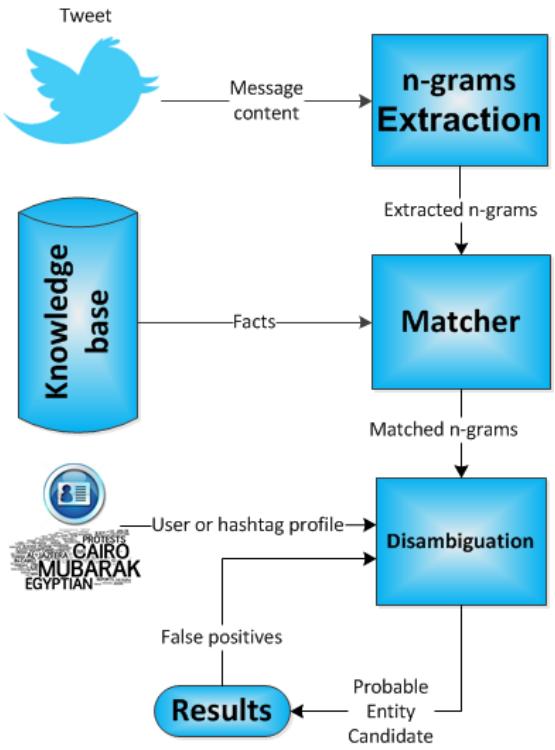


Fig. 1: Proposed Approach for Twitter NEE & NED.

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.

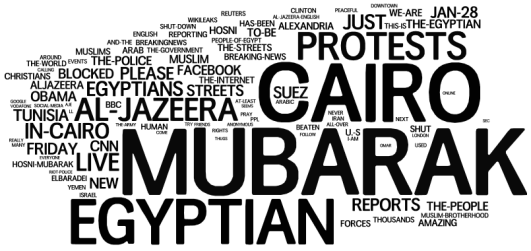
4 Conclusion and Future Work

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.

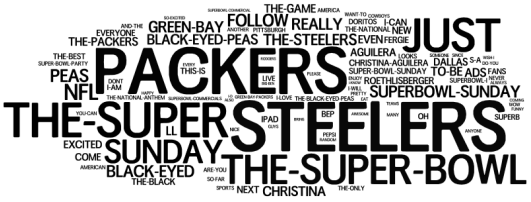
In our future work, we aim to enhance our results by considering a wider context than a single message for NED, applying relation weights for reducing the impact of non-distinguishing highly-connected entities, and to study the portability of our approach across multiple languages.

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.

LODIE: Linked Open Data for Web-scale Information Extraction

Fabio Ciravegna, Anna Lisa Gentile, and Ziqi Zhang

Department of Computer Science, The University of Sheffield, UK
{f.ciravegna,a.l.gentile,z.zhang}@dcs.shef.ac.uk

Abstract. This work analyzes research gaps and challenges for Web-scale Information Extraction and foresees the usage of Linked Open Data as a groundbreaking solution for the field. The paper presents a novel methodology for Web scale Information Extraction which will be the core of the LODIE project (Linked Open Data Information Extraction). LODIE aims to develop Information Extraction techniques able to (i) scale at web level and (ii) adapt to user information need. We argument that for the first time in the history of IE this will be possible given the availability of Linked Data, a very large-scale information resource, providing annotated data on a growing number of domains.

1 Introduction

Information Extraction (IE) is the technique for transforming unstructured textual data into structured representation that can be understood by machines. It is an essential technique to automatic knowledge capture, and has been an active research topic for decades. With the exponential growth of the Web, an unprecedented amount of data is made available online. Extracting information from this gigantic data source - or to be called Web-scale IE in the rest of this paper - in an effective and efficient way has been considered a major research challenge. Over the years, many different approaches [1–5] have been proposed. Nevertheless, the current state of the art has mainly addressed tasks for which resources for training are available (e.g. the TAP ontology in [1]) or use generic patterns to extract generic facts (e.g. [2], OpenCalais.com). The limited availability of resources for training has so far prevented the study of the generalised use of large-scale resources to port to specific user information needs.

This paper introduces the Linked Open Data Information Extraction (LODIE) project, a 3-year project that focuses on the study, implementation and evaluation of IE models and algorithms able to perform efficient user-centric Web-scale learning by exploiting Linked Open Data (LOD). Linked Data is [. . .] a recommended best practice for exposing, sharing, and connecting data [. . .] using URIs and RDF (www.linkeddata.org). LOD is ideally suited for supporting Web-scale IE adaptation because it is: (i) very large scale, (ii) constantly growing, (iii) covering multiple domains and (iv) being used to annotate a growing number of pages that can be exploited for training. The latter is particular interesting for IE: with the creation of schema.org, major players like Google,

Yahoo! and Bing are currently inviting Web content creators to include LOD-based microformats in their webpages in order to make the data and information contained understandable to search engines and Web robots. Similarly, RDFa is being adopted to produce annotations (http://www.w3.org/TR/xhtml-rdfa-primer). Researchers are starting to consider the use of LOD for Web-scale IE, however the approaches adopted so far are limited in scope to recognising tables [6], and extraction of specific answers from large corpora [7], but a generalised approach to the use of LOD for training large scale IE is still missing. LODIE will fill this gap by studying how an imprecise, redundant and large-scale resources like LOD can be used to support Web-scale user-driven IE in an effective and efficient way. The idea behind the project is to adapt IE methods to detailed user information needs in a completely automated way, with the objective of creating very large domain-dependent and task-dependent knowledge bases.

The remainder of this paper is organised as follows: Section 2 briefly introduces state of the art on Web-scale IE and the use of LOD in IE; Section 3 discusses the research gaps and challenges that LODIE aims to address; Section 4 introduces the LODIE methodology and architecture; Section 5 describes evaluation plan; and Section 6 concludes this paper.

2 Related Work

Adapting IE methods to Web-scale implies dealing with two major challenges: large scale and lack of training data. Traditional IE approaches apply learning algorithms that require large amount of training data, typically created by humans. However, creating such learning resources at Web-scale is infeasible in practice; meanwhile, learning from massive training datasets can be redundant and quickly become intractable [8].

Typical Web-scale IE methods adopt a light-weight iterative learning approach, in which the amount of training data is reduced to a handful of manually created examples called "seed data". These are searched in a large corpus to create an "annotated" dataset, whereby extraction patterns are generalised using some learning algorithms. Next, the learnt extraction patterns are re-applied to the corpus to extract new instances of the target relations or classes. Mostly these methods adopt a bootstrapping pattern where the newly learnt instances are selected to seed the next round of learning. This is often accompanied by some measures for assessing the quality of the newly learnt instances in order to control noisy data. Two well-known earlier systems in this area are Snowball [9] and KnowItAll [1, 2]. Snowball iteratively learns new instances of a given type of relation from a large document collection, while KnowItAll learns new entities of predefined classes from the Web. Both have inspired a number of more recent studies, including StatSnowball [10], ExtremeExtraction [4], NELL [3] and PROSPERA [5]. Some interesting directions undertaken by these systems include exploiting background knowledge in existing knowledge bases or ontologies to infer and validate new knowledge instances, and learning from negative seed data. While these systems learn to extract predefined types of information based on (limited) training data, the TextRunner [2] system proposes the "Open Information Extraction", a new paradigm that

exploits generic patterns to extract generic facts from the Web for unlimited domains without predefined interests.

The emergence of LOD has opened an opportunity to reshape Web-scale IE technologies. The underlying multi-billion triple store[1] and increasing availability of LOD-based annotated webpages (e.g., RDFa) can be invaluable resources to seed learning. Researchers are starting to consider the use of LOD for Web-scale information extraction. However, so far research in this direction has just taken off and the use of Linked Data is limited. Mulwad et al. [6] proposed a method to interpret tables based on linked data and extract new instances of relations and entities from tables. The TREC2011 evaluation on the Related Entity Finding task [7] has proposed to use LOD to support answering generic queries in large corpora. While these are relevant to our research, full user-driven complex IE task based on LOD is still to come.

LODIE will address these gaps by focussing on the following research questions: (i) How to let users define Web-IE tasks tailored to their own needs? (ii) How to automatically obtain training data (and filter noise) from the LOD? (iii) How to combine multi-strategy learning (e.g., from both structured and unstructured contents) to avoid drifting away from the learning task? (iv) How to integrate IE results with LOD?

3 LODIE - User-centric Web-scale IE

In LODIE we propose to develop an approach to Web-scale IE that enables fully automated adaptation to specific user needs. Users will be supported in defining their tasks using the LOD and IE methods and algorithms will be able to adapt to the new tasks using LOD as background knowledge. LOD will provide ontologies to formalise the user information need, and will enable seeding learning by providing instances (triples) and webpages formally annotated via RDFa or Microformats. Such background knowledge will be used to seed semi-supervised Web-scale learning. Output from the IE task will be both a set of instances to publish on the LOD, as well as a set of annotations which will provide provenance for the generated instances.

The use of an uncontrolled and constantly evolving, community provided set of independent Web resource for large-scale training is totally untapped in the current state of the art. Research has shown that the relation between the quantity of training data and learning accuracy follows a non-linear curve with diminishing returns [11]. On LOD the majority of resources are created automatically by converting legacy databases with limited or no human validation, thus errors are present [12]. Similarly, community-provided resources and annotations can contain errors, imprecision [13], spam, or even deviations from standards [14]. Also, large resources can be redundant, i.e. contain a large number of instances that contribute little to the learning task, while introducing considerable overhead. For example, the uptake of RDFa and microformat annotations is mainly happening at sites that generate webpages automatically, e.g. using a database back-end (e.g. eCommerce sites). Very regular annotations present very limited variability, and hence (i) high overhead for the learners (which will have to cope with thousands

[1] http://www4.wiwiss.fu-berlin.de/lodcloud

of examples providing little contribution) and (ii) the high risk of overfitting the model. For this reason, LODIE will put particular focus on measures and strategies to filter background knowledge to obtain noiseless and efficient learning.

The main contributions by LODIE will be:

- A method to formalise user requirements for Web-scale IE via LOD. We introduce methods based on ontology patterns [15] both to allow users to formalise their information needs and to identify relevant LOD resources to power adaptation to the task.
- Methods to evaluate the quality of LOD data and to select the optimal subset to seed learning. We introduce two measures: (i) Variability: to select seeds able to provide the learner with the optimal variety, so to avoid overfitting and overhead; this is expected to increase recall in extraction; (ii) Consistency to identify noisy data; this is expected to increase the precision of the IE process while reducing overhead during learning.
- The development of efficient, iterative, semi-supervised, multi-strategy Web-scale learning methods robust to noise and able to avoid drifting away when re-seeding. The methods will be able to exploit local and global regularities (e.g. page and site-wide regularities) as well redundancy in information [16].
- An evaluation process where we will test the above mentioned models in a number of tasks in order to compare them with the state of the art, both by defining tasks to be reused by other researchers and by participating in international competitions on large scale IE. The level of complexity of using large scale uncontrolled resources to seed Web-scale IE has never been previously addressed.

4 LODIE - Architecture and Methodology

We define Web-scale IE as a tuple: $< T, O, C, I, A >$ where: T is the formalisation of the user information needs (i.e. an IE Task); O is the set of ontologies on the LOD. C is a large corpus (typically the Web) which can be annotated already in part (C_L) with RDFa/Microformats; we refer to the unannotated part as C_U. I represents a collection of instances (knowledge base) defined according to O; I_L is a subset of I containing instances already present on the LOD; I_U is the subset of I containing all the instances generated by the IE process when the task is executed on C. A is a set of annotations and consists of two parts: A_L are found in C_L, and A_U are created by the IE process; A_U can be the final set or the intermediate sets created to re-seed learning.

The proposed method for IE applies a semi-supervised approach, based on identification of weak seeds for learning (high recall) followed by a filtering process that ensures only the candidates that are reasonably certain (precision) are used to (re)seed learning. We will work on an extension of the model we presented in [17] where (i) an initial set of seed instances I_L is identified, (ii) candidate annotation A_L and A_U are identified from C_L and C_U; (iii) a learning model is learned using (C_C, I_L, A_L, A_U), (iv) information is extracted by applying the model to C_C to generate I_U, A_U and (v) the new annotations are used to reseed another round of learning.

The overview of the LODIE approach is shown in Figure 1. The workflow includes the formalisation of the task T using LOD, the identification and optimisation of I and A to seed learning, the study of semi-supervised multi-strategy IE learning models, and the publication of A_U and I_U to LOD.

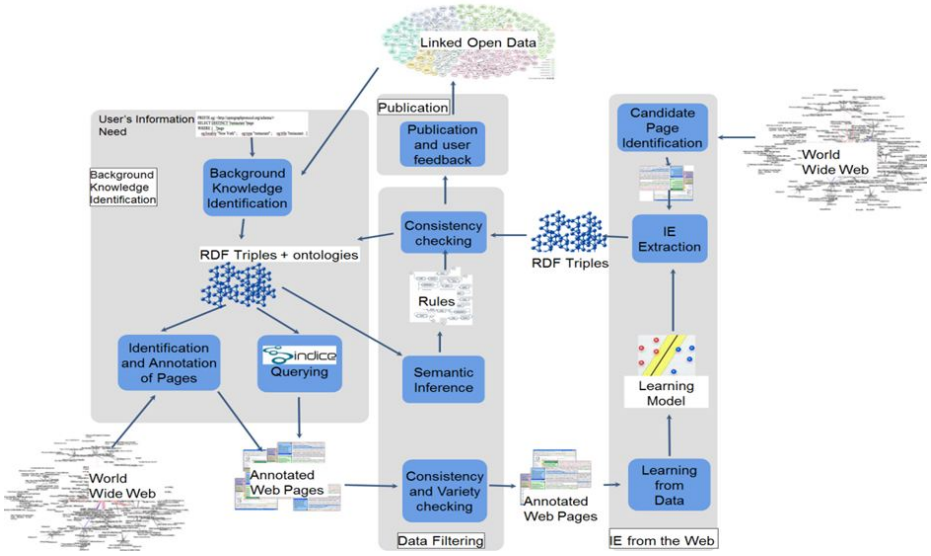


Fig. 1. Architecture diagram.

4.1 User needs formalisation

The first requirement for adapting Web-scale IE to specific user needs is to support users in formalising their information needs in a machine understandable format. Formally we define the user needs as a function: $T = f(O) \rightarrow O_L$ identifying a view on the LOD ontologies describing the information extraction task. T will be materialised in the form of an OWL ontology. We propose two ways to define T. The baseline strategy will be bottom up and will include: (i) identifying manually relevant ontologies and concepts on the LOD by using search engines like Swoogle (swoogle.umbc.edu) and Watson (watson.kmi.open.ac.uk) and (ii) manually defining a view on them using a standard tool like the Neon Toolkit (neon-toolkit.org). The second, more challenging strategy will be top-down and will be based on: (i) the formalisation of user needs using Content Ontology Design Patterns (Content ODP) [18] and (ii) the matching of the resulting ontology with existing LOD ontologies using Reeningeering ODPs.

An ODP is a reusable successful solution to a recurrent modelling problem. Content ODPs are patterns that describe a conceptualization addressing specific requirements,

e.g. in terms of competency questions or reasoning tasks. Content ODPs can be manifested as OWL ontologies, i.e. small OWL building blocks. User requirements will be described in terms of specializations of general Content ODPs, i.e. by specialising the general Content ODPs using user terminology. This will generate an ideal ontology describing the task. This ideal task will then have to be mapped to the reality of the LOD. First, relevant ontologies will be found using search engines like Swoogle and Watson. Then, transformational ODPs are used to turn the generated ontology into a view on the LOD by matching its concepts and relations with those actually found on the LOD. We will use Reeningeering Patterns, e.g. transformation recipes [19], currently proposed for semantically grounded triplifications. Reengineering Patterns will here be applied to map the user-generated semantically grounded ontology to an existing LOD ontology. This represent a kind of reverse approach than generally used in literature where Reengineering ODPs are used to map a database schema to a semantically grounded ontology. We will develop a user interface to define the IE task which will guide the user in an effective and efficient way. We will identify relevant Content and Reengineering ODP for the IE task and if necessary develop new ones. Application of patterns will be done using the Neon XD Tools plugin [20] and the Semion tool (stlab.istc.cnr.it/stlab/Semion).

4.2 Learning seed identification and filtering

A set of triples I_L relevant to the users need are identified as side effect of the definition of T: they can be retrieved from existing LOD knowledge bases associated with the types in T. We will use search engines like Sindice to identify RDFa and Microformat A_L which are associated to the types in T (if available). To these, we will add further candidates A_U identified by searching the Web for linguistic realisation of the triples I_L. In order to reduce noise due to ambiguity of the linguistic realisations [17], we will look for co-occurrence of known related instances in the same textual contexts (e.g. sentences [3]), and structural elements (e.g. tables and lists [21]) and apply focussing techniques (e.g. relevant ranking [7]).

These annotations together with A_L are used by the multi-strategy learning process to create new candidate annotations and instances. We have adopted similar approaches in [17] and [21]. Before feeding the identified annotations to the learning process, they will be filtered to ensure high quality in training data. This is achieved by using two measures, the measures of consistency and variability.

Filtering seeds - consistency measure: We will define a measure of consistency to filter A to prevent the learning algorithm to be misled by spurious data. Our hypothesis is that good data should present consistency with respect to the learning task. We will cast filtering as a problem of detecting noise in training data [22, 23]. These methods usually apply an ensemble of supervised classifiers to the training data and identify the noisy examples as those demonstrating high level of inconsistency in terms of the labels produced by classifiers. However in doing so, the classifiers used to detect noisy examples are constructed initially from a training set already containing noise, which may introduce bias in the process [23].

We propose to evaluate the consistency of the annotations by applying unsupervised clustering techniques and study the cluster membership of individual examples. We will map each $a \in A$ to a feature vector representing its form (superficial and semantic), other entities it appears with (together with their types and their reciprocal relations, from simple co-occurrence to specific relations), and other words it appears with in the sentence, etc. Then we will exploit unsupervised clustering techniques to split the data into clusters. Each generated cluster will be associated to a specific class by assigning the type of the largest majority of instances; ambiguous clusters will be discarded. The clustering procedure will be repeated iteratively under different settings; each a will then be assigned a value of consistency, which will be a function of how a consistently scores in the clusters associated to its actual type during the iterative process.

To minimise computation we will apply sampling methods to A to create a representative sample of manageable size. We will introduce methods to mathematically formulate the assessment of consistency based on an annotations cluster membership behaviour. The consistency score will be used to confirm the validity of a both before seeding (or re-seeding) learning and before the generation of the final set of annotations.

Optimising seeds - variability measure: Large numbers of examples in a very large resource like the LOD can contribute little to learning while substantially increase the computational overhead. The issue is increased when semi-supervised algorithms use self-learning (i.e. re-seeding) as strategy (e.g., [1, 3]) because, due to the nature of information redundancy on the Web, it is highly likely that a large portion of the reseeding data is also redundant. Very little has been done to prevent this issue in large scale IE. We hypothesize that good data should also present variability with respect to the learning task. Thus we introduce the notion of variability in the IE task and propose a novel measure to address this.

Given the annotations $t_A \subseteq A$ associated with one specific type t, we use the variability measure to evaluate t_A and select a subset $t_{A'} \subseteq t_A \subseteq A$ to (re-)seed learning for the type t. The measure of variability is adapted from the consistency measure. We will start by mapping each $a \in A$ to a feature vector representing its form in the same way as in the consistency measure. Then we will apply an agglomerative clustering algorithm [24] so that t_A will be clustered into a number of groups and the centroid of each cluster can be computed. The variability of the data collection t_A should reflect the number of clusters derived naturally and the distribution of members in each cluster. Intuitively, a higher number of clusters imply a higher number of groups of different examples, which ensures more extraction patterns to be learnt to ensure coverage; while even distribution of cluster members ensures the patterns can be generalised for each group. We hypothesize the variability of each $a \in A$ be dependent on the general variability of the collection, and on their distance to the centroid of each cluster because intuitively, the closer an element is to the centroid, the more representative it is for the cluster. We will introduce methods to mathematically formulate the variability based on these factors. At the end of the process we will have selected a subset $t_{A'} \subseteq t_A \subseteq A$.

4.3 Multi-strategy Learning

The seed data identified and filtered in the previous steps are submitted to a multi-strategy learning method, which is able to work in different ways according to the type of webpages the information is located in: (i) a model M_S able to extract from regular structures such as tables and lists; (ii) a model M_W wrapping very regular web sites generated by backing databases and (iii) a model M_T for information in natural language based on lexical-syntactic extraction patterns.

As for extracting from regular structures, following early work by [25, 26], we will adopt a strategy able to exploit the dependencies among entities expressed in one page/site to learn to extract from that page. As an example, for tables we will build a feature model based on text in each cell, as well as text from column label and text in the possibly related entities (text from cells in the same row). Moreover, when two or more annotations $a_W \in A$ of compatible type W appear in the same substructure (e.g. same column) in a document in C_U, and other candidates $a_X \in A$ of compatible type X bearing a relation r with a_W can be found in other parts of the same structure (e.g. other columns in the same table), we will hypothesize that all the other elements in those sub-structures will be of the type W and X and carry the same relation r. As a result, we will output a number of potential annotations $a \in A_U$ for each candidate in the table. To decide the best type assignment for each column we will initially experiment with strategies such as least common ancestors and majority [26] and compare and combine them with methods exploiting an enhanced feature model, that will take into account the semantics and restrictions in O [21].

For learning to wrap a site given one of its pages containing a potential reference to $a_{jW} \in A$, we will check if other pages from the same site are on the to do list for T and contain other $a_{iW} \in A$ of compatible type W in equivalent position (i.e. same XPath). If they do, we will suppose the site is to be wrapped and will extract from all the site pages that follow the identical XPath structure. As a result, we will output a number of potential annotations $a \in A_U$. Exploiting structural patterns of web pages for Information Extraction is often referred as wrapper induction [27]. We will experiment with both bottom-up and top-down strategies to wrapping [28] and combine structural and content elements from the pages.

Finally for all other cases, we will learn shallow patterns. As opposed to approaches based on complex machine learning algorithms (e.g. random walks in [24]), we will focus on lexical-syntactic shallow pattern generalization algorithms. The patterns will be generalised from the textual context of each $a \in A$ and will be based on features such as words (lexical), part of speech (syntactic) and expected semantics such as related entity classes. We will base the algorithm on our previous research in [21]. The innovation will be focused on modifying the algorithm to account for negative examples, and enriching the pattern representation with semantics mined from external knowledge resources, such as fine-grained entity labels as in [29]. The patterns are then applied to other webpages to create new candidate annotations.

At the end of this process, we concatenate the candidate annotations extracted by each learning strategy and create a collection of candidates $a \in A_U$. These will refer to instances already known (I_L) as well as new instances (I_U). The goal of the next steps will be to create new triples in I. Also to form an iterative semi-supervised learning

pattern, the annotations will be selected to reseed the new round of learning. In order to prevent the learner from drifting away if noisy data is permitted to creep in re-seeding [30, 28, 3], we will filter the candidate A_U by their consistency, and optimize them by their variability (see previous section). The learning continues in a never-ending fashion to consistently update the knowledge base [3]. Results will be visualized as soon as they are produced and users will be able to subscribe to their queries and be notified and updated when new facts of interest are mined.

4.4 Publication of new triples in the LOD

We will develop methods to enable the learned knowledge to be published and integrated into the LOD by exposing a SPARQL endpoint. In order to do so, the candidates A_U identified by IE will be assigned to a URI, i.e. a unique identifier. We call this step disambiguation [17]. The core of our disambiguation process will be exploiting features to obtain the optimal representation of each candidate set. We will use both co-occurrence based features (gathered from the context of occurrence of a given noun phrase) and relational features (obtained by exploring relational properties in the ontologies) [31]. As scalability is a major requirement both in terms of T and C, we will explore methods with minimum requirements in computational terms such as simple feature overlapping based methods [32] and string distance metrics [13]. We will compare their effectiveness with that of more computationally intensive machine learning methods such as HMM [17], random walks [24] etc.

Finally, in order to correct mistakes and improve the quality of both data and learning, a user-friendly interface will be created to enable users to provide feedback by correcting mistakes in both the knowledge base and the annotations. Strategies such as those employed by WIQA (Information Quality Assessment Framework) [33] using different information filtering policies will be employed. Corrections made by users are collected as feedback to the learning process. These are fed into new learning cycles and it is anticipated that with minimum and voluntary user feedback the learning process can improve over time. It has been shown that learning systems benefit largely from very little human supervision [11].

5 Evaluation

In order to test the effectiveness of the IE algorithms we will test both the suitability of the approach to formalise the user needs and the suitability of the approach to IE.

As for the definition of user needs, we will test the approach by giving a task described in natural language to experts in IE with a reasonable understanding of LOD and asking them to define an equivalent IE task. Evaluation will consider (i) feasibility and efficiency: can a user develop a task in a reasonable time with limited overhead using ontology patterns? We will test this by timing the task and comparing with the use of the baseline method; (ii) effectiveness: is the result really representative of the user needs? Are the resulting task ontology and the associated triples/annotations suitable to

seeding IE? This will be assessed in two ways: on the one hand users will have to judge the resulting T, A_L and I_L as relevant to their needs; on the other hand we will evaluate the returned triples in terms of usefulness to learning using the quality measures described below.

As for the effectiveness of the IE process, we will measure empirically different aspects of the learning strategy, from the different algorithms, different versions of the measures, etc. To separate this aspect of evaluation from the user evaluation, we will define a new task based on population of sections of the schema.org ontology and we will test the effectiveness of the IE system in different configurations. Typically standard measures of evaluation for IE are based on precision; since the unbounded domain and sheer amount of data on the Web makes it largely impossible to study other measures such as recall. However, besides precision, we will attempt also a partial evaluation of recall by providing the system with just a fraction of the available A_L and checking recall with respect to the A_L not provided for training. Moreover, we plan to participate in comparative large scale IE evaluations such as the TAC Knowledge Base Population [34] or the TREC Entity Extraction task [7] to compare our technology with the state of the art.

6 Conclusion

LODIE is a project addressing complex challenges that we believe are novel and of high interest to the scientific community. It is timely because (i) for the first time in the history of IE a very large-scale information resource is available, covering a growing number of domains and (ii) of the very recent interest in the use of Linked Data for Web extraction. Potential for exploitation is very high. A number of challenges are ahead and require the use of technologies from fields such as knowledge representation and reasoning, IE and machine learning. We intend to use knowledge patterns to formalise user requirements for Web-scale IE. We will develop efficient iterative semi-supervised multi-strategy Web-scale learning methods robust to noise and able to avoid drifting away when re-seeding. Particular focus will be put on efficient and robust methods: we will develop and test methods to evaluate the quality of LOD data for training and to select the optimal subset to seed learning.

Acknowledgments

The LODIE project (Linked Open Data Information Extraction) is funded by the Engineering and Physical Sciences Research Council, Grant Reference: EP/J019488/1.

References

1. Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Web-Scale Information Extraction in KnowItAll (Preliminary Re-

sults). In: WWW2004 Proceedings of the 13th international conference on World Wide Web. (2004) 100–110
2. Banko, M., Cafarella, M., Soderland, S., Broadhead, M., Etzioni, O.: Open information extraction for the web. In: IJCAI'07 Proceedings of the 20th international joint conference on Artifical intelligence. (2007) 2670–2676
3. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Jr, E.R.H., Mitchell, T.M.: Toward an Architecture for Never-Ending Language Learning. In: Proceedings of the Conference on Artificial Intelligence (AAAI). (2010) 1306–1313
4. Freedman, M., Ramshaw, L.: Extreme extraction: machine reading in a week. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, ACL (2011) 1437–1446
5. Nakashole, N., Theobald, M.: Scalable knowledge harvesting with high precision and high recall. WSDM '11 Proceedings of the fourth ACM international conference on Web search and data mining (1955) (2011) 227–236
6. Mulwad, V., Finin, T., Syed, Z., Joshi, A.: Using linked data to interpret tables. In: First International Workshop on Consuming Linked Data (COLD2010). (2010)
7. Balog, K., Serdyukov, P.: Overview of the TREC 2010 Entity Track. In: Proceedings of the Nineteenth Text REtrieval Conference (TREC 2010), NIST (2011)
8. Joachims, T.: Making large scale SVM learning practical. In B. Scholkprof, C.J.C.Borges, Smola, A., eds.: Advances in kernel methods. MIT Press, Cambridge, MA, USA (1999) 169–184
9. Agichtein, E., Gravano, L., Pavel, J.: Snowball: a prototype system for extracting relations from large text collections . ACM SIGMOD . . . (2001) 612
10. Zhu, J.: StatSnowball : a Statistical Approach to Extracting Entity. In: WWW '09 Proceedings of the 18th international conference on World wide web. (2009) 101–110
11. Thompson, C.A., Hall, V., Mooney, R.J.: Active Learning for Natural Language Parsing and Information Extraction LEARNING SYSTEMS. In: Proceedings of the Sixteenth International Conference on Machine Learning. ICML 99. (1999) 406–414
12. Auer, S., Dietzold, S., Lehmann, J., Hellmann, S., Aumueller, D.: Triplify Light-Weight Linked Data Publication from Relational Databases. WWW '09 Proceedings of the 18th international conference on World wide web (2009) 621–630
13. Lopez, V., Nikolov, A., Sabou, M., Uren, V.: Scaling up question-answering to linked data. In: Proceedings of the 17th international conference on Knowledge engineering and management by the masses. EKAW10. (2010) 193–210
14. Halpin, H., Hayes, P., McCusker, J.: When owl: sameas isn't the same: An analysis of identity in linked data. In: Proceedings of 9th International Semantic Web Conference ISWC 2010. (2010) 305–320
15. Gangemi, A., Presutti, V.: Towards a pattern science for the Semantic Web. Semantic Web **0** (2010) 1–7
16. Blanco, L., Bronzi, M., Crescenzi, V., Merialdo, P., Papotti, P.: Redundancy-driven web data extraction and integration. Procceedings of the 13th International Workshop on the Web and Databases - WebDB '10 (2010)
17. Rowe, M., Ciravegna, F.: Disambiguating identity web references using Web 2.0 data and semantics. Web Semantics: Science, Services and Agents on the World Wide Web **8**(2-3) (July 2010) 125–142
18. Presutti, V., Gangemi, A.: Content ontology design patterns as practical building blocks for web ontologies. In: Conceptual Modeling-ER 2008. (2008) 128–141
19. Nuzzolese, A., Gangemi, A.: Fine-tuning triplification with Semion. Proceedings of the 1st Workshop on Knowledge Injection and Extraction from LD at EKAW 2010 (2010)

20. Blomqvist, E., Presutti, V., Daga, E., Gangemi, A.: Experimenting with eXtreme Design. In: Proceedings of the 17th international conference on Knowledge engineering and management by the masses. EKAW10. (2010) 120–134
21. Ciravegna, F., Chapman, S., Dingli, A.: Learning to harvest information for the semantic web. The Semantic Web: (1) (2004)
22. Jiang, Y., Zhou, Z.h.: Editing Training Data for kNN Classifiers with. In: ISNN 2004, International Symposium on Neural Networks. (2004) 356–361
23. Valizadegan, H., Tan, P.: Kernel Based Detection of Mislabeled Training Examples. In: Proceedings of the Seventh SIAM International Conference on Data Mining. (2007) 309–319
24. Iria, J., Xia, L., Zhang, Z.: Wit: Web people search disambiguation using random walks. SemEval '07 Proceedings of the 4th International Workshop on Semantic Evaluations (June) (2007) 480–483
25. Milne, D., Witten, I.: Learning to link with wikipedia. In: CIKM '08 Proceedings of the 17th ACM conference on Information and knowledge management. (2008) 509–518
26. Limaye, G., Sarawagi, S., Chakrabarti, S.: Annotating and Searching Web Tables Using Entities , Types and Relationships. Proceedings of the VLDB Endowment **3**(1-2) (2010) 1338–1347
27. Kushmerick, N.: Wrapper Induction for information Extraction. In: IJCAI97. (1997) 729–735
28. Dalvi, N., Kumar, R., Soliman, M.: Automatic wrappers for large scale web extraction. Proceedings of the VLDB Endowment **4**(4) (2011) 219–230
29. Kazama, J., Torisawa, K.: Inducing gazetteers for named entity recognition by largescale clustering of dependency relations. In: Proceedings of ACL-08: HLT. (2008) 407–415
30. Curran, J., Murphy, T., Scholz, B.: Minimising semantic drift with mutual exclusion bootstrapping. In: Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics. (2007) 172–180
31. Krishnamurthy, J., Mitchell, T.: Which noun phrases denote which concepts? In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Volume 15213. (2011) 570–580
32. Banerjee, S., Pedersen, T.: An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. In: CICLing '02: Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing, London, UK, Springer-Verlag (2002) 136–145
33. Bizer, C., Cyganiak, R.: Quality-driven information filtering using the WIQA policy framework. Web Semantics: Science, Services and Agents on the World Wide Web **7**(1) (January 2009) 1–10
34. Ji, H., Grishman, R.: Knowledge base population: Successful approaches and challenges. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - HLT 11. (2011) 1148–1158

Ontologies as a Source for the Automatic Generation of Grammars for Information Extraction Systems

Thierry Declerck and Paul Buitelaar

DFKI GmbH, Language Technology Lab,
Stuhlsatzenhausweg 3, D-66123 Saarbruecken, Germany
declerck@dfki.de
Unit for Natural Language Processing, DERI,
National University of Ireland, Galway
paul.buitelaar@deri.org

Abstract. Grammars for Natural Language Processing (NLP) applications are generally built either by linguists – on the basis of their language competence, or by automated tools applied to existing large corpora of language data — using either supervised or unsupervised methods (or a combination of both). Domain knowledge usually played just a little role in this process. The increasing availability of extended knowledge representation systems, like taxonomies and ontologies, is giving the opportunity to consider new approaches to the (automated) generation of processing grammars, especially in the field of domain-oriented Information Extraction (IE). The reason for this being that most of the taxonomies and ontologies are equipped with natural language expressions included in ontology elements like labels, comments or definitions. These de facto established relations between (domain) knowledge and natural language expressions can be exploited for the automatic generation of domain specific NLP and IE grammars. We describe in this paper steps leading to this automation.

Keywords: Ontology-based Information Extraction, Grammar Generation, Business Reporting Standards

1 Introduction

In the last 10-15 years we have experienced a huge increase of available knowledge sources of various types, like taxonomies or ontologies, which are also available online. The more recent establishment of the linked (open) data framework[1] has further boosted this development, making available a tremendous amount of (interlinked) knowledge objects in the web. Some formal and logic-based knowledge representation languages like RDF(s) and OWL[2], which are used for encoding

[1] See http://linkeddata.org/

[2] RDF(s) stands for "Resource Description Framework (schema)" and OWL for "Web Ontology Language". See http://www.w3.org/TR/rdf-schema/ and http://www.w3.org/TR/owl-features/ respectively.

these knowledge objects, have foreseen various possibilities to include natural language expressions.

These expressions can be part of RDF URI references, identifying ontological resources (e.g. natural language string used in `rdf:ID`), a fragment (e.g. natural language string in `rdf:about` statements) or marking empty property elements (kind of leaf nodes in a graph, using the `rdf:resource` statement). Examples of the use of such reference elements in ontologies are given below[3]:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:ex="http://example.org/stuff/1.0/"
         xml:base="http://example.org/here/">
  <rdf:Description rdf:ID="snack">
    <ex:prop rdf:resource="fruit/apple"/>
  </rdf:Description>
</rdf:RDF>
```

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Seq rdf:about="http://example.org/favourite-fruit">
    <rdf:_1 rdf:resource="http://example.org/banana"/>
    <rdf:_2 rdf:resource="http://example.org/apple"/>
    <rdf:_3 rdf:resource="http://example.org/pear"/>
  </rdf:Seq>
</rdf:RDF>
```

Natural language expressions can also be used in taxonomies and ontologies as the content of RDF annotation properties, like `rdfs:label` and `rdfs:comment`, as this is exemplified below[4]:

```
<rdf:Property ID="hasAccessTo">
  <rdfs:label xml:lang="en">has access to</rdfs:label>
  <rdfs:comment xml:lang="en">Relates an Access Rule
     to the resources to which the rule applies.
     The inverse relation is 'accessedBy'</rdfs:comment>
  <rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-schema#Resource"/>
  <rdfs:domain rdf:resource="#ResourceAccessRule"/>
  <rdfs:isDefinedBy resource="http://www.w3.org/2001/02/acls/ns#"/>
</rdf:Property>
```

In this paper, we focus on the content of annotation properties since they contain "real" natural language expressions. And additionally, labels and comments locally support multilingualism by means of language tags of RDF literals, i.e. `xml:lang`, whereas this is not the case for RDF URI references.

[3] Examples are taken from the (revised) RDF/XML Syntax Specification, a W3C Recommendation from 2004/02/10, see `http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/`.

[4] The slightly modified example is taken from `http://www.w3.org/2001/Talks/0710-ep-grid/slide21-0.html`.

2 Labels as a Source for Ontology-Based Information Extraction

The use of labels in knowledge representation systems is nowadays widely supported, as can be seen for example in the XBRL[5] taxonomies representing different legislations for business reporting, in the FMA ontology[6] for human anatomy or in the RadLex ontology[7] encoding radiology terms. Figure 1 shows an example from the XBRL taxonomy of the Belgian National Bank, where the reader can see the use of labels, also including the xml:lang feature, in a multilingual setting, relating natural language expressions to the concept `FixedAssets`.

```
<loc xlink:label="FixedAssets_loc" xlink:type="locator"
 xlink:href="pfs-2011-04-01.xsd#pfs_FixedAssets" />
<labelArc xlink:from="FixedAssets_loc" xlink:to="FixedAssets_lab"
 xlink:type="arc"
 xlink:arcrole="http://www.xbrl.org/2003/arcrole/concept-label" />

<label xlink:label="FixedAssets_lab" xlink:type="resource"
 xlink:role="http://www.xbrl.org/2003/role/label" xml:lang="fr">
 Actifs immobilisés</label>
<label xlink:label="FixedAssets_lab" xlink:type="resource"
 xlink:role="http://www.xbrl.org/2003/role/label" xml:lang="nl">
 Vaste activa</label>
<label xlink:label="FixedAssets_lab" xlink:type="resource"
 xlink:role="http://www.xbrl.org/2003/role/label" xml:lang="de">
 Anlagevermögen</label>
<label xlink:label="FixedAssets_lab" xlink:type="resource"
 xlink:role="http://www.xbrl.org/2003/role/label" xml:lang="en">
 Fixed assets</label>
```

Figure 1: Example of multilingual labels in the taxonomy for business reporting of the Belgian National Bank.

This combination of domain-specific knowledge and terms suggests that the main task of Ontology-Based Information Extraction (OBIE) is consisting now in the mapping or unification of language data available in ontologies/taxonomies and language data as available in running text. While the labels in the example displayed in Figure 1 are very simple and finding matches in running text would be straightforward, let us just give another example for showing the complexity of the task of matching labels in taxonomies and relevant expressions in free text, considering another concept of the Belgian National Bank taxonomy (displaying only the English label):

[5] XBRL (eXtended Business Reporting Language) "is a language for the electronic communication of business and financial data", see `http://www.xbrl.org/GettingStarted`.

[6] See `http://sig.biostr.washington.edu/projects/fm/`.

[7] See `http://bioportal.bioontology.org/ontologies/42801`.

```
<label xlink:label="GuaranteesGivenByThirdPartiesBehalfEnterprise_lab"
 xlink:type="resource" xlink:role="http://www.xbrl.org/2003/role/label"
 xml:lang="en">Guarantees given by third parties on behalf of the
 enterprise</label>
```

Figure 2: Example of a more complex label used in the taxonomy of the Belgian National Bank.

The IE task consists in mapping the text "Guarantees given by third parties on behalf of the enterprise" to a running text, in which the choice and the order of words is not corresponding to the term used in the label. And also one needs to find the naming of an enterprise in the text and so to provide for an instance to the abstract notion of "enterprise" in the term. In this, OBIE extends the typical task of IE in the fact that the extraction of information from documents is not done any more on the base of language knowledge and pre-specified domain templates, but that it is now relying on a central semantic dimension – an ontology or a taxonomy, which can have dynamic properties. OBIE needs thus both access to representations of linguistic units of interest (i.e. names, terms, phrases, relation-expressing verbs or nouns, prepositions, etc.), as sued in labels, and to semantically structured representations of object classes of interest (i.e. domain concepts and properties).

3 The *lemon* Model for the Representation of Language Data in Ontologies

In order to support the mapping of language data in ontology labels and the language data in running text, there is the need to render the language data comparable and thus to represent it in a standardized way. For this we propose as a first step the lexicalization of the labels of ontologies, associating to them typical linguistic information, like lemma, morphology, syntax. But there is also a need to state the particular relation of such lexicalized labels to the ontology elements they are associated with. We opted for solving this representation issue for the *lemon* (LExicon Model for ONtologies) model, which is designed to represent lexical information about linguistic units (e.g. words, phrases, terms) relative to an ontology.[8] *lemon* uses ontological elements, which are termed 'references', identified by URIs, to represent the semantics of a linguistic unit, i.e. by specifying a class or a property that can be linked to. The lexical information for the linguistic unit will then be expressed relative to this semantic 'reference', e.g. the fact that this ontology element will be typically realized by a noun "asset" with plural form "assets", modified by the adjective "fixed" (considering here our example in Figure 1). Similar for the more complex term "Financial liabilities at amortised cost", for which the *lemon* representation will encode the fact that the prepositional head has to be "at". *lemon* is based on a clear separation between semantics and morpho-syntax, while at the same time enabling the

[8] See [6].

precise specification of their correspondences. A simplified lemon entry ("asset") is shown in Figure 3.

```
:asset_noun lemon:canonicalForm [ lemon:writtenRep "asset" ] ;
    lemon:altForm [ lemon:writtenRep "assets"  ;
                lexinfo:number lexinfo:plural ] ;
    lexinfo:partOfSpeech lexinfo:noun .
    lemon:sense [ lemon:ref <http://www.ebr.org/xbrl#Asset> ] ;
```

Figure 3: A simplified representation of the lemon entry for *asset*. The semantics of the word (its "lexical sense"') is encoded by a reference to an XBRL concept.

Given this rich structure and content of *lemon* ontology-lexicons there is much potential for exploiting them in the generation of ontology-specific extraction grammars that capture semantic as well as linguistic aspects of the lables used in the domain ontology.

4 Exploiting Structural Information from the Ontology

Additionally to this representation of the lexical and linguistic content in RDF encoded *lemon* entries, we need to derive from the ontology element the *lemon* entry is referring to some structural information, so for example the class-hierarchy in which the element is introduced, and information about associated properties. So for example the xEBR[9] ontology we are working with is stating that " hasFixedAssetsTotal" is a property with domain class "FixedAssetPresentation", and has as range a monetary value, which we specified as an xsd type, which states that a monetary value consists of both an amount and a currency. The owl representation of this property is shown below in Figure 4:

```
<owl:DatatypeProperty
 rdf:about="http://www.dfki.de/lt/xebr.owl#hasFixedAssetsTotal">
 <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#monetary"/>
 <rdfs:domain
  rdf:resource="http://www.dfki.de/lt/xebr.owl#FixedAssetsPresentation"/>
 <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
 <rdfs:label xml:lang="en">fixed assets [total]</rdfs:label>
</owl:DatatypeProperty>
```

Figure 4: The owl-xml representation of the property "hasFixedAssetsTotal".

[9] xEBR is a core taxonomy for various national XBRL taxonomies. See `http://www.monnet-project.eu/Monnet/Monnet/English/Navigation/XBRLEuropeanBusinessRegisterxEBR`. In the project Monnet (www.monnet-project.eu/) , we have "upgraded" eXBR onto an ontology, and interfacing this ontology with other financial ontologies, derived from examples from stock exchange pages (Xetra, Euronext). eXBR, as well as Xetra and Euronext, are multilingual. See [5] for more details. Here we are dealing with English and German.

Since the class "FixedAssetPresentation" is in a part-of relation to the class "AssetsPresentation", being itself a subclass of "KeyBusinessFigureReport", which is defined for a specific duration, this temporal value is inherently valid for the property "hasFixedAssetsTotal"'. In this case the IE system knows that if in a document an expression is found that could be attached to the ontology element "FixelAssetTotal", the system also has to find as minimal condition, in relevant segments of the processed documents, expressions for both a monetary and a temporal value, in order to allow the system to populate the ontology with adequate values for the property.

5 Generating Grammars for the OBIE system

Combining both the *lemon* representation of the textual content of an ontology element label and the structural information about this element, a simple extraction grammar rule is being generated as follows[10]:

```
[fixed asset]
     N-plu & monetary-value & temporal-value
     => xebr:hasFixedAssetsTotal
```

The simplified rule specifies that if in a certain textual context (sentence, clause, or in a financial table) a natural language expression can be matched to the term "fixed assets"', while in the textual context also both a monetary and a temporal expressions can be found, then the whole construction can be marked as being of type "xebr:hasFixedAssetsTotal". The monetary and temporal values found in the text are used then for populated the ontology elements related to the report of the company under consideration. The compound noun"fixed asset" is represented here in its ground from (lemma), with the additional information that it should be used in the plural form. This rule can be applied to the following sentence from a financial report as follows, where we mark-up by in-line annotation the elements that lead to the detection of the XBRL concept in text and to the subsequent ontology population:

```
[In [2011](period) the Kuehne + Nagel Group invested
[CHF 207 million](monetary) in fixed assets] (xebr:hasFixedAssetsTotal)
```

A more complete version of this example involves the generation of an extraction rule that captures the relation between the actual monetary value mentioned in this sentence (CHF 207 million) and the property xebr:hasFixedAssetsTotal. This corresponds to the (simplified) following rule[11]:

[10] Those rules, partly derived from the *lemon* representation, are encoded in the NooJ formalism (`www.nooj4nlp.net/`). We present here a more readable pseudo-code. See also [3] for an overview on how ontologies can interact with NooJ linguistic data.

[11] The rule gives the impression that a fixed order for the semantic elements is reuird, but NooJ foresees a mechanism (the feature "ONCE") that allows to find the elements at most one time in a certain fragment, in no specific order.

```
[fixed asset]N-plu  .* [X](monetary) .* [Y](date)
                => xebr:hasFixedAssets[range = X]
```

Obviously this version of the analysis is also far from complete, as we need to represent also the linguistic and semantic aspects of a monetary value, of intermediate words and phrases (invested (date) in), and names (Kuehne + Nagel Group), in order not allow only semantic annotation of the text with relevant ontology elements, but also to populate the ontology, with the information that the company Kuehne + Nagel Group has a report covering the year 2011, with the additional information about the specific amount for fixed assets.

6 Actual Experiments

As a test for the derived IE grammars, we started to process the so-called "Portrait" of companies displayed in the bi-lingual web presence of the German Börse. Our ontology background is the MFO integrated ontology, as descibed in [5], and our main goal is to extract eXBR (Business Reporting) information, but the tools are also extracting information about activitiy fields of companies, etc. In a first experiment, we focus on the 30 companies, as they are listed in DAX. Once the system has been adjusted to the specific type of input, we will test the system on the 130 companies listed in the related stock exchanges SDAX, MDAX and TecDAX. First preliminary results are encouraging, since our system can for example extract the following information from the Portait of the DAX-listed company "adidas".[12]:

```
<DATE><DAY>31</DAY> <MONTH>12</MONTH> <YEAR>2011</YEAR></DATE>
<XEBR+NetTurnover+13,3 Mrd>
.....
<DATE>Zum  <DAY>31.</DAY>
 <MONTH><NP HEAD>Dezember</HEAD></NP></MONTH> <YEAR>2011</YEAR></DATE>
<EMPLOYEE+46.000>
```

The two sentences from which this information is extracted are: "Im Jahr 2011 erzielte die adidas Gruppe einen Konzernumsatz in Höhe von 13,3 Mrd." (*In 2011, the adidas Group generated net sales in an amount of 13.3 billion*) and "Zum 31. Dezember 2011 beschäftigte die adidas Gruppe mehr als 46.000 Mitarbeiter." (*Effective December 31, 2011, the adidas Group employed more than 46,000 people.*)

The most difficult part is to detect in the running text term variants for the labels of the xEBR labels. We are working on automated term extraction for supporting this task (see [4]). Please also note that we can specify the exact date for the xEBR concept, although the text just specify "2011", since in the Business Reporting domain the end of the year is the typical date. In other cases, the full date is always specified in the text.

[12] See `http://www.boerse-frankfurt.de/en/aktien/adidas+ag+DE000A1EWWW0/unternehmensdaten` for the English version and `http://www.boerse-frankfurt.de/de/aktien/adidas+ag+DE000A1EWWW0/unternehmensdaten` for the German version.

7 Conclusion

In this short paper we have described the process of generating automatically from ontologies grammar rules for information extraction systems. A pre-requisite for this is to perform a lexicalization of the labels of ontology elements and to represent this information using for example the *lemon* model. A *lemon* lexicon provides a very rich description of the language data used in labels and includes a reference to a semantic element (a class or a property in an ontology), a term structure (decomposition of the label into sub-terms), and linguistic knowledge (lexical features of word forms used in the term). Combined with structural information about the "location"' of the ontology element in the whole ontology, an efficient definition and even automatic derivation of extraction grammars is being proposed and currently tested.

Acknowledgments. This work is supported in part by the European Union under Grant No. 248458 for the Monnet project.

References

1. Aggarwal, N., Wunner, T., Arcan, M., Buitelaar, P., O'Riain, S.: A Similarity Measure based on Semantic, Terminological and Linguistic Information. In: Proceedings of the 6th InternationalWorkshop on Ontology Matching collocated with the 10th International Semantic Web Conference (ISWC-2011) Bonn, Germany (2011)
2. Declerck, T., Lendvai, P., Wunner, T.: Linguistic and Semantic Features of Textual Labels in Knowledge Representation Systems. In: Proceeedings of ISA6, ISO/ACL-SIGSEM Workshop, January 11-12, Oxford (2011)
3. Declerck, T. Lendvai, P., Váradi, T., Koleva, K.: Integration of Ontological Semantic Resources in NooJ. In: Proceedings of the 2011 International NooJ Conference. Cambridge Scholars Publishing (2012)
4. Federmann, C., Gromann, D., Declerck, T., Hunsicker, S., Krieger, H.U., Budin, G. Multilingual Terminology Acquisition for Ontology-based Information Extraction. In: Proceedings of the 10th Terminology and Knowledge Engineering Conference, Pages 166-175, Madrid, Spain (2012)
5. Krieger, H.K., Declerck, T., Nedunchezhian, A.K.: MFO - The Federated Financial Ontology for the MONNET Project. In: Proceedings of the 4th International Conference on Knowledge Engineering and Ontology Development, Barcelona, Spain (2012)
6. McCrae, J., Spohr, D., Cimiano, P.: Linking Lexical Resources and Ontologies on the Semantic Web with lemon. In: In Proceedings of the 2011 Extended Semantic Web Conference (2011)
7. Wunner, T., Buitelaar, P., O'Riain, S.: Semantic, Terminological and Linguistic Interpretation of XBRL. In: Proceedings of EKAW, October 11-15, Lisbon (2010)

Identifying Consumers' Arguments in Text

Jodi Schneider[1] and Adam Wyner[2]

[1] Digital Enterprise Research Institute, National University of Ireland
jodi.schneider@deri.org
[2] Department of Computer Science, University of Liverpool, Liverpool, UK
adam@wyner.info

Abstract. Product reviews are a corpus of textual data on consumer opinions. While reviews can be sorted by rating, there is limited support to search in the corpus for statements about particular topics, e.g. properties of a product. Moreover, where opinions are justified or criticised, statements in the corpus indicate arguments and counterarguments. Explicitly structuring these statements into arguments could help better understand customers' disposition towards a product. We present a semi-automated, rule-based information extraction tool to support the identification of statements and arguments in a corpus, using: argumentation schemes; user, domain, and sentiment terminology; and discourse indicators.

Keywords: argumentation schemes, information extraction, product reviews

1 Introduction

Product reviews such as found on Amazon or eBay represent a source of data on consumer opinions about products. Current online tools allow reviews to be sorted by star rating and comment threads. Yet, there is no support to search through the data for statements about particular topics, e.g. properties of the product. Such statements are distributed throughout the corpus, making it difficult to gain a coherent view. Moreover, reviewers justify their opinions as well as support or criticise the opinions of others; that is, reviewers provide arguments and counterarguments. Extracting data about particular topics and structuring it into arguments would be informative: it could help producers better understand consumers' disposition to their products; and, it could help consumers make sense of the product options, as reported in the reviews, and so then decide what to buy. We present an information extraction tool to support the extraction of arguments from reviews, using: user, domain, and sentiment terminology; discourse indicators; and argumentation schemes.

To set the context, consider the information in the reviews from the point of view of a consumer or manufacturer. We use product reviews from the Amazon consumer web site about buying a camera as a use case. From the consumer side, suppose a photo enthusiast wants to buy a new camera that gives quality indoor pictures. The enthusiast consults a shopping website and reads the reviews of camera models. The information in product reviews about this topic is dispersed through a number of reviews, using different terminology, and expressing opinions on different sides of the situation. So, currently, the enthusiast must read through the reviews, keeping in mind the relevant

statements, organising and relating them; this is a difficult task. Instead, the enthusiast would like all statements bearing on the camera's indoor picture quality to be reported and sorted according to whether the statement supports the claim that the camera gives quality indoor pictures or supports the claim that it does not. Moreover, it is not sufficient for the enthusiast to be provided with one 'layer' of the argument, since those statements which support or criticise the claim may themselves be subject to support or criticism. From the manufacturer's side, there is a related problem since she wishes to sell a product to a consumer. Looking at the reviews, the manufacturer must also extract information about specific topics from the corpus and structure the information into a web of claims and counterclaims. With this information, the manufacturer could have feedback about the features that the consumer does or doesn't like, the problems that the consumer experiences, as well as the proposed solutions.

There are a variety of complex issues to address. For instance, to overcome the linearity of the corpus and terminological variation, we want a tool that searches and extracts information from across the corpus using semantic annotations, allowing us to find statements about the same semantic topic; searches for strings do not suffice since the same semantic notion might be expressed with different strings. Sentiment identifiers, which signal approval or disapproval, are relevant. Discourse markers indicate relationships between statements, e.g. premise or claim. In addition, users argue from a *point of view:* different user classes, e.g. amateurs and professionals, argue differently about the same object.

While a fully automated system to reliably extract and structure all such information is yet in the future, we propose a semi-automated, rule-based text analytic support tool. We first manually analyse the corpus, identifying the sorts of semantic information to be annotated. We develop reasoning patterns, *argumentation schemes*, and identify slots in these schemes to be filled. The schemes represent different aspects of how users reason about a decision to buy a product. We structure the schemes into a decision tree, hypothesising a main scheme which is used to argue for buying the product. This main scheme is supported by subsidiary schemes that argue for premises of the main scheme. In turn, the subsidiary schemes are grounded in textual information related to the user and the representation of the product. In effect, we reverse engineer an argumentative expert system which takes as input material from the corpus. Thus, the schemes give us *targets* for information extraction in the corpus, namely, those components that can be used to instantiate the argumentation schemes. The information extraction tool supports the identification of relevant information to instantiate the argumentation schemes. As a result of the analysis and instantiation, we gain a rich view on the arguments for or against a particular decision. The novelty is that the tool systematically draws the analyst's attention to relevant terminological elements in the text that can be used to ground defeasible argumentation schemes.

The outline of the paper is as follows. In Section 2, we discuss our use case and materials. Several components of the analysis are presented in Section 3: user, domain, and sentiment terminology; and discourse indicators. The argumentation schemes that we propose to use are given in Section 4. The tool is outlined in Section 5, followed by sample results in Section 6. Related work is discussed in Section 7, and we conclude in Section 8 with some general observations and future work.

2 Use Case and Materials

As a use case, we take reviews about buying the (arbitrarily chosen) Canon PowerShot SX220 HS Digital Camera from the Amazon UK e-commerce website[3], where a very typical question is: *Which camera should I buy?* There are 99 reviews in our corpus, distributed as shown in Table 1.

Table 1. Review distribution by star rating

5-star	54
4-star	27
3-star	9
2-star	8
1-star	1

In these product reviews, many topics are discussed. By careful reading and analysis, we find comments about cameras such as their features and functions. Further, accessories, such as memory cards, batteries, and cases are also discussed – both with regard to their necessity or utility, and their suitability. The brand reputation and warranty are discussed. Users also give conditions of use – recommendations for who the camera would or would not suit, and warnings and advice about how to get the best results. These incorporate the purpose or context in which the camera is or could be used (e.g. "traveling") or values that the camera fulfils (e.g."portable"). Users also give clues to their own experience and values, by talking about how they evaluated the camera, their experience with photography, or personal characteristics (e.g. "ditzy blonde").

Point of view is key to making sense of the overall discussion. For subjective aspects, the impact of a statement may depend on the extent to which consumers share values and viewpoints. Such qualitative aspects of the reviews are not captured by quantitative measures of the discussion since the most popular comment may not advance the analysis with respect to that user or may only sway individuals who are susceptible to popular opinion. Given this, we focus on representing justifications and disagreements with respect to classes of users.

In the course of our manual examination of the corpus, we identified five "components" of an analysis: several consumer argumentation schemes; a set of discourse indicators, and user, domain, and sentiment terminology. The user and domain terminology are used to instantiate the schemes, while the discourse indicators and sentiment terminology structure the interrelationships between the statements within a scheme (e.g. premises, claim, and exception) and between schemes (e.g. disagreement). We begin by discussing the last four components, then turn to argumentation schemes in Section 4.

3 Components of Analysis

The objective of information extraction in our context is to extract statements about a topic (e.g. a camera takes good pictures indoors) and structure them into arguments for (e.g. justifications for this claim) or against it (counterclaims and their justifications).

[3] Accessed 2012-07-22 `https://www.amazon.co.uk/product-reviews/B004M8S152/re`

In the following, we briefly outline the components of our analysis, which are implemented in the tool discussed in Section 5. In our approach, we identify a *terminological pool* that helps us investigate the source text material for relevant passages; thus, we presume that we can search throughout the corpus to instantiate an argumentation scheme using the designated terminology.

In our approach to analysis of the source material, we have presumed that in the context of product reviews, contributors are trying to be as helpful, informative, and straightforward as possible, so the interpretation of language is at *face value*. In other contexts, problematic, interpretive aspects of subjectivity may arise, e.g. irony or sarcasm, which require significant auxiliary, extra-textual knowledge to accurately understand. For our purposes, we do not see irony or sarcasm as a significant problem as we can rely on the normative reading of the text that is shared amongst all readers.

Camera Domain We have terminology from the camera domain that specifies the objects and properties that are relevant to the users. Analysing the corpus, consumer report magazines (e.g. *Which?*), and a camera ontology[4], we identified some of the prominent terminology. These refer both to parts of the camera (e.g. lens, li-ion battery) as well as its properties (e.g. shutter speed). While users may dispute particular factual matters about a camera, these remain objective aspects about the camera under discussion.

User Domain Users discuss topics relative to their point of view, knowledge, and experience. This introduces a *subjective aspect* to the comments. For instance, whether an amateur finds that that a particular model of camera takes *very poor* pictures indoors may not agree with an expert who finds that the same model takes *good* pictures indoors; each is evaluating the quality of the resulting pictures relative to their own parameter of quality and experience with camera settings. To allow such user-relative judgements, we introduce user terminology bearing on a user's attributes (e.g. age), context of use (e.g. travel), desired camera features (e.g. weight), quality expectations (e.g. information density), and social values (e.g. prestige).

Discourse Indicators Discourse indicators express discourse relations within or between statements [1] and help to organise statements into larger scale textual units such as an argument. The analysis of discourse indicators and relations is complex: there many classes of indicators, multiple senses for instances of indicators depending on context, and implicit discourse relations. However, in this study, we keep to a closed class of explicit indicators that signal potentially relevant passages; it remains for the analyst to resolve ambiguities in context.

Sentiment Terminology We use *sentiment* terminology that signals lexical semantic contrast: *The flash worked poorly* is the semantic negation of *The flash worked flawlessly*, where *poorly* is a negative sentiment and *flawlessly* is a positive sentiment. An extensive list of terms is classified according to a sentiment scale from highly negative to highly positive [2]. Text analytic approaches to sentiment analysis are well-developed, but for our purposes we take this relatively simple model to integrate with other components.

In the following, we provide argumentation schemes that use the camera and user terminology. The discourse indicators and sentiment terminology are only used in the tool to identify relevant passages to instantiate the schemes.

[4] `http://www.co-ode.org/ontologies/photography/`

4 Argumentation schemes

Argumentation schemes represent prototypical patterns of defeasible reasoning [3]. They are like logical syllogisms in that they have premises, an implicational rule (e.g. *If....Then....*), and a conclusion that follows from the premises and rule. Moreover, they can be linked as in proof trees. Yet, unlike classical syllogisms, the conclusion only defeasibly follows since the rule or the conclusion may not hold. Argumentation schemes have been formalised [4] and can be used for abstract argumentation [5]. Example schemes include *practical reasoning*, *expert opinion*, and *analogy*. However, schemes are not widely used to support text analysis, are not tied to user terminology, and not usually tied to some particular domain. This paper makes progress in addressing these issues. In this section we develop a number of argument schemes found in customer reviews, based on manual review of the corpus. Our approach is to remain grounded in the source, and to choose example schemes based on their relevance to arguing for or against purchase of the product. In this way, the schemes give us *targets* for information extraction in the corpus: in particular, the targets are those textual passages that can be used to instantiate the argumentation schemes.

4.1 Argumentation Schemes - Abstract

We present the schemes propositions with variables such as aP_1; the list of premises is understood to hold conjunctively and the conclusion follows; the rule is left implicit.

User Classification With this scheme, we reason from various attributions to a user to the class of the user. This scheme is tied to the particular data under consideration, but could be generalised. We have a variety of users such as amateur or professional.

User Classification Argumentation Scheme (AS1)

1. *Premise*: Agent *x* has user's attributes $aP_1, aP_2, ...$
2. *Premise*: Agent *x* has user's context of use $aU_1, aU_2, ...$
3. *Premise*: Agent *x* has user's desirable camera features $aF_1, aF_2, ...$
4. *Premise*: Agent *x* has user's quality expectations $aQ_1, aQ_2, ...$
5. *Premise*: Agent *x* has user's values $aV_1, aV_2, ...$
6. *Premise*: User's desirable camera features $aF_1, aF_2, ...$ promote/demote user's values $aV_1, aV_2, ...$
   Conclusion: Agent *x* is in class *X*.

Camera Classification We have a scheme for classifying the camera. Note that we have distinguished a user's context of use from a camera's context of use (and similarly for other aspects); in a subsequent scheme (AS3), these are correlated.

Camera Classification Argumentation Scheme (AS2)

1. *Premise*: Camera *y* has camera's context of use $cU_1, cU_2, ...$
2. *Premise*: Camera *y* has camera's available features $cF_1, cF_2, ...$
3. *Premise*: Camera *y* has camera's quality expectations $cQ_1, cQ_2, ...$
   Conclusion: Camera *y* in class *Y*.

Combining Schemes for Camera Evaluation To reason about the camera and the course of action, we use some ontological reasoning, i.e. the class of the camera and of the user, plus argumentation. Given that a user is in class X with certain requirements and a camera is in class Y with certain features, and the features meet the requirements, then that camera is appropriate. The argument that conjoins the user and camera schemes works as a filter on the space of possible cameras that are relevant to the user. We realise this as follows.

Appropriateness Argumentation Scheme (AS3)

1. *Premise*: Agent *x* is in user class *X*.
2. *Premise*: Camera *y* is in camera class *Y*.
3. *Premise*: The camera's contexts of use satisfy the user's context of use.
4. *Premise*: The camera's available features satisfy the user's desireable features.
5. *Premise*: The camera's quality expectations satisfy the user's quality expectations.
   Conclusion: Cameras of class *Y* are appropriate for agents of class *X*.

Premises (1) and (2) of the appropriateness scheme (AS3) are the conclusions of the user (AS1) and consumer (AS2) classification schemes, respectively. The other premises (3)-(5) have to be determined by subsidiary arguments which nonetheless ground variables in the same way (in Logic Programming terms, the variables are *unified*). Each of these subsidiary schemes have a similar form, where premises correlate elements from AS1 and AS2 and conclude with one of the premises of (3)-(5). The redundancy ensures that the variables match across schemes. We leave such *intermediary schemes* as an exercise for the reader.

Practical Reasoning The objective of reasoning in this case is for the user to decide what camera to buy. The reasoning is based on the user and the camera. This information is then tied to the decision to buy the camera. Since reasoning about the camera relative to the user is addressed elsewhere in the reasoning process, our scheme (AS4) is a simplification of [6].

Consumer Relativised Argumentation Scheme (AS4)

1. *Premise*: Cameras of class *Y* are appropriate for agents of class *X*.
2. *Premise*: Camera y is of class *Y*.
3. *Premise*: Agent x is of class *X*.
   Conclusion: Agent x should buy camera y.

The important point is that if the class of the camera and user do not *align*, or if there are counterarguments to any of the premises or conclusions, then the conclusion from AS4 would not hold.

5 Components of the Tool

To build an analytic tool to explore and extract arguments, we operationalise the components needed to recognise in the text some of the relevant elements identified in Section 4. In this section, we briefly describe the relevant aspects of the General Architecture for Text Engineering (GATE) framework [7] and samples of how we operationalise the components. In Section 6, we show results of sample queries.

5.1 GATE

GATE is a framework for language engineering applications, which supports efficient and robust text processing [7]; it is an open source desktop application written in Java that provides a user interface for professional linguists and text engineers to bring together a wide variety of natural language processing tools and apply them to a set of documents. The tools are formed into a pipeline (a sequence of processes) such as a sentence splitter, tokeniser, part-of-speech tagger, morphological analyser, gazetteers, Java Annotation Patterns Engine (JAPE) rules, among other processing components. For our purposes, the important elements of the tool to emphasise are the gazetteers and JAPE rules: a gazetteer is a list of words that are associated with a central concept; JAPE rules are transductions that take annotations and regular expressions as input and produce annotations as output. Our methodology in using GATE is described elsewhere [8], and in this paper, we focus just on the key relevant elements - the gazetteers and JAPE rules.

Once a GATE pipeline has been applied to a corpus, we can either view the annotations of a text by using the ANNIC (ANNotations In Context) corpus indexing and querying tool [9] or view them *in situ* in a whole text. We illustrate both.

5.2 Gazetteers and JAPE Rules

In section 4, we presented terminology for discourse indicators and the camera domain. The terminology is input to text files such as *cameraFeatures.lst* for terms relating to the camera domain and *conclusion.lst* for terms that may indicate conclusions. The lists are used by a *gazetteer* that associates the terms with a *majorType* such as *cameraproperty* or *conclusion*. JAPE rules convert these to annotations that can be visualised and used in search. For example, suppose a text has a token term "lens" and GATE has a gazetteer list with "lens" on it; GATE finds the string on the list, then annotates the token with majorType as *cameraproperty*; we convert this into an annotation that can be visualised or searched for such as *CameraProperty*. A range of terms that may indicate conclusions are all annotated with *Conclusion*. We can also create annotations for complex concepts out of lower level annotations. In this way, the gazetteer provides a *cover concept* for related terms that can be queried or used by subsequent annotation processes.

In the implementation, we have gazetteer lists for camera domain terminology and for user domain terminology, one list each for conclusions, premises, and contrast, and a range of sentiment terminology lists. Samples of the lists (with number of items) are:

- conclusion.lst (26): be clear, consequent, consequently, deduce, deduction, ...
- cameraFeatures.lst (130): 14X Optical Zoom, action shots, AF tracking, ...
- posThree.lst (172): astound, best, excellent, splendid, ...
- userContextOfUse (32): adventure, ambient light indoors, astronomy photos, ...

In the next section, we show sample results.

6 Sample Results

To identify passages that can be used to instantiate the argumentation schemes, we use ANNIC searches to investigate the entire corpus. Figure 1 shows a result of a search for

negative sentiment, followed by up to 5 tokens, followed by a user context; the search returns six different strings that match the annotation pattern.

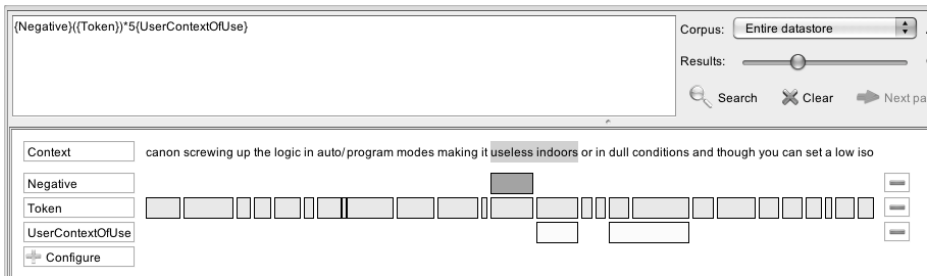


Fig. 1. Sample output from an ANNIC search.

We can also look at annotations *in situ* in a text. Figure 2 shows one review document, with a variety of annotation types, where different highlights indicate different annotation types (differentiated with colour in the original); from this review we extract instantiations for the user and camera schemes. This passage makes the argument that the camera is not appropriate since the user's context of use – baby pictures – does not match the camera context of use. In other words, we use the annotations to instantiate the two schemes below.

We have been researching loads to find a great new camera since we had a baby as our camera is old and about 1 in 3 pictures is very blurry now. We decided on this camera as it looked excellent and had good reviews, but we have sadly been disappointed with it and will not be keeping it.
On the plus side, the zoom feature is fantastic and the quality and detail you can get while zooming in over a distance is incredible. Pictures of our son's brightly coloured play toys are amazing. If you want to take photos mainly of things with bright colours, in good daylight, this is a great camera.
However, the camera does not seem to perform nearly as well for pictures of objects with less bright contrasting colours. Most of our pictures of paler objects, people, etc were not up to nearly the same standard, especially in low light. As we specifically wanted the camera for baby pictures it is therefore not suitable for us.
Also, as other reviewers have mentioned, the flash is very badly placed. It is on the top corner exactly where I generally tend to hold a camera so most times my fingers prevented it popping up properly and I had to manually raise it if I wanted to use it.

Fig. 2. An annotated review.

User Classification Argumentation Scheme - Baby Picture Reviewer

1. *Premise*: Agent *x* has user attributes: *little experience*.
2. *Premise*: Agent *x* has constraints: *single camera*.
3. *Premise*: Agent *x* has context of use *portrait*.
4. *Premise*: Agent *x* has user's desirable camera features *easy to hold, flash doesn't require user attention, zoom*.
5. *Premise*: Agent *x* has quality expectations *good pictures of pale objects, good pictures of objects that don't have contrast*.
6. *Premise*: Agent *x* has values *good reviews, photo quality, photo detail*.
7. *Conclusion*: Agent *x* is in class *Novice*.

Camera Classification Argumentation Scheme (AS2) - Baby Picture Reviewer

1. *Premise*: Camera *y* has camera's context of use *daylight*.
2. *Premise*: Camera *y* has camera's available features *zoom, flash*.
3. *Premise*: Camera *y* has camera's quality expectations *annoying flash, amazing for bright colours, poor when colours do not contrast (people, pale objects), good quality with zoom, good detail with zoom*.
   Conclusion: Camera *Canon PowerShot SX220* in class *daylight, contrast-oriented, zoom camera*.

One argument against the above camera classification is given by another reviewer: "This camera takes amazing low light photos...". Based on the full text of that review, we can instantiate the camera classification argumentation scheme differently, as follows:

Camera Classification Argumentation Scheme (AS2) - Great low light

1. *Premise*: Camera *y* has camera's context of use *video, photos*.
2. *Premise*: Camera *y* has camera's available features *HD video recording, screen, zoom, flash, colour settings*.
3. *Premise*: Camera *y* has camera's quality expectations *lens shadow, awkward flash location, vibrant colours*.
   Conclusion: Camera *Canon PowerShot SX220* in class *video, general photo camera*.

This shows some advantages of argumentation schemes. First of all, they can help an analyst make explicit the points of contention between reviews. The reviews disagree on the camera's quality expectations: this particular disagreement could not easily be discovered statistically from the text. Second, we can separate out different levels of subjective information to be found in the reviews. The user classification scheme separates the purely subjective information that cannot be attacked from the camera classification scheme, which can be fruitfully attacked. Further, by classifying cameras and users, an entire line of reasoning follows: we only need to instantiate those two schemes.

Some issues do arise, and will need to be considered in future work. First, we cannot always instantiate some premises. For example, users may not indicate user attributes or constraints in a review. In that situation, presumptive values could be used, or found elsewhere in the corpus. Second, there are other arguments and counterarguments that are made. For instance, some reviews suggest ways of dealing with the popup flash so that it's not annoying, making the camera more comfortable to use indoors. To handle more types of arguments and counterarguments, we will want to develop further argumentation schemes. Some negative implications depend on a deeper analysis of the camera domain, for instance: "You need to learn all functions in order to shoot really good photos." or "People look either washed out or with a flat looking red/orange complexion." Other arguments, such as arguments from expertise, are common, and should be analysed further to provide support for information extraction.

7 Related Work and Discussion

In this section, we outline related work, which includes opinion and review mining, user preferences, and ontological approaches, and use of argumentation. What makes our proposal novel and unique is the combination of *rule-based* text analytics, user models, and defeasible argumentation schemes, which together highly structure the representation of information from the source materials. In previous work we have introduced argumentation schemes for understanding evaluative statements in reviews as arguments from a point of view [18]. Our earlier, preliminary implementation, used a single argumentation scheme [6]; this paper extends that work by implementing user terminology and increasing the specification of camera terminology, and by using a cascade of argumentation schemes, where the conclusions of two schemes are the premises of the appropriateness scheme.

Opinion and Review Mining Existing work includes review mining – information extraction using sentiment terminology [10] – and feature extraction of pros and cons [19]. Matching customers to the most appropriate product based on the heterogeneity of customer reviews, rather than just statistical summaries, is an important problem; Zhang et al. develop sentiment divergence metrics, finding that the central tendency or polarity of reviews is insufficient [20]. Our goals, in matching customers to products by distinguishing views based on a customer profile, are similar; unlike that study, we focus on textual analysis, rather than statistical summarization of the text.

User Preferences Case-based reasoning has been used to incorporate critique-based feedback and preference-based feedback into recommendation systems. [21]. To predict ratings in Chinese-language restaurant reviews, Liu et al. model how frequently users comment on features ('concern') and how frequently they rate features lower than average ('requirement') in order to predict ratings [22]. Rather than inferring user preferences from multiple reviews written by a user, we extract user information from a single review; although some personal information (such as the user demographics) is consistent across items in different departments (such as books, movies, consumer electronics, clothing, etc.), the key information about the user is that related to the product, which depends on the category, and in some cases on the item being purchased. For instance, preferences about an item having a flash or a viewfinder are not universal amongst consumer electronics, but apply mainly to cameras.

Ontology-related approaches Yu et al. automatically construct a hierarchical organization for aspects from product reviews and domain knowledge [23]. This approach could be used to further enhance our extraction systems, and there are available tools in GATE to support this: OwlExporter is a GATE plugin for populating OWL ontologies via NLP pipelines [24]; KIM uses an ontology and knowledge base to add semantic annotations based on information extraction [25].

Argumentation Argumentation schemes have been used as a theoretical framework for reviews [26]. Another closely related problem is argumentation mining – using natural language processing to detect disagreement [11,12,13] or stance [14,15].

8 Conclusions and Future Work

We have presented an information extraction tool that supports the identification of relevant information to instantiate argumentation schemes, by annotating discourse indicators as well as user, domain, and sentiment terminology. Textual fragments are associated with annotation types, highlighting the role the text may play in instantiating an argumentation scheme. As we can identify positive and negative sentiment, we can find statements that contribute to arguments for or against other statements. The novelty of our proposal is the combination of *rule-based* text analytics, terminology for various particular components of the analysis, and defeasible argumentation schemes, which together highly structure the representation of information from the source materials. As a result of the analysis and instantiation, we can provide a rich, articulated analysis of the arguments for or against a particular decision.

In future work, we plan to further instantiate the schemes using the tool, noting where they work as intended and where they stand to be improved. Along with this, conceptual issues will be addressed, for instance to clarify distinctions between the camera's quality expectations and features as well as to support matches between a user's values and camera properties. We will develop additional schemes bearing on, for example, expertise, comparison, or particular features (e.g. warranties). An evaluation exercise will be carried out using a web-based annotation editor and evaluation tool, GATE Teamware, to measure the extent of interannotator agreement on the annotation types. Important *logical* developments would be an ontology for users and cameras that would support text extraction and import of scheme instances into an argumentation inference engine to test inferences.

Acknowledgements

The first author's work was supported by Science Foundation Ireland under both Grant No. SFI/09/CE/I1380 (Líon2) and a Short-term Travel Fellowship. The second author gratefully acknowledges support by the FP7-ICT-2009-4 Programme, IMPACT Project, Grant Agreement Number 247228. The views expressed are those of the authors.

References

1. Webber, B., Egg, M., Kordoni, V.: Discourse structure and language technology. Natural Language Engineering (December 2011) Online first.
2. Nielsen, F.Å.: A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. Making Sense of Microposts at ESWC 2011 (2011)
3. Walton, D., Reed, C., Macagno, F.: Argumentation Schemes. Cambridge Univ. Press (2008)
4. Wyner, A., Atkinson, K., Bench-Capon, T.: A functional perspective on argumentation schemes. In Proceedings of the 9th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2012). (2012) 203–222
5. Prakken, H.: An abstract framework for argumentation with structured arguments. Argument and Computation **1**(2) (2010) 93–124
6. Wyner, A., Schneider, J., Atkinson, K., Bench-Capon, T.: Semi-automated argumentative analysis of online product reviews. In: Proceedings of the 4th International Conference on Computational Models of Argument (COMMA 2012). (2012)

7. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A framework and graphical development environment for robust NLP tools and applications. In: Proceedings of the Association for Computational Linguistics (ACL'02). (2002) 168–175
8. Wyner, A., Peters, W.: On rule extraction from regulations. In Legal Knowledge and Information Systems - JURIX 2011, IOS Press (2011) 113–122
9. Aswani, N., Tablan, V., Bontcheva, K., Cunningham, H.: Indexing and querying linguistic metadata and document content. In: Proceedings of 5th International Conference on Recent Advances in Natural Language Processing, Borovets, Bulgaria (2005)
10. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval **2**(1-2) (January 2008) 1–135
11. Albert, C., Amgoud, L., de Saint-Cyr, F.D., Saint-Dizier, P., Costedoat, C.: Introducing argumention in opinion analysis: Language and reasoning challenges. In: Sentiment Analysis where AI meets Psychology (SAAIP at IJCNLP'11). (2011)
12. Saint-Dizier, P.: Processing natural language arguments with the <TextCoop>platform. Argument & Computation **3**(1) (2012) 49–82
13. Wyner, A., Mochales-Palau, R., Moens, M.F., Milward, D.: Approaches to text mining arguments from legal cases. In Semantic Processing of Legal Texts. Volume 6036 of Lecture Notes in Computer Science. Springer (2010) 60–79
14. Abbott, R., Walker, M., Anand, P., Tree, J.E.F., Bowmani, R., King, J.: How can you say such things?!?: Recognizing disagreement in informal political argument. In: Proceedings of the NAACL HLT 2011 (2011)
15. Walker, M.A., Anand, P., Abbott, R., Tree, J.E.F., Martell, C., King, J.: That's your evidence?: Classifying stance in online political debate. Decision Support Systems (2012)
16. Mann, W.C., Thompson, S.A.: Rhetorical structure theory: Toward a functional theory of text organization. Text-Interdisciplinary Journal for the Study of Discourse **8**(3) (1988) 243–281
17. Somasundaran, S., Wiebe, J.: Recognizing stances in ideological on-line debates. In: Proceedings of the NAACL HLT 2010, (2010) 116–124
18. Wyner, A., Schneider, J.: Arguing from a point of view. In: First International Conference on Agreement Technologies. AT '12 (2012)
19. Liu, B. "Opinion Mining and Sentiment Analysis." In: Web Data Mining. Springer, (2011) 459–526
20. Zhang, Z., Li, X., Chen, Y.: Deciphering word-of-mouth in social media: Text-based metrics of consumer reviews. ACM Trans. Manage. Inf. Syst. **3**(1) (April 2012) 5:1–5:23
21. Smyth, B.: Case-based recommendation. The Adaptive Web. Volume 4321 of Lecture Notes in Computer Science. Springer (2007) 342–376
22. Liu, H., He, J., Wang, T., Song, W., Du, X.: Combining user preferences and user opinions for accurate recommendation. Electronic Commerce Research and Applications (2012)
23. Yu, J., Zha, Z.J.J., Wang, M., Wang, K., Chua, T.S.S.: Domain-assisted product aspect hierarchy generation: towards hierarchical organization of unstructured consumer reviews. In: Proceedings of EMNLP '11 (2011) 140–150
24. Witte, R., Khamis, N., Rilling, J.: Flexible ontology population from text: The OWL exporter. In: LREC 2010. (2010) 3845–3850
25. Popov, B., Kiryakov, A., Kirilov, A., Manov, D., Ognyanoff, D., Goranov, M.: KIM semantic annotation platform. In The Semantic Web - ISWC 2003. Volume 2870 of Lecture Notes in Computer Science. Springer (2003) 834–849
26. Heras, S., Atkinson, K., Botti, V., Grasso, F., Julián, V., McBurney, P.: Applying argumentation to enhance dialogues in social networks. In: CMNA 2010. (2010)
27. Sporleder, C., Lascarides, A.: Using automatically labelled examples to classify rhetorical relations: A critical assessment. Natural Language Engineering **14**(3) (2008) 369–416

Identifying and Extracting Quantitative Data in Annotated Text*

Don J.M. Willems[1], Hajo Rijgersberg[1], and Jan L. Top[1,2]

[1] Wageningen UR, Food and Biobased Research
[2] Dept. of Computer Sciences, Vrije Universiteit Amsterdam

Abstract. In science it is difficult to reuse quantitative scientific data. For example, it is not possible to search for quantitative data in papers in a directed way, such as using the query "Select the storage modulus of dairy product A after the temperature has decreased from 90 to 4 °C". This is caused by the fact that data is made available in (relatively) free formats as in scientific papers, spreadsheets, or databases, all with limited annotation and description of the way they were obtained. Meaning is lost, for example about what the numbers relate to (quantities and units are often poorly indicated). Many researchers, especially in the physical and computer sciences use LaTeX in their creation of scientific papers. In this paper we present a set of LaTeX-style files, which use the terminology defined in wurvoc.org, that can be used to annotate scientific papers. These style files define a set of commands, each representing a specific quantity or unit. If the LaTeX is typeset into a PDF file, quantities and units in the PDF will be annotated with the appropriate references (URIs) to the corresponding concepts in the OM ontology. This will not only disambiguate the use of these quantities and units, but will also enable us to extract triples from the PDF, facilitating the use of SPARQL queries to answer advanced quantitative question.

1 Introduction

Many scientific papers are written using the LaTeX typesetting system and published as PDF. It is desirable to process this knowledge automatically. We propose a method to add semantic annotations to LaTeX files, extending the typesetting methods that LaTeX offers. Using these annotations we can automatically extract information from the generated PDF files.

In scientific research one is often looking for quantitative data. One can find these for example in external databases and spreadsheets or in scientific papers. However, it is difficult to look for quantitative data on the Web in a directed way. For example, the query "Select the storage modulus of dairy product A after the temperature has decreased from 90 to 4 °C" can not be carried out because computer tools are not able to link the correct numbers to the correct units and correct quantities. Another example is to give maximum, minimum and average values observed for parameter X in a set of papers. The problem is caused by the fact that data are expressed in

* This publication was supported by the Dutch national program COMMIT.

relatively free formats, such as in text or in spreadsheets. The structure of the data is often not clear for a computer. A lot of research is being done on parsing the structure of scientific papers including tables (see for instance [1]), which often contain a lot of quantitative data. The annotation of the data, however, is usually limited, including quantities and units that are often poorly indicated [2]. In other words, the context to which the numbers refer is often lost.

To approach these problems, formal terminology is developed in the discipline of the Semantic Web [3]. This terminology can be made publicly available through the Web, so that it can be used (referred to) in digital sources [4]. The ideal situation would be to have all information available in formal languages. This, however, requires a major effort in creating a consistent conceptual model of scientific knowledge and significant advances in automatic parsing and annotation of scientific texts. The work presented in this paper represents an important step in annotating texts with formal concepts.

Often quantitative work, in for instance the physical sciences, is presented in scientific papers that were created using the LaTeX typesetting system[5]. LaTeX is especially suited for the physical and computer sciences because of the extensive support for mathematical typesetting. The use of LaTeXstyle files provides a convenient way to add annotations to content, not only to the main text of an article but also to tables and even graphics created in LaTeX.

In this exercise we focus on the annotation and extraction of quantities and units, defined in our Ontology of units of Measure and related concepts (OM; see Section 3) from annotated content. Using custom LaTeX commands, quantities and units with semantic annotations are inserted into the PDF with the appropriate references (URIs) to the corresponding concepts in the OM ontology. This will not only disambiguate the use of these quantities and units, but will also enable us to extract RDF [6] triples from the PDF, enabling the use of SPARQL [7] queries to answer advanced quantitative questions. This may greatly enhance searching and the processing of data in general. Because a computer will suffer less from ambiguity owing to the annotations, it can reach a higher quality in the support that it offers.

In this paper we will first focus on related work (Section 2). Subsequently, in Section 3, we will briefly describe OM, discuss aliases in LaTeX, propose commands for referring to quantities and units in LaTeX, and describe the method for transforming equations from LaTeX to RDF. In Section 4 we will present a few examples of annotated scientific texts and in Section 5 we will discuss the results.

2 Related work

General developments in the area of semantic publishing, have been presented in several papers [8,9]. The authors express the need for annotating publications and extracting structured information from them. Otherwise the sheer number of references hinders interaction between related scientific activities. Assessing and integrating previous work benefits significantly if factual information is (also) available as Linked Open Data. These authors indicate the need for explicating the structure of arguments in scientific discourse, as well as a structured presentation of the con-

cepts and relations between concepts. Our work is complementary, in the sense that we start by disclosing numerical facts which can be related to other concepts. We take a pragmatic approach by building on standard practice in writing LaTeX documents.

Some approaches exist for semantically annotating LaTeX files. STEX, Semantic Markup for TeX/LaTeX [10], consists of a collection of TeX macro packages that allows the user to markup TeX/LaTeX documents semantically, turning the documents in a format for mathematical knowledge management (MKM). The method focuses on mathematical relations, collections, and formats of numbers (e.g., decimals). STEX, however, cannot be used to annotate quantities and units.

The SIunits package for LaTeX [11] is a package that provides support for typesetting units, in more or less the same way as we do. Quantities, however, do not appear as such in this package. Moreover, as the package only provides typesetting, the concepts are not linked to a centrally available vocabulary on the Semantic Web.

SALT, Semantically Annotated LaTeX[12], provides a means for externalising rhetorical and argumentation captured within a publication's content. However, the approach does not relate to mathematical concepts or quantities and units.

Mathematics can be expressed in XML using MathML [13] and can as such be incorporated into web pages. Equations in MathML can be expressed in two distinct formats: i) Presentation encoding, which as the name suggest supports the construction of traditional mathematical notation. ii) Content encoding supports the "encoding of the underlying mathematical meaning of an expression" [13]. While the scope of MathML itself does not include units, they can, however, be expressed in MathML [14], including a reference to the URI of a concept in a formal vocabulary of units (such as OM).

Ideally, structured information should be extracted automatically from publications. Frameworks such as GATE [15] provide functionality for automatically annotating text. It does not, however, provide the ability to automatically annotate equations, where only symbols are used for quantities and units, or graphics.

In previous work we have annotated quantities and units in Excel files, using an add-in for Excel we developed along with web services disclosing quantities and units from OM, and operations that can be performed on these terms, such as dimension and unit consistency checks on formulas and returning possible units for a particular quantity. In the present work we relate terms in LaTeX documents to this same ontology (OM [16]), extending the use of OM concepts to a broader audience.

3 Method

Typesetting (the creation of a visual representation of a text) using LaTeX is done using the TeX typesetting engine [17] developed by Donald Knuth in the 1970s and 1980s. TeX provides a set of low level declarations or *commands* for typesetting. LaTeX, developed by Leslie Lamport in the 1980s, provides a set of higher level commands that can be used to easily create documents without having to worry about their typographical appearance [5]. These sets of commands can easily be extended (and often are) by users to define a set of personal commands for typesetting particular pieces

of information that are often used. These commands are either defined in the main LaTeX source file or in style files which can be imported into the main LaTeX file.

In this paper we present a LaTeX package (as a set of style files) that uses terminology as offered through our ontology platform `wurvoc.org`.

3.1 Ontology of units of Measure and related concepts (OM)

The ontology that we use to refer to in the LaTeX files is OM. OM is an ontology based on older ontologies of units of measure, such as EngMath by Tom Gruber [18]. In earlier work [16] we have compared OM, EngMath and other ontologies, and OM appeared to be the most extended ontology, e.g. defining the most of the relevant concepts in the quantitative domain, such as "quantity", "unit of measure", "dimension", "measure", "measurement scale", etc.

OM defines concepts such as unit, quantity and dimension. Quantities are related to units of measure and measurement scales that can be used to express them using the relation `\unit_of_measure`. Units of measure are defined by some observable standard phenomenon, such as the length of the path travelled by light in a vacuum during a time interval of 1/299 792 458 of a second, for meter. Measures, such as "3 kilogram" are used to indicate values of quantities. Multiples and submultiples of units have a prefix, such as in kilogram and millimetre.

Systems of units organise quantities and units of measure, e.g. the International System of Units (SI). Such a system defines base units and derived units. Base units are units that cannot be defined in terms of other units (e.g. metre and second). Base units can be combined into derived units, such as for example metre per second ($\mathrm{m\,s^{-1}}$).

OM is based on a semi-formal description of the domain of units of measure, drafted from several paper standards that we have analysed, e.g. the Guide for the Use of the International System of Units [19], by the NIST. For a full list of statements, the sources that we have used, and ontological choices made, see previous work [16].

OM is modelled in OWL 2 [20]. The ontology is published as Linked Open Data [21] through our vocabulary and ontology portal wurvoc.[3] OM can be used freely under the Creative Commons 3.0 Netherlands license.

3.2 Aliases in LaTeX

When using LaTeX it is often preferable to create *aliases* for often used (complex) command structures instead of retyping these command structures again and again.

For instance, LaTeX source code becomes more difficult to interpret when units are used in an equation. To create a statement like:

$$G = 6.673 \times 10^{-11}\,\mathrm{N\,m^2\,kg^{-2}} \tag{1}$$

which is the gravitational constant, the following LaTeX source code can be used:

[3] http://www.wurvoc.org/vocabularies/om-1.8/. The objective of wurvoc.org is to publish vocabularies and associated web services relevant to the general domain of physical units and quantities and in particular the domains of life sciences and agrotechnology. In wurvoc one can browse vocabularies and directly interface with them.

```
G = 6.673\times 10^{-11} \mathrm{N} \mathrm{m^2} \mathrm{kg^{-2}}
```

Units are written in a non-bold but upright font (i.e. not cursive as is used for quantities and variables).

To make things easier, authors using LaTeX construct self-defined aliases. For instance, for the unit for the gravitational constant we might define a new command:

```
\newcommand{\Gunit}{\mathrm{N} \mathrm{m^2} \mathrm{kg^{-2}}}
```

and for exponents:

```
\newcommand{\E}[1]{\times10^{#1}}
```

The author can then use his custom defined commands (or aliases) `\Gunit` and `\E` to insert the correct unit and exponent. Equation 1 can then be typeset using

```
G = 6.673\E{-11} \Gunit
```

which is much easier to interpret by humans.

Sets of often used aliases can be created and distributed using style files. These LaTeX examples use custom commands to provide easier typesetting of mathematical expressions. We would like to use these typesetting commands (aliases) to insert semantic information into the mathematical expressions.

3.3 Semantic annotations

As aliases are used quite often by authors, it becomes possible to add extra information to the output produced when typesetting LaTeX files. The extra information we would like to provide in scientific texts are links (URIs) to ontological definitions of the quantities and units used as defined in the Ontology of units of Measure (OM, [22], prefix is `om:`). To this end we have created a set of style files (a package) that define a large set of aliases that not only create the correct symbols and layout for quantities and units, but also provides annotated links to the ontological concepts describing these quantities and units. As most LaTeX source files are typeset into PDF files these days, we have decided to use PDF annotations (more specifically hyperlinks) to create the links to the ontological concepts. The URI of the annotated concept is added to the PDF as a hyperlink and will generally only be visible when the mouse cursor hovers above the linked text.

Using the hyperref package, hyperlinks are inserted into the PDF produced by LaTeX by defining aliases with a custom command:

```
1  \newcommand{\annot}[3]{
2     \ifthenelse{\isempty{#3}}
3       {\href{om:#1}{#2}}
4       {\href{om:#1}{#3}}
5  }
```

In the first line, the command `\annot` is defined with three parameters. The first parameter is the part of the URI in the OM ontology of the concept (quantity or unit) that comes after `om:`, which is the base URI for the OM ontology. The second parameter is the default symbol used for this quantity or unit, and the third (optional) parameter can be used by an author to insert a custom symbol for the same concept. The second line checks whether the author has used an optional symbol. If not, the third line will insert the default symbol (the second parameter) with a hyperlink consisting of the base URI concatenated with the first parameter. If the author used the third parameter for a custom symbol, this symbol is used instead, with the same URI (line 4).

The `\annot` command is defined in the `om.sty` style file provided in our OM-LaTeX distribution. A few other commonly used commands, such as `\vect` to typeset vectors, `\E` to typeset exponents such as 4.2×10^3, and `\unit` to typeset units are also provided in this file.

To annotate an equation like:

$$||\mathbf{a}|| = 5.433 \times 10^{-1}\,\mathrm{m\,s^{-2}} \tag{2}$$

which would normally be produced by the source code:

```
||\vect{a}|| = 5.433 \times10^{-1}\unit{m}\unit{s^{-2}}
```

can now be obtained, with the same result, using the following code:

```
||\Acceleration || = 5.433\E{-1}\metrePerSecondSquared
```

This LaTeX code, while not much shorter, is more easy to interpret by humans. For all units and quantities in OM a human readable alias, such as `\Acceleration`, is provided in the LaTeX style files. Aliases from the SIUnits package [11] will also be included, so that texts created with SIUnits can easily be converted to include OM annotations.

More importantly, however, for our purposes, is the addition of the hyperlink pointing to the relevant concept. To facilitate this, the following aliases were defined:

```
1  \newcommand{\Acceleration}[1][]{
2     \annot{Acceleration}{\vect{a}}{#1}
3  }
4
5  \newcommand{\metrePerSecondSquared}[1][]{
6    \annot{metre_per_second_squared}{\unit{m} \unit{s^{-2}}}{#1}
7  }
```

In line 2, we use the `\annot` command to create an alias for the quantity acceleration with URI: `om:Acceleration` and default symbol '**a**'. In line 6 the same is done for the unit metre per second squared (URI: `om:metre_per_second_squared`). The first parameter to the `\annot` command (`Acceleration` in line 2, and `metre_per_second_squared` in line 6) provide semantic annotations to the mathematical expression. The second (`\vect{a}` in line 2, and `\unit{m} \unit{s^{-2}}` in line 6) and third parameters (`#1` in both line 2 and 6) are only concerned with typesetting.

All LaTeX commands representing quantities and units can also be used with user defined symbols simply by adding an (optional) parameter to a command. For instance, the command `\LuminousFlux` produces the symbol for the quantity luminous flux 'F' with a link to the related concept (`om:Lumious_flux`) in the OM ontology. If the author wants to use another symbol to represent luminous flux, he or she can achieve this by specifying the alternative symbol as an argument:
`\LuminousFlux[\Phi]` produces 'Φ', still linked with the same concept in the OM ontology. If desired sub- and superscripts can also be used in the argument:
`\LuminousFlux[F_{\lambda}]` produces 'F_λ', again linked with the same URI.

3.4 URI and equation extraction

When using the typesetting tool `pdflatex` to create PDF files from the LaTeX source, the URIs representing the unit and quantity concepts are inserted as hyperlinks into the PDF. To use these annotations we have to parse the PDF files to find the hyperlinks (URIs). Using Apache's PDFBox `http://pdfbox.apache.org/` we were able to create a small Java tool to parse the PDF files and extract the URIs representing concepts in OM and linking these URIs to the text.

Using this setup we are able to extract OM concepts (units and quantities) from a text generated with OM-annotated LaTeX. We would, however, also like to extract the semantics of statements like $V = 15.2\,\mathrm{m}^3$ (i.e. we would like to extract the fact that the quantity *volume* has a value of 15.2 in units of *cubic metre*). To this end we have also added the functionality of finding binary (=, <, >, ≈, etc.) relations in the text to the extraction tool.

When a PDF is parsed by the extraction tool, URIs for units and quantities, numeric values and binary relations are tagged in the text. Operators, like `\E` are also recognised, and in the case of exponents, the value is changed accordingly (e.g. 5.2×10^3 is changed to 5200). The tool then applies rules to find patterns in the text like:

```
[QUANTITY] [BINARY_RELATION] [VALUE] [UNIT]
```

If the tool comes across such a pattern, the combination of quantity, relation, value, and unit is stored. For instance the equation:

$$E_\mathrm{k} = 1.209 \times 10^{-2}\,\mathrm{eV} \tag{3}$$

is extracted as:

```
1 [QUANTITY=om:Kinetic_energy] [BINARY_RELATION='=']
2        [VALUE=0.01209] [UNIT=om:electronvolt]
```

In this manner quantitative statements can be extracted from PDF generated with OM annotated LaTeX.

3.5 Transformation to RDF

The result of the extraction can then be transformed into RDF statements using the OM ontology. For instance, the following equation

$$F = 15.2\,\mathrm{N} \tag{4}$$

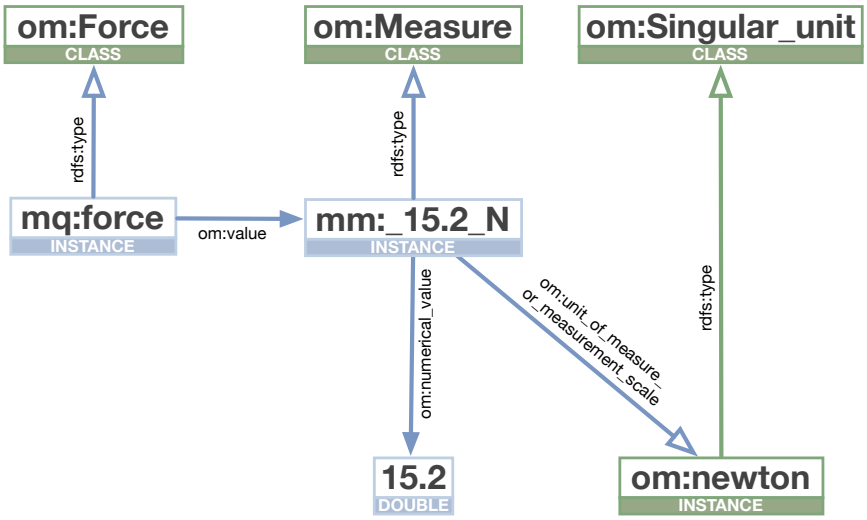


Fig. 1. Extracted RDF representing Equation 4. This graph only represents OM-specific data; other information such as provenance data are present in the full RDF graph.

can be transformed into RDF (in turtle format [23]):

```
mq:force om:value mm:_15.2_N;
    a om:Force .
mm:_15.2_N a om:Measure ;
    om:numerical_value "15.2"^^xsd:double ;
    om:unit_of_measure_or_measurement_scale om:newton .
```

where `mm` and `mq` are prefixes for custom defined namespaces (possibly pointing to the URI for the original text, thereby ensuring provenance) for measures and quantities respectively, and `om` is the prefix for the OM namespace. This statement can also be visualised as a graph (Figure 1).

The current extraction tool is not only able to create the statements to model the equation in RDF, but it is also able to export these RDF statements to an RDF triple store, where it can be combined with other semantic data extracted from the PDF, or obtained from other sources.

4 Real-world examples

The number of detected measures depends on the type of paper; experimental papers tend to contain more measures than theoretical papers. For example the fifth page of a paper on water vapour sorption in gluten and starch films contains the following text:

> [...] The obtained parameter values for starch films ($T_g = 540 \pm 10\,\mathrm{K}$ and $\Delta C_p = 0.32 \pm 0.02\,\mathrm{J\,g^{-1}\,K^{-1}}$) differ from the values obtained by van der Sman and

> Meinders (2011) from the data of starches from different botanical sources and obtained with different experimental techniques ($T_g \approx 475\,K$, $\Delta C_p \approx 0.43\,J\,g^{-1}\,K^{-1}$) but are in close agreement with the values obtained by Bizot et al. (1997) for pea amylose determined using DSC (T_g = 589 K and $\Delta C_p = 0.27\,J\,g^{-1}\,K^{-1}$). [...][4]

which contained six measures. Our extraction tool extracted all six measures correctly, for example, the following RDF triples were extracted for the last measure of change of water specific heat ($\Delta C_p = 0.27\,J\,g^{-1}\,K^{-1}$):

```
1    mq:specific_heat om:value mm:_0.27_JpgK;
2        a om:Specific_heat.
3    mm:_0.27_JpgK rdfs:type om:Measure;
4        om:numerical_value "0.27"^^xsd:double;
5        om:unit_of_measure_or_measurement_scale
5                om:joule_per_gram_kelvin.
```

As one can see the measure has been extracted successfully and is correctly modelled in terms of OM. At the moment, we assume that ΔC_p is one symbol for a specific quantity and not a mathematical operation. Finally, it is not possible to add error values to the conceptual model (e.g. $T_g = 540 \pm 10\,K$) and to distinguish between = and $\approx$.

The extracted triples do not specify the source of the specific heat (i.e. the specific heat for pea amylose determined using DSC). This is a case for further (automatic) semantic annotation beyond OM. Including more extensive annotation would make the semantic information even more valuable. One could start searching in scientific RDF databases for articles on "specific water vapour heat" with a value between "0.2" and "$0.3\,J\,g^{-1}\,K^{-1}$".

As a second example of measures in an experimental paper consider the abstract of a paper on observations of a young star. The abstract alone contains six measures:

> We present CS(J=2-1) interferometric observations obtained with the Nobeyama Millimeter Array (NMA) toward a protostar (GH2O 092.67 + 03.07) in the Cygnus OB7 giant molecular cloud (distance = 800 pc). The data clearly indicate the presence of a compact (size $\simeq 8 \times 10^3$ AU) and young out-flow with dynamical time scale $\simeq$ 3500 year. [...] We derive a total mass of $\simeq 0.6\,M_\odot$ and $\simeq 12\,M_\odot$ for the outflow and disk respectively. The comparison of the NMA data with a simple model of infalling disk indicates a mass of the central object in the range $4.0 < M < 7.5\,M_\odot$. [...][5]

In this example the names of the quantities are annotated as text (not math, e.g. 'size $\simeq 8 \times 10^3$ AU') as in: `\Diameter[size]` and is actually parsed correctly by our extraction tool:

[4] Laura Oliver, Marcel B.J. Meinders, Dynamic water vapour sorption in gluten and starch films, Journal of Cereal Science, **54**-3 (2011), pages 409-416.

[5] Bernard, J.P., Dobashi, K. , Momose, M.: Out flow and disk around the very young massive star GH2O 092.67+03.07. Astronomy and Astrophysics **350** (1999), pages 197-203.

```
mq:size om:value mm:_8000_AU;
    a om:Diameter.
mm:_8000_AU rdfs:type om:Measure;
    om:numerical_value "8000.0"^^xsd:double;
    om:unit_of_measure_or_measurement_scale
              om:astronomical_unit.
```

Please note that the numerical value containing an exponent (8×10^3) is interpreted correctly (8000).

5 Discussion

Embedding numerical facts in otherwise textual documents incurs a tension between the use of natural language and structured formats. We submit that scientists should be able to put their arguments forward with minimal technical constraints. On the other hand, embedding RDF-OWL type annotations eliminates ambiguity and simplifies computer processing. For example, consider the following statement:

> The water vapor permeability for optimal crispness and crumb softness retention was 8×10^{-9} g/(m s Pa).[6]

It is possible to request the author to annotate individual quantities and units of measure (and concepts), but it would also be possible to have the author provide RDF triples for the entire sentence. The first option seems less attractive from a computer processing perspective. In that case, more effort is required to parse the information into an equivalent RDF triple afterwards. Nevertheless we choose to stay close to normal writing as much as possible. By annotating at the level of quantities and units only, precisely enough formalisation is provided to enable automated construction of the composite triple. Moreover, by using the alias mechanism provided by LaTeX and our definition of `\annot`, the natural language style is approximated as much as possible.

This paper describes how units and quantities can be annotated. However, the value of such annotation is limited if it is not clear to which objects or phenomena these quantities refer. For example, $V = 15.2\,\mathrm{m}^3$ only becomes a useful fact if we know that it refers to a container containing water, or even to a specific container in an experiment. This would require annotating objects and phenomena using domain-specific ontologies, and relating them to the quantities used. A simple generalisation of our approach is to include the full URI in the `\annot` construct. This allows the user to link any object to an ontological class or instance. However, some heuristic processing would still be needed to link these objects to the annotated quantities. We consider this a necessary step in our method, but beyond the scope of the present paper.

[6] Anita Hirte, Rob J. Hamer, Marcel B.J. Meinders, Kevin van de Hoek, Cristina Primo-Martín. Control of crust permeability and crispness retention in crispy breads. Food Research International, Volume 46, Issue 1, April 2012, Pages 92-98

Finally we note that OM, the ontology of quantities and units, is accessible through a set of web services that provide additional functionality if data is annotated along the above lines. They allow automatic checking of combinations of units and quantities for correctness and completeness, but also automatic unit conversions. These can be useful aids during paper writing or reading.

6 Conclusion

For the research presented in this paper we have created a set of style files for LaTeX that refer to concepts from an ontology of units and quantities. By using the commands used in these style files, quantities and units are annotated directly. The concept's URI is included as a hyperlink when generating the PDF. Using these annotations we have been able to extract triples from the PDF and insert them into an RDF triple store, which can be queried with specific querying constraints. The LaTeX style files and corresponding PDF extraction tool will be made available in the near future.

In a broader sense it becomes feasible to do more with the annotated data, such as unit conversion, checking of dimension and unit consistency, integrating, performing computational methods on the data, etc. This functionality is available via OM web services [16]. To make the data even more reusable, it will be important to extract other concepts than quantities and units, such as the object or event that a value of such quantity refers to.

Ideally we should be able to annotate existing papers automatically. Frameworks such as GATE [15], which provide automatic annotation will play an important role in this endeavour. In earlier work [2] we have drafted heuristic rules for interpreting and formalising quantitative information in spreadsheets. This research could be extended towards quantitative information in scientific papers. At this moment, however, automatic annotation of measurements cannot be performed reliably enough in the cases we observed, which are intrinsically ambiguous and incomplete [2]. So, manual (and therefore, user-validated) annotation by authors is still required. The described method in this paper helps to annotate quantitative concepts such as quantities and units of measure, using the embedded URLs.

As the user will likely be using alias commands in LaTeX anyway, extending these with semantic annotations does not require extra effort for the user and these annotations are, therefore, relatively for free. The user only needs to include the OM-LaTeX package and is then able to use aliases with names close to the actual names of the units or measures, making the text easier to read.

In the light of extending this approach, we aim to investigate whether integration with STEX, SIunits, or SALT is possible. And as it is useful to annotate mathematical relations and operators, we will, moreover, define the URIs for relations and operators and more in OQR (Ontology of Quantitative Research) [24].

References

1. Oro, E., Ruffolo, M.: PDF-TREX: An Approach for Recognizing and Extracting Tables from PDF Documents. In: Proceedings of the 10th International Conference on Document Analysis and Recognition. (july 2009) 906 –910

2. van Assem, M., Rijgersberg, H., Wigham, M., Top, J.: Converting and Annotating Quantitative Data Tables. In Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J., Horrocks, I., Glimm, B., eds.: Proc. 9th Int'l Semantic Web Conf. (ISWC'10). Number 6496 in LNCS, Springer-Verlag (2010)
3. Berners-Lee, T., Hendler, J.: Publishing on the Semantic Web. Nature (April 26) (2001) 1023–1025
4. Hey, T., Trefethen, A.: Cyberinfrastructure for e-science. Science **308** (2005) 817 – 821
5. Mittelbach, F., Goossens, M., Carlisle, J.B.D., Rowley, C.: The LaTeXCompanion, 2nd edition (TTCT series). Addison-Wesley, Reading, Massachusetts
6. W3C: Resource Description Framework (RDF). http://www.w3.org/RDF/ (2004)
7. Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF. http://www.w3.org/TR/rdf-sparql-query/ (2008)
8. De Waard, A.: From proteins to fairytales: Directions in semantic publishing. IEEE Intelligent Systems (March/April) (2010)
9. Shum, S.B., Clark, T., de Waard, A., Groza, T., Handschuh, S., Sandor, A.: Scientific discourse on the semantic web : A survey of models and enabling technologies. Semantic Web Journal Interoperability Usability Applicability (Special Issue on Survey Articles) (2010)
10. Kohlhase, M.: Semantic Markup for TEX/LATEX. (2004)
11. Heldoorn, M.: The SIunits package. Consistent applications of SI units. (2007)
12. Groza, T., Handschuh, S., MŽller, K., Decker, S.: Salt - semantically annotated latex for scientific publications. Lecture Notes in Computer Science **4519** (2007) 518–532
13. Ausbrooks, R., Buswell, S., Carlisle, D., Chavchanidze, G., Dalmas, S., Devitt, S., Diaz, A., Dooley, S., Hunter, R., Ion, P., Kohlhase, M., Lazrek, A., Libbrecht, P., Miller, B., Miner, R., Rowley, C., Sargent, M., Smith, B., Soiffer, N., Sutor, R., Watt, S.: Mathematical Markup Language (MathML) Version 3.0. http://www.w3.org/TR/MathML3/ (2010)
14. Harder, D.W., Devitt, S.: Units in MathML. http://www.w3.org/TR/mathml-units/ (2003)
15. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A framework and graphical development environment for robust NLP tools and applications. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics. (2002)
16. Rijgersberg, H., Wigham, M., Top, J.L.: How semantics can improve engineering processes: A case of units of measure and quantities. Advanced Engineering Informatics **25**(2) (2011) 276–287
17. Knuth, D.E.: The TeXbook. Addison-Wesley (1986)
18. Gruber, T., Olsen, G.: An Ontology for Engineering Mathematics. In: Fourth International Conference on Principles of Knowledge Representation and Reasoning, Morgan Kaufmann (1994)
19. Taylor, B.N.: Guide for the use of the International System of Units (SI). 2008 edn. Technical report, National Institute of Standards and Technology (2008)
20. W3C: Owl 2 web ontology language. Technical report, World Wide Web Consortium (W3C) (2009)
21. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data - The Story So Far. International Journal on Semantic Web and Information Systems **5** (2009) 1–22
22. Rijgersberg, H., van Assem, M., Wigham, M., Broekstra, J., Top, J.: Ontology of units of measure (OM). http://www.wurvoc.org/vocabularies/om-1.8/ (2010)
23. Beckett, D., Berners-Lee, T.: Turtle - Terse RDF Triple Language. http://www.w3.org/TeamSubmission/turtle/ (2011)
24. Rijgersberg, H., Top, J.L., Meinders, M.: Semantic Support for Quantitative Research Processes. Intelligent Systems **24**(1) (2011) 37–46

Scenario-Driven Selection and Exploitation of Semantic Data for Optimal Named Entity Disambiguation

Panos Alexopoulos, Carlos Ruiz, and José-Manuel Gómez-Pérez

iSOCO, Avda del Partenon 16-18, 28042, Madrid, Spain,
{palexopoulos,cruiz,jmgomez}@isoco.com

Abstract. The rapidly increasing use of large-scale data on the Web has made named entity disambiguation a key research challenge in Information Extraction (IE) and development of the Semantic Web. In this paper we propose a novel disambiguation framework that utilizes background semantic information, typically in the form of Linked Data, to accurately determine the intended meaning of detected semantic entity references within texts. The novelty of our approach lies in the definition of a structured semi-automatic process that enables the custom selection and use of the semantic data that is optimal for the disambiguation scenario at hand. This process allows our framework to adapt to the particular characteristics of different domains and scenarios and, as experiments show, to be more effective than approaches primarily designed to work in open domain and unconstrained situations.

1 Introduction

Information Extraction (IE) involves the automatic extraction of structured information from texts, such as entities and their relations, in an effort to make the information of these texts more amenable to applications related to Question Answering, Information Access and the Semantic Web. In turn, named entity resolution is an IE subtask that involves detecting mentions of named entities (e.g. people, organizations or locations) within texts and mapping them to their corresponding entities in a given knowledge source. The typical problem in this task is ambiguity, i.e. the situation that arises when a term may refer to multiple different entities. For example, "Tripoli" may refer, among others, to the capital of Libya or to the city of Tripoli in Greece. Deciding which reference is the correct one in a given text is a challenging task which a significant number of approaches have been trying to facilitate for a long time now [2] [3] [6] [7] [5] [8].

The majority of these approaches rely on the strong contextual hypothesis of Miller and Charles [9] according to which terms with similar meanings are often used in similar contexts. The role of these contexts, which practically serve as disambiguation evidence, is typically played by already annotated documents (e.g. wikipedia articles) which are used to train term classifiers. These classifiers

link a term to its correct meaning entity, based on the similarity between the term's textual context and the contexts of its potential entities [8] [10].

An alternative kind of disambiguation evidence that has recently begun to be used are semantic structures like ontologies and Linked Data [7] [6] [12]. The respective approaches typically employ graph-related measures to determine the similarity between the graph formed by the entities found within the ambiguous term's textual context and the graphs formed by each candidate entity's "neighbor" entities in the ontology. The candidate entity with the best matching graph is assumed to be the correct one.

An obvious limitation of this is the need for comprehensive semantic information as input to the system; nevertheless the increasing availability of such information on the Web, typically in the form of Linked Data, can help overcome this problem to a significant degree. Still, however, effectiveness of these approaches is highly dependent on the degree of alignment between the content of the texts to be disambiguated and the semantic data to be used. This means that the ontology's elements (concepts, instances and relations) should cover the domain(s) of the texts to be disambiguated but should not contain other additional elements that i) do not belong to the domain or ii) do belong to it but do not appear in the texts.

To show why this is important, consider an excerpt from a contemporary football match description saying that *"Ronaldo scored two goals for Real Madrid"*. To disambiguate the term "Ronaldo" in this text using an ontology, the only contextual evidence that can be used is the entity "Real Madrid", yet there are two players with that name that are semantically related to it, namely Cristiano Ronaldo (current player) and Ronaldo Luis Nazario de Lima (former player). Thus, if both relations are considered then the term will not be disambiguated. Yet, the fact that the text describes a contemporary football match suggests that, in general, the relation between a team and its former players is not expected to appear in it. Thus, for such texts, it would make sense to ignore this relation in order to achieve more accurate disambiguation.

Unfortunately, current approaches do not facilitate such a fine-grained control over which parts of a given ontology should be used for disambiguation in a given scenario and which not. Some of them allow the constraining of the concepts to which the potential entities may belong [6] [8], but they do not do the same for relations nor do they provide any structured process and guidelines for better execution of this task. That is because their goal is to build scenario and domain independent disambiguation systems where a priori knowledge about what entities and relations are expected to be present in the text is usually unavailable. Indeed, this is the case in scenarios involving news articles, blog posts, tweets and generally texts whose exact content cannot really be predicted. Yet there can be also specialized scenarios where such predictions can be safely made.

One such scenario is the one above about football match descriptions. This was in the context of the project BuscaMedia[1] and involved the disambiguation of football related entities within texts describing highlights of football matches.

[1] http://www.cenitbuscamedia.es/

The nature of these texts made safe the assumption that the entities expected to be found in them were players, coaches and teams and that the relations implied between them were the ones of current membership (i.e. players and coaches related to their current team). A similarly specialized scenario was in the project GLOCAL[2], involving the disambiguation of location entities within historical texts describing military conflicts. Again, the nature of these texts allowed us to expect to find in them, among others, military conflicts, locations where these conflicts took place and people and groups that participated in them.

Given that, in this paper we define a novel ontology-based disambiguation framework that is particularly applicable to similar to the above scenarios where knowledge about what entities and relations are expected to be present in the texts is available. Through a structured semi-automatic process the framework enables i) the exploitation of this a priori knowledge for the selection of the subset of domain semantic information that is optimal for the disambiguation scenario at hand, ii) the use of this subset for the generation of disambiguation evidence and iii) the use of this evidence for the disambiguation of entities within the scenario's texts. As we will show in the rest of the paper, this process allows our system to be more effective in such constrained scenarios than other disambiguation approaches designed to work in unconstrained ones.

The rest of the paper is as follows. Section 2 presents related work while section 3 describes in detail our proposed framework. Section 4 presents experimental results regarding the framework's effectiveness in the two application scenarios mentioned above. Finally, in sections 5 and 6 we make a critical discussion of our work, we summarize its key aspects and we outline the potential directions it could take in the future.

2 Related Work

A recent ontology-based entity disambiguation approach is described in [7] where an algorithm for entity reference resolution via Spreading Activation on RDF Graphs is proposed. The algorithm takes as input a set of terms associated with one or more ontology elements and uses the ontology graph and spreading activation in order to compute Steiner graphs, namely graphs that contain at least one ontology element for each entity. These graphs are then ranked according to some quality measures and the highest ranking graph is expected to contain the elements that correctly correspond to the entities.

Another approach is that of [4] where the application of restricted relationship graphs (RDF) and statistical NLP techniques to improve named entity annotation in challenging Informal English domains is explored. The applied restrictions are i) domain ones where various entities are a priori ruled out and ii) real world ones that can be identified using the metadata about entities as they appear in a particular post (e.g. that an artist has released only one album, or has a career spanning more than two decades).

[2] http://glocal-project.eu/

In [5] Hassel et al. propose an approach based on the DBLP-ontology which disambiguates authors occurring in mails published in the DBLP-mailing list. They use ontology relations of length one or two, in particular the co-authorship and the areas of interest. Also, in [12] the authors take into account the semantic data's structure, which is based on the relations between the resources and, where available, the human-readable description of a resource. Based on these characteristics, they adapt and apply two text annotation algorithms: a structure based one (Page Rank) and a content-based one.

Several approaches utilize Wikipedia as a highly structured knowledge source that combines annotated text information (articles) and semantic knowledge (through the DBPedia[3] [1] and YAGO [13] ontologies). For example, DBPedia Spotlight [8] is a tool for automatically annotating mentions of DBPedia resources in text by using i) a lexicon that associates multiples resources to an ambiguous label and which is constructed from the graph of labels, redirects and disambiguations that DBPedia ontology has and ii) a set of textual references to DBPedia resources in the form of Wikilinks. These references are used to gather textual contexts for the candidate entities from wikipedia articles and use them as disambiguation evidence.

A similar approach that uses the YAGO ontology is the AIDA system [6] which combines three entity disambiguation measures: the prior probability of an entity being mentioned, the similarity between the contexts of a mention and a candidate entity, and the semantic coherence among candidate entities for all mentions together. The latter is calculated based on the distance between two entities in terms of type and subclassOf edges as well as the number of incoming links that their Wikipedia articles share.

The difference between the above approaches and our framework is detected in the way they treat the available semantic data. For example, Spotlight uses the DBPedia ontology only as an entity lexicon without really utilizing any of its relations, apart from the redirect and disambiguation ones. Thus, it's more text-based than ontology-based. On the other hand, AIDA builds an entity relation graph by considering only the type and subclassOf relations as well as "assumed" relations inferred by the links within the articles. The problem with this approach is that important semantic relations that are available in the ontology are not utilized and, of course, there is no control over which edges of the derived ontology graph should be utilized in the given scenario. Such control is not provided either in [7] or any of the rest aforementioned approaches except for that of [5] which, however, is specific for the scientific publications domain.

3 Proposed Disambiguation Framework

Our framework targets the task of entity disambiguation based on the intuition that a given ontological entity is more likely to represent the meaning of an ambiguous term when there are many ontologically related to it entities in the

[3] http://dbpedia.org

text. These related entities can be seen as **evidence** whose quantitative and qualitative characteristics can be used to determine the most probable meaning of the term. For example, consider a historical text containing the term "Tripoli". If this term is collocated with terms like *"Siege of Tripolitsa"* and *"Theodoros Kolokotronis"* (the commander of the Greeks in this siege) then it is fair to assume that this term refers to the city of Tripoli in Greece rather than the capital of Libya.

Nevertheless, as we already showed in the introduction, which entities and to what extent should serve as evidence in a given scenario depends on the domain and expected content of the texts that are to be analyzed. For that, the key ability our framework provides to its users is to construct, in a semi-automatic manner, semantic evidence models for specific disambiguation scenarios and use them to perform entity disambiguation within them. In particular, our framework comprises the following components:

- A **Disambiguation Evidence Model** that contains, for a given scenario, the entities that may serve as disambiguation evidence for the scenario's target entities (i.e. entities we want to disambiguate). Each pair of a target entity and an evidential one is accompanied by a degree that quantifies the latter's evidential power for the given target entity.
- A **Disambiguation Evidence Model Construction Process** that builds, in a semi-automatic manner, a disambiguation evidence model for a given scenario.
- An **Entity Disambiguation Process** that uses the evidence model to detect and extract from a given text terms that refer to the scenario's target entities. Each term is linked to one or more possible entity uris along with a confidence score calculated for each of them. The entity with the highest confidence should be the one the term actually refers to.

In the following paragraphs we elaborate on each of the above components.

3.1 Disambiguation Evidence Model and its Construction

For the purposes of this paper we define an ontology as a tuple $O = \{C, R, I, i_C, i_R\}$ where

- C is a set of concepts.
- I is a set of instances.
- R is a set of binary relations that may link pairs of concept instances.
- i_C is a concept instantiation function $C \rightarrow I$.
- i_R is a relation instantiation function $R \rightarrow I \times I$.

The **Disambiguation Evidence Model** defines for each ontology instance which other instances and to what extent should be used as evidence towards its correct meaning interpretation. More formally, given a domain ontology O, a disambiguation evidence model is defined as a function $dem : I \times I \rightarrow [0, 1]$. If

$i_1, i_2 \in I$ then $dem(i_1, i_2)$ is the degree to which the existence, within the text, of i_2 should be considered an indication that i_1 is the correct meaning of any text term that has i_1 within its possible interpretations.

To construct the optimal evidence model for a given disambiguation scenario we proceed as follows: First, based on the scenario, we determine the concepts the instances of which we wish to disambiguate (e.g. players, teams and managers for the football match scenario). Then, for each of these concepts, we determine the related to them concepts whose instances may serve as contextual disambiguation evidence. The result of the above analysis should be a disambiguation evidence concept mapping function $ev_C : C \rightarrow C \times R^n$ which given a target concept $c_t \in C$ returns the concepts which may act as evidence for it along with the ontological relations whose composition links this concept to the target one. Table 1 contains an example of such a function for the football match descriptions scenario where, for instance, soccer players provide evidence for other soccer players that play in the same team. This mapping, shown in the second row of the table, is facilitated by the composition of the relations **dbpprop:currentclub** (that relates players to their current teams) and its inverse one **is dbpprop:currentclub of** (that relates teams to their current players). Table 2 illustrates a similar mapping for the military conflict texts scenario.

Table 1. Sample Disambiguation Evidence Concept Mapping for Football Match Descriptions

Target Concept	**Evidence Concept**	**Relation(s) linking Evidence to Target**
dbpedia-owl:SoccerPlayer	dbpedia-owl:SoccerClub	is dbpprop:currentclub of
dbpedia-owl:SoccerPlayer	dbpedia-owl:SoccerPlayer	dbpprop:currentclub, is dbpprop:currentclub of
dbpedia-owl:SoccerClub	dbpedia-owl:SoccerPlayer	dbpprop:currentclub
dbpedia-owl:SoccerClub	dbpedia-owl:SoccerManager	dbpedia-owl:managerClub
dbpedia-owl:SoccerManager	dbpedia-owl:SoccerClub	is dbpedia-owl:managerClub of

Using the disambiguation evidence concept mapping, we can then automatically derive the disambiguation evidence model *dem* as follows: Given a target concept $c_t \in C$ and an evidence concept $c_e \in C$ then for each instance $i_t \in i_C(c_t)$ and $i_e \in i_C(c_e)$ that are related to each other through the composition of relations $\{r_1, r_2, ..., r_n\} \in ev_C(c_t)$ we derive the set of instances $I_t \subseteq I$ which share common names with i_t and are also related to i_e through $\{r_1, r_2, ..., r_n\} \in ev_C(c_t)$. Then the value of *dem* for this pair of instances is computed as follows:

$$dem(i_t, i_e) = \frac{1}{|I_t|} \tag{1}$$

Table 2. Sample Disambiguation Evidence Concept Mapping for Military Conflict Texts

Target Concept	**Evidence Concept**	**Relation(s) linking Evidence to Target**
dbpedia-owl:PopulatedPlace	dbpedia-owl:MilitaryConflict	dbpprop:place
dbpedia-owl:PopulatedPlace	dbpedia-owl:MilitaryConflict	dbpprop:place, dbpedia-owl:isPartOf
dbpedia-owl:PopulatedPlace	dbpedia-owl:MilitaryPerson	is dbpprop:commander of, dbpprop:place
dbpedia-owl:PopulatedPlace	dbpedia-owl:PopulatedPlace	dbpedia-owl:isPartOf
dbpedia-owl:MilitaryPerson	dbpedia-owl:MilitaryConflict	dbpprop:commander

The intuition behind this formula is that the evidential power of a given entity is inversely proportional to the number of different target entities it provides evidence for. If, for example, a given military person has fought in many different locations with the same name, then its evidential power for this name is low.

3.2 Entity Disambiguation Process

The entity reference resolution process for a given text document and a disambiguation evidence model starts by extracting from the text the set of terms T that match to some instance belonging to a target or an evidence concept, that is some $i \in i_C(c)$, $c \in C_t \cup C_e$. Along with that we derive a term-meaning mapping function $m : T \rightarrow I$ that returns for a given term $t \in T$ the instances it may refer to. We also consider I_{text} to be the superset of these instances.

Then we consider the set of potential target instances found within the $I^t_{text} \subseteq I_{text}$ and for each $i_t \in I^t_{text}$ we derive all the instances i_e from I_{text} for which $dem(i_t, i_e) > 0$. Subsequently, by combining the evidence model dem with the term meaning function m we are able to derive an entity-term support function $sup : I^t_{text} \times T \rightarrow [0, 1]$ that returns for a target entity $i_t \in I^t_{text}$ and a term $t \in T$ the degree to which t supports i_t:

$$sup(i_t, t) = \frac{1}{|m(t)|} \sum_{i_e \in m(t)} dem(i_t, i_e) \tag{2}$$

Using this function we are able to calculate for a given term in the text the confidence that it refers to the entity $i_t \in m(t)$ as follows:

$$conf(i_t) = \frac{\sum_{t \in T} K(i_t, t)}{\sum_{i'_t \in m(t)} \sum_{t \in T} K(i'_t, t)} * \sum_{t \in T} sup(i_t, t) \tag{3}$$

where $K(i_t, t) = 1$ if $sup(i_t, t) > 0$ and 0 otherwise. In other words, the overall support score for a given candidate target entity is equal to the sum of the

entity's partial supports (i.e. function *sup*) weighted by the relative number of terms that support it. It should be noted that in the above process we adopt the one referent per discourse approach which assumes one and only one meaning for a term in a discourse.

4 Framework Application and Evaluation

To evaluate the effectiveness of our framework we applied it in the two scenarios we mentioned in the introduction, the one involving disambiguation in football match descriptions and the other in texts describing military conflicts. In both cases we used DBPedia as a source of semantic information and we i) defined a disambiguation evidence model for each scenario and ii) used these models to perform entity disambiguation in a representative set of texts. Then we measured the precision and recall of the process. Precision was determined by the fraction of correctly interpreted terms (i.e. terms for which the interpretation with the highest confidence was the correct one) to the total number of interpreted terms (i.e. terms with at least one interpretation). Recall was determined by the fraction of correctly interpreted terms to the total number of annotated terms in the input texts. It should be noted that all target terms for disambiguation in the input texts were known to the knowledge base (i.e. DBPedia).

Finally, the results of the above evaluation process were compared to those achieved by two publicly available semantic annotation and disambiguation systems, namely DBPedia Spotlight [4] [8], AIDA[5] [6]. The two systems were chosen for comparison because i) they also use DBPedia as a knowledge source and ii) they provide some basic mechanisms for constraining the types of entities to be disambiguated, though not in the same methodical way as our framework does. Practically, the two systems merely provide the users the capability to select the classes whose instances are to be included in the process. In all cases, it should be made clear that the goal of this comparison was not to disprove the effectiveness and value of these systems as tools for open domain and unconstrained situations but rather to verify our claim that our approach is more appropriate for disambiguation in "controlled" scenarios, i.e. scenarios in which a priori knowledge about what entities and relations are expected to be present in the text is available. A useful evaluation of popular semantic entity recognition systems for open scenarios may be found at [11].

4.1 Football Match Descriptions Scenario

In this scenario we had to semantically annotate a set of textual descriptions of football match highlights like the following: *"It's the 70th minute of the game and after a magnificent pass by Pedro, Messi managed to beat Claudio Bravo. Barcelona now leads 1-0 against Real."*. These descriptions were used as metadata of videos showing these highlights and our goal was to determine, in an

[4] http://dbpedia-spotlight.github.com/demo/index.html
[5] https://d5gate.ag5.mpi-sb.mpg.de/webaida/

unambiguous way, which were the participants (players, coaches and teams) in each video. The annotated descriptions were then to be used as part of a semantic search application where users could retrieve videos that showed their favorite player or team, with much higher accuracy.

To achieve this goal, we applied our framework and we built a disambiguation evidence model, based on DBPedia, that had as an evidence mapping function that of table 1. This function was subsequently used to automatically calculate (through equation 1) the function *dem* for all pairs of target and evidence entities. Table 3 shows a small sample of these pairs where, for example, Getafe acts as evidence for the disambiguation of Pedro Leon because the latter is a current player of it. Its evidential power, however, for that player is 0.5, since in the same team there is another player with the same name (i.e. Pedro Rios Maestre).

Table 3. Examples of Target-Evidential Entity Pairs for the Football Scenario

Target Entity	**Evidential Entity**	**dem**
dbpedia:Real_Sociedad	dbpedia:Claudio_Bravo_(footballer)	1.0
dbpedia:Pedro_Rodriguez_Ledesma	dbpedia:FC_Barcelona	1.0
dbpedia:Pedro_Leon	dbpedia:Getafe_CF	0.5
dbpedia:Pedro_Rios_Maestre	dbpedia:Getafe_CF	0.5
dbpedia:Lionel_Messi	dbpedia:FC_Barcelona	1.0

Using this model, we applied our disambiguation process in 50 of the above texts, all containing ambiguous entity references. The overall number of references was 126 with about 90% of them being ambiguous. In average, each ambiguous entity reference had 3 possible interpretations with player names being the most ambiguous. Table 4 shows the results achieved by our approach as well as by DBPedia Spotlight and AIDA. It should be noted that when using the latter systems, we used their concept selection facilities in order to constrain the space of possible interpretations. Still, as one can see from the table data, the constraining of the semantic data that our custom disambiguation evidence model facilitated (e.g. the consideration of only the current membership relation between players and teams) was more effective and managed to yield significantly better results.

Table 4. Entity Disambiguation Evaluation Results in the Football Scenario

System/Approach	**Precision**	**Recall**	F_1 **Measure**
Proposed Approach	84%	81%	82%
AIDA	62%	56%	59%
DBPedia Spotlight	85%	26%	40%

4.2 Military Conflict Texts Scenario

In this scenario our task was to disambiguate location references within a set of textual descriptions of military conflicts like the following: *"The Siege of Augusta was a significant battle of the American Revolution. Fought for control of Fort Cornwallis, a British fort near Augusta, the battle was a major victory for the Patriot forces of Lighthorse Harry Lee and a stunning reverse to the British and Loyalist forces in the South"*. For that we used again DBPedia and we defined the disambiguation evidence mapping function of table 2 which, in turn, produced the evidence model that is (partially) depicted in table 5.

Table 5. Examples of Target-Evidential Entity Pairs for the Miltary Conflict Scenario

Location	**Evidential Entity**	**dem**
dbpedia:Columbus,_Georgia	James H. Wilson	1.0
dbpedia:Columbus,_New Mexico	dbpedia:Pancho_Villa	1.0
dbpedia:Beaufort_County,_South_Carolina	dbpedia:Raid_at_Combahee_Ferry	1.0
dbpedia:Beaufort_County,_South_Carolina	dbpedia:James_Montgomery_(colonel)	1.0
dbpedia:Beaufort_County,_North_Carolina	dbpedia:Battle_Of_Washington	1.0
dbpedia:Beaufort_County,_North_Carolina	dbpedia:John_G._Foster	1.0

Using this model, we applied, as in the football scenario, our disambiguation process in a set of 50 military conflict texts, targeting the locations mentioned in them. The average reference ambiguity of this set was 5 in a total of 55 locations. Table 6 shows the achieved results which verify the ability of our framework to improve disambiguation effectiveness.

Table 6. Entity Disambiguation Evaluation Results in the Military Conflict Scenario

System/Approach	**Precision**	**Recall**	F_1 **Measure**
Proposed Approach	88%	83%	85%
DBPedia Spotlight	71%	69%	70%
AIDA	44%	40%	42%

5 Discussion

It should have been made clear from the previous sections that our framework is not independent of the content or domain of the input texts but rather adaptable to them. That's exactly its main differentiating feature as our purpose was not to build another generic disambiguation system but rather a reusable framework that can i) be relatively easily adapted to the particular characteristics of the domain and application scenario at hand and ii) exploit these characteristics in

order to increase the effectiveness of the disambiguation process. Our motivation for that was that, as the comparative evaluation of the previous section showed, the scenario adaptation capabilities of existing generic disambiguation systems can be inadequate in certain scenarios (like the ones described in this paper), thus limiting their applicability and effectiveness.

Of course, the usability and effectiveness of our approach is directly proportional to the content specificity of the texts to be disambiguated and the availability of a priori knowledge about their content. The greater these two parameters are, the more applicable is our approach and the more effective the disambiguation is expected to be. The opposite is true as the texts become more generic and the information we have out about them more scarce. A method that could a priori assess how suitable is our framework for a given scenario would be useful, but it falls outside the scope of this paper. Also, the framework's approach is not completely automatic as it requires some knowledge engineer or domain expert to manually define the scenario's disambiguation evidence mapping function. Nevertheless, this function is defined at the schema level thus making the number of required mappings for most scenarios rather small and manageable.

Finally, although we haven't formally evaluated the scalability of our approach, the fact that our framework is based on the constraining of the semantic data to be used makes us expect that it will perform faster than traditional approaches that use the whole amount of data. Furthermore, as the disambiguation evidence model may be constructed offline and stored in some index, the most probable bottleneck of the process will be the phase of determining the candidate entities for the extracted terms rather than the resolution process. Nevertheless, a more rigorous scalability study will have to be made as part of future work.

6 Conclusions and Future Work

In this paper we proposed a novel framework for optimizing named entity disambiguation in well-defined and adequately constrained scenarios through the customized selection and exploitation of semantic data. First we described how, given a priori knowledge about the domain(s) and expected content of the texts that are to be analyzed, one can use the semantic data and define an evidence model that determines which and to what extent semantic entities should be used as contextual evidence for the disambiguation task at hand. Then we described the process through which such a model can be actually used for this task. The overall framework was experimentally evaluated in two specific scenarios and the results verified its superiority over existing approaches that are designed to work in open domains and unconstrained scenarios.

Future work will focus on the further automation of the disambiguation evidence model construction by means of data mining and machine learning techniques. Moreover, an online tool to enable users to dynamically build such models out of existing semantic data and use them for disambiguation purposes, will be developed.

Acknowledgements

This work was supported by the Spanish project CENIT-2009-1026 BuscaMedia and by the European Commission under contract FP7- 248984 GLOCAL.

References

1. Auer, S., Bizer, C., Kobilarov, G., Lehmann J., Cyganiak, R., Ives, Z.G.: DBpedia: A Nucleus for a Web of Open Data. In Proceedings of the 6th International Semantic Web Conference, pages 722-735, 2007.
2. Fader, A., Soderland, S., Etzioni, O.: Scaling wikipedia-based named entity disambiguation to arbitrary web text. In Proceedings of the WikiAI 09 - IJCAI Workshop: User Contributed Knowledge and Artificial Intelligence: An Evolving Synergy, Pasadena, CA, USA, July 2009.
3. Ferragina, P., Scaiella, U.: TAGME: on-the-fly annotation of short text fragments (by wikipedia entities). In Proceedings of the 19th ACM international conference on Information and knowledge management, ACM, New York, SA, 1625-1628.
4. Gruhl, D., Nagarajan, M., Pieper, J., Robson, C., Sheth A.P.: Context and domain knowledge enhanced entity spotting in informal text. In Proceedings of the 8th International Semantic Web Conference, pages 260-276, 2009.
5. Hassell, J., Aleman-Meza, B., Arpinar, I.: Ontology-driven automatic entity disambiguation in unstructured text. In Proceedings of the 3rd European Semantic Web Conference, pages 44-57, Springer Berlin, Heidelberg, 2006.
6. Hoffart, J., Yosef, M.A., Bordino, I., Frstenau, H, Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G.: Robust disambiguation of named entities in text. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Stroudsburg, PA, USA, 782-792.
7. Kleb, J., Abecker, A.: Entity Reference Resolution via Spreading Activation on RDF-Graphs. In Proceedings of the 7th European Semantic Web Conference, pages 152-166, Springer Berlin, Heidelberg, 2006.
8. Mendes, P.N., Jakob, M., Garcia-Silva, A., Bizer, C.: DBpedia spotlight: shedding light on the web of documents. In Proceedings of the 7th International Conference on Semantic Systems, ACM, New York, USA, 1-8, 2011.
9. Miller, G., Charles, W.: Contextual correlates of semantic similarity. Language and Cognitive Processes, 6(1):128.
10. Pilz, A., Paass, G.: Named entity resolution using automatically extracted semantic information. Workshop on Knowledge Discovery, Data Mining, and Machine Learning, page 84-91, 2009.
11. Rizzo G., Troncy, R.: NERD: A Framework for Evaluating Named Entity Recognition Tools in the Web of Data. In 10th International Semantic Web Conference, Demo Session, pages 1-4, Bonn, Germany, 2011.
12. Rusu, D., Fortuna, B., Mladenic, D.: Automatically Annotating Text with Linked Open Data. In 4th Linked Data on the Web Workshop (LDOW 2011), 20th World Wide Web Conference, Hyderabad, India, 2011.
13. Suchanek, F.M., Kasneci, G., Weikum, G.: YAGO: A Core of Semantic Knowledge. In 16th World Wide Web Conference, 2007.