Private & Confidential, available to CeADAR Members only

CeADAR – Centre for Applied Data Analytics Research

Enterprise Ireland Data Analytics Technology Centre

OntoCore - State of the Art Review

Document Type:	Literature Review
Project Title:	CeADAR
DDN Theme:	1 – Intelligent Analytic Interfaces
Theme Leader:	Sarah Jane Delany (DIT)
DDN Sub-Theme:	Ease of Interaction
Authors:	Robert Ross, Eoghan O'Shea
Document Version:	1.0
Date of Delivery to ISG:	10th August, 2015
Number of pages:	12

Copyright © the authors. Confidential – not to be circulated without permission.

CeADAR is a research partnerhip comprising University College Dublin, University College Cork, and Dublin Institute of Technology.

http://www.ceadar.ie

Private & Confidential, available to CeADAR Members only

Contents

Private & Confidential, available to CeADAR Members only

Executive Summary

In this review, we discuss the state-of-the-art in technologies relating to the OntoCore project. At the heart of the OntoCore project is a proposal to analyze textual data and enhance that data through metadata augmentation based on the extraction of relevant information from Linked Open Data sources.

The review looks in depth at three specific technology areas which together will comprise the OntoCore Project.

- The first of these is Named Entity Recognition (NER). This is the process whereby important entities or things are identified in the input of raw unstructured data such that this data can be subsequently enhanced.
- The second area of interest will be Linked Open Data (LOD) which is an open access initiative which aims to give developers and end users access to vast amount of collated Linked Data, which can be used for a variety of applications. Enhancement of unstructured metadata through Linked Open Data is a significant research challenge, requiring the selection of the relevant Linked Open Data which can match given named entities.
- As such, our third area of interest is Automated Annotation of Text, whereby the most relevant data is found (from LOD or elsewhere) and used to annotate textual input.

We will also discuss in this review the possibility of ontology & taxonomy induction, that is, the automatic learning and construction of ontologies and taxonomies directly from text.

Finally, we will identify in this review some commercial systems that perform some of the same tasks as planned by OntoCore, but with certain limitations in each case.

Private & Confidential, available to CeADAR Members only

1 Introduction

OntoCore seeks to use named entity recognition to identify basic terminology (keywords) from unstructured or semi-structured content, to enrich this with Linked Open Data (LOD), and then relate this enriched terminology back to the original content as metadata.

2 Industry Benefit

Organizations are commonly overloaded with unstructured or semi-structured content which while often being sufficient for current business objectives and operation structures, can potentially be used in a more effective, automated manner which would increase yields and open up new product opportunities. Identifying the concepts held within this data, and the relationships between these concepts, is a crucial step in enabling these new opportunities. Domain ontologies provide the underlying terminological and relational infrastructure that is necessary for this goal, but are usually costly to create, requiring manual construction. A more automated system for domain ontology population has a very real benefit to organizations in leveraging data. This project will develop and evaluate such a framework for ontology extraction for use in varied applications.

3 Technology Description

The OntoCore demonstrator will be a pipeline tool for the automated analysis of content and the production of suggested ontological structures from this data. OntoCore will build on two key developments in the structured data community over the past ten years:

- Algorithms for ontology extraction from unstructured data – for example, the OntoLearn platform.
- Linked Open Data (LOD) repositories which provide a semantic bootstrapping.

The demonstrator will adopt the following workflow:

- Basic terminology extraction from unstructured or semi-structured content.
- Terminology enrichment with Linked Open Data content.
- Content enhancement with enriched terminology through metadata creation.

An enriched terminology is, in practice, an instance rich domain ontology and will be the primary output of the analysis engine. In addition to using this information directly to retroactively edit content as defined above, the enriched terminology will also be stored in, for example, RDF (Resource Description Framework) for future use and editing through an ontology editor such as Protégé[1]. The content will also be directly accessible for loading into 3^{rd} party reasoners and related applications for further use or development.

Achieving the above objective is dependent on a number of separate natural language processes, information extraction, and knowledge representation and reasoning technologies being combined efficiently. These technologies include:

[1] http://protege.stanford.edu/

Private & Confidential, available to CeADAR Members only

- **Named Entity Recognition –** the process of identifying meaningful instances in text;
- **Linked Open Data –** a large scale open access linked data representation;
- **Automatic Annotation Methods –** technologies dedicated to finding the most relevant data (from LOD and elsewhere) to annotate textual input.

We will review the state of the art in these key technological areas (Sections 3.1, 3.2, and 3.3). In Section 3.4 we also consider the related theme of Ontology Induction whereby an ontological domain model – consisting of axiomatic descriptions of classes and the relationships that can hold between them – is automatically or semi-automatically constructed. Finally, in Section 3.5, we will review a number of different commercial systems available in the semantic-processing field that allow entity extraction, tagging and metadata creation, but which have limitations with regard to OntoCore.

3.1 Named Entity Recognition

Named Entity Recognition (NER) is a key process in Information Extraction that takes text as input and produces a list of identified and partially classified proper names from that text. Entities here typically refer to People, Places, and Organizations, but many other types of domain specific and domain independent concepts might be the subject of identification. NER has a long research history (both in terms of academic research and practical tool development), and there are a number of overview research papers and chapters on the topic (See e.g., [17], [21], [1]).

A 'Named Entity' is a concrete thing that is named in a text. Unlike a part of speech *type* like a *noun* or *noun phrase*, a Named Entity is not simply a word but rather a semantic representation of the item which is assumed to be referred to in the text. Thus a number of different text tokens in a text input might refer to the same Named Entity. Consider for example the terms *Professor Jones*, *the Professor*, *Michael Jones*, and even the pronoun *him* might refer to the same Named Entity, i.e., the same person. In general Named Entity Recognition is useful in improving our understanding of text and is a precursor to important Information Extraction tasks like anaphora resolution. In our current context Named Entities are the most important items in a text and our goal is to attribute our metadata to these entities. By performing Named Entity Recognition we can ensure that all references to a particular Named Entity in the text are successfully annotated with the correct Linked Data without overpopulating the data model.

Named Entity Recognition can be challenging due principally to the many different terms that can be used to refer to the same entity in a text. A given entity will often have many different terms applied to it, and within these, small spelling or abbreviation variations can complicate matters. Lexical ambiguities such as homonyms and metonyms can cause considerable difficulty in NER. Homonyms are words that sound like each other but have different meanings, e.g., May the month and May the name, while metonyms are words that refer to a more abstract entity than in its standard form, i.e., 'school' may refer to either the building or an educational institute. On a more practical level Named Entity Recognition is also complicated by domain factors. While places and people have been widely studied and have

Private & Confidential, available to CeADAR Members only

general purpose tools available, highly specified domains, as commonly found in professional texts, can contain references to specific business processes, chemicals, or even laws. Yet another complication is seen in cross-lingual work. Named Entity Recognition tools are built for specific languages. Porting these tools to new languages often requires considerable effort - particularly where those languages have complex morphological systems.

Fortunately a number of mature Named Entity Recognition tools are available which can be used to process unstructured textual data. The two most notable of these are the Stanford Named Entity Recognizer [10], and the Named Entity Recognition component of the NLTK framework [3]. Here our discussion relates to the Stanford NER tool. The Stanford NER tool is based on a linear chain Conditional Random Field framework [24]. It comes pre-built with models for Person, Place, and Organization classes in English, and options for defining additional features and class types. While the default implementation of the Stanford NER tool includes good performance over People, Places, and Organizations, the support of domain specific terms will require the retraining of the classifier with suitably annotated data.

3.2 Linked Open Data

Linked Open Data (LOD) is an initiative that aims to provide a vast amount of structured information that is publicly available and interlinked via means of the world wide web. While it is in principle possible to openly publish many standalone databases, the key benefit of LOD is the connection of database instances by means of well-defined concepts and relationships which allow otherwise unrelated data sources to be coherently related. Linked Open Data has emerged out of streams of work in both Open Data initiatives and the Semantic Web and is now commonly viewed as best practice for openly publishing structured information that is suitable for use in the development of intelligent applications [4, 28].

At a low-level a collection of LOD databases are essentially massive collections of interlinked assertions. For example the popular DBpedia collection consists of some 431 Million triples as of the time of writing. The authoring of these assertions, the semantic mechanisms by which these assertions are related, and the mechanisms by which we can reason efficiently over these large collections of facts are the key technological challenges in the LOD domain. Here we will briefly mention a number of the key technologies which underpin LOD authoring and use (references are given to more complete sources of information that can be consulted for further details):

- **Ontology -** An ontology is a formal specification of a domain of concepts expressed in some representation language. This form of ontology is typically refereed to as ontology with a small 'o'. Ontology with a big 'O' refers to the more general domain of philosophy which attempts to provide a meta-physical description of the world or other domain. Ontologies are usually the high-level formal descriptions which relate individual set resources within LOD.
- **Taxonomy -** A taxonomy can be thought of as a simplified ontology in which only concepts and the hierarchical relationships between those concepts are considered. Thus all relationships within a taxonomy are 'isA' relationships, and no other relationships between concepts in the taxonomy can be encoded. Taxonomies are thus considered easier to model and reason over than full ontologies.

Private & Confidential, available to CeADAR Members only

- **RDF -** Resource Description Framework (RDF) is a data format for the storage of data suitable for collection within a LOD collection [12, 19]. The language (or rather collection of specifications) is designed to allow the easy authoring of conceptual descriptions of entities that are to be studied. There is a large collection of specifications included in the RDF body of knowledge that provide alternative serialization formats. The most commonly occurring variant of these serializations is an XML based format.
- **Triples -** From a relational database perspective, RDF can be conceived of as a three column database wherein each row captures an assertion between a subject and an object. The columns thus capture the assertion type, the subject, and the object respectively. Individual assertion rows are typically referred to as triples. Triples are frequently used as a measurement of ontology size, and within this context are often used as a measurement of LOD collection size.
- **Description Logic -** Description Logic is a form of logic that is weaker than full First Order Logic but is expressive enough that a considerable amount of information can be asserted and reasoned over [16]. Description logic is powerful enough for most common applications on the semantic web including the linking of data types in any LOD collection.
- **OWL -** OWL refers to the Web Ontology Language. [27, 14]. OWL is a representation language for ontologies suitable for use in LOD. There are a number of variants of OWL available that are related to variants of Description Logic along with different mechanisms of serializing the language.
- **Reasoner -** and specifically a Description Logic or OWL reasoner is a tool that can read in ontologies and other semantic specifications from RDF or OWL and perform reasoning tasks on the asserted data. Reasoning types include checking subclass relationships, checking if an object is an instance of a specific type, or checking if a particular relationship holds between two instances. There are a number of powerful reasoners available for use in practical tasks[2]. These include Pellet [23] and Fact++ [25].
- **Ontology Editor -** An ontology editor is a user tool that facilitates the authoring of general and domain specific ontologies. The editor can typically be configured for a specific ontology representation language type, e.g., OWL. The most popular Ontology Editor is currently the Protégé tool [11].
- **SPARQL -** A language for querying ontology instances contained in reasoners. SPARQL is based broadly on SQL but has constructs specifically required for the querying of ontologies.

Linked Open Data technology builds on three additional technology classes apart from the data storage and reasoning mechanisms just outlined. These technology classes are: URIs and HTTP for data instance naming and communication processing, respectively, and the Linked Data Platform which is a specific instance of specifications which define a RESTFul HTTP[3] service for Linked Data Use. There are a number of Linked Open Data datasets which pull together many individual linked data repositories. The most popular of these LOD datasets is DBpedia [2] and Freebase [5]. Taking DBPedia as an example dataset, its LOD database was built through the automated analysis of Wikipedia articles to construct

[2] http://owl.cs.manchester.ac.uk/tools/list-of-reasoners/

[3] `https://en.wikipedia.org/wiki/Representational_state_transfer`

Private & Confidential, available to CeADAR Members only

an open accessible general information database. DBpedia is based on RDF technology and describes over 4.6 million entities through 3 billion RDF triples. DBpedia also links to external LOD databases and can be used as a standalone knowledge base or as an online service through a SPARQL endpoint.

3.3 Annotating Text with Structured Information

In this section we will review works that have addressed the question of automated annotation of text (or web pages) through reference to pre-existing semantic resources. In this area there have been a number of efforts. De Virgilio [9], for example, investigated the use of automated methods to annotate web pages with semantic information - although in their case the data types queried were not specifically LOD resources.

More relevant to our work, Rusu et al. specifically address the question of text annotation via Linked Open Data [22]. They investigated methods which can be used to select the most relevant data from the LOD datasets for annotation of textual input. The two methods investigated were: (a) a variation of Pagerank which created a graph structure from LOD resources so that the most relevant sources of information could be identified from the LOD input; and (b) a text similarity based measure employing a comparison of input sentence contexts and the comments / labels often attached to concepts in LOD resources. Rusu et al. found that Pagerank generally performed best on data identification, and hence provides a solid basis for any LOD based annotation task.

3.4 Ontology & Taxonomy Induction

LOD datasets such as DBpedia aggregate data that have often been asserted in a formal or semi-formal format elsewhere. For example Wikipedia information boxes (partly structured) are a valuable tool in the construction of LOD datasets. The merging of LOD datasets often assumes that explicit domain ontologies exist, but these are by no means easy to author. Historically ontologies have been created manually using either pen and paper or more recently ontology editors like Protégé. For the use of ontologies and linked data applications to become more widespread, it is important that tools be leveraged to assist in the automated extraction of ontologies from unstructured textual information.

The learning or induction of Ontologies and Taxonomies from text has become an active area of research and development in recent years. In 2015 for example we have seen the first instance of a shared task on the automated construction of taxonomies [6]. Taking taxonomy construction as an example, the general process consists of: (a) identification of terms / named entities in the input text source; (b) induction of isA relationships between terms (e.g. a Dog isA Mammal, etc.); and (c) induction of a well formed hierarchy from the individually induced relationships. Of the three broad steps (a) is the well understood NER problem introduced earlier; (b) is a more complex process where various methods ranging from stem analysis, string parsing, and consultation with other forms of information, can be used to induce and weight individual assertions. The final step, taxonomy induction, is generally considered the most difficult of all steps as the reconciling of contradictory and incomplete information amongst individual rules is a non-trivial problem.

Private & Confidential, available to CeADAR Members only

In the past 10 years there have been a number of individual pieces of work that have looked at the creation of specific ontology instances from text (see e.g., [8], [7], [15], [20], [18]. One of the more prominent examples of work in this area has been *OntoLearn* and its more recent successor *OntoLearn Reloaded* [26]. Unlike many other examples of previous work, OntoLearn Reloaded is a complete Ontology Induction method which learns concepts and relations from text through the extraction of terms, definitions, and hypernyms (Note that a hypernym is a word with a broad meaning constituting a category into which words with more specific meanings fall, e.g., vehicle is a hypernym of car or colour is a hypernym of orange).

While algorithms such as OntoLearn Reloaded significantly simplify the task of Ontology construction, an unsupervised induction can be considerably improved upon through human intervention. With this in mind, [13] have recently examined the inclusion of active learning principles in the knowledge base construction process. The basic principle of active learning is that a machine learning or data mining based system should consult with a human expert or adviser in certain boundary decisions. Thus the resultant knowledge base is not human built, but rather has human input in cases which may have otherwise been problematic.

3.5 Commercial Systems

Three commercial systems that each perform some elements of what OntoCore seeks to achieve are described in this section.

The first of these is the Semaphore system by Smartlogic[4]. This system allows users to create ontology/taxonomy models using an editor. Using these ontologies, Semaphore can extract entity relationships within a document (using rule-based classification), use these to tag the document content and produce usable metadata for filtering, etc. However, we note that unlike OntoCore's planned automated system based on LOD, this system has the drawback of requiring pre-created ontologies in order to produce the metadata.

The second system we have identified is the contentClassifier technology from conceptSearching[5] which can identify multi-word concepts in unstructured text, allowing the creation of semantic metadata which can then be classified to organizationally defined taxonomies. The system claims to perform automatic intelligent metadata generation as content is created or ingested, together with an automated classification of content to one or more nodes in one or more taxonomies. Unlike the planned OntoCore system, contentClassifier does not make use of LOD to enrich keywords, etc, relying instead on what they call "compound term processing" to identify key (multi-word) concepts in unstructured data, using this to produce the semantic metadata, before classifying it based on pre-existing taxonomies. This system therefore likely lacks the richness of metadata possible from the LOD enriched terminology planned for the OntoCore system.

The third system is the PoolParty Semantic Suite[6], which allows the creation of taxonomies and for these to be taxonomies to be manually extended in an editor using chosen linked

[4] http://www.smartlogic.com/what-we-do/products-overview

[5] http://www.conceptsearching.com/wp/products/

[6] https://www.poolparty.biz/

Private & Confidential, available to CeADAR Members only

open data (DBpedia), with additional text-mining (entity extraction) and content annotation tools (tagging) available. While PoolParty Semantic Suite makes use of LOD, it is clear that this system is unlike OntoCore, in that OntoCore seeks to automatically annotate content directly from LOD, not use it solely to manually extend a taxonomy.

4 Conclusions

4.1 Limitations with the State of the Art

We remind the reader that OntoCore seeks to use named entity recognition to identify basic terminology (keywords) from unstructured or semi-structured content, to enrich this with LOD and then relate this enriched terminology to the original content as metadata.

While each of the three commercial systems discussed in Sect. 3.5 perform some of these tasks, *none of them* is an exact match for what OntoCore seeks to achieve using LOD.

We can conclude therefore that the planned OntoCore system is not replicated in industry and would make a valid platform technology.

4.2 Conclusion

In this report, we have reviewed the state-of-the-art in the technologies relating to the OntoCore project. We have focused on three principal technologies: Named Entity Recognition, Linked Open Data and Automatic Annotation of Text. We have also considered the related theme of Ontology Induction whereby an ontological domain model is automatically or semi-automatically constructed. Finally, we have reviewed a number of commercial systems in the general area of semantic-processing, and we conclude that none of these perform the same task as planned for OntoCore, i.e., the automatic annotation of text using LOD.

References

[1] Samet Atdag and Vincent Labatut. A comparison of named entity recognition tools applied to biographical texts. In *Systems and Computer Science (ICSCS), 2013 2nd International Conference on*, pages 228–233. IEEE, 2013.

[2] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. *Dbpedia: A nucleus for a web of open data*. Springer, 2007.

[3] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python*. " O'Reilly Media, Inc.", 2009.

[4] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data-the story so far. *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, pages 205–227, 2009.

[5] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.

[6] Georgeta Bordea, Paul Buitelaar, Stefano Faralli, and Roberto Navigli. Semeval-2015 task 17: Taxonomy extraction evaluation (texeval). *Science*, 452(465):429–441, 2015.

[7] Philipp Cimiano, Andreas Hotho, and Steffen Staab. Learning concept hierarchies from text corpora using formal concept analysis. *J. Artif. Intell. Res.(JAIR)*, 24:305–339, 2005.

Private & Confidential, available to CeADAR Members only

[8] Philipp Cimiano and Johanna Völker. Text2onto. In *Natural language processing and information systems*, pages 227–238. Springer, 2005.

[9] Roberto De Virgilio. Rdfa based annotation of web pages through keyphrases extraction. In *On the Move to Meaningful Internet Systems: OTM 2011*, pages 644–661. Springer, 2011.

[10] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics, 2005.

[11] John H Gennari, Mark A Musen, Ray W Fergerson, William E Grosso, Monica Crubézy, Henrik Eriksson, Natalya F Noy, and Samson W Tu. The evolution of protégé: an environment for knowledge-based systems development. *International Journal of Human-computer studies*, 58(1):89–123, 2003.

[12] Richard DF Harris and Elias Tzavalis. Inference for unit roots in dynamic panels where the time dimension is fixed. *Journal of econometrics*, 91(2):201–226, 1999.

[13] Hiroshi Kajino, Akihiro Kishimoto, Adi Botea, Elizabeth Daly, and Spyros Kotoulas. Active learning for multi-relational data construction. In *Proceedings of the 24th International Conference on World Wide Web*, pages 560–569. International World Wide Web Conferences Steering Committee, 2015.

[14] Deborah L McGuinness, Frank Van Harmelen, et al. Owl web ontology language overview. *W3C recommendation*, 10(10):2004, 2004.

[15] Olena Medelyan, Ian H Witten, Anna Divoli, and Jeen Broekstra. Automatic construction of lexicons, taxonomies, ontologies, and other knowledge structures. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(4):257–279, 2013.

[16] Boris Motik and Ulrike Sattler. A comparison of reasoning techniques for querying large description logic aboxes. In *Logic for programming, artificial intelligence, and reasoning*, pages 227–241. Springer, 2006.

[17] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[18] Roberto Navigli, Paola Velardi, and Stefano Faralli. A graph-based algorithm for inducing lexical taxonomies from scratch. In *IJCAI*, pages 1872–1877, 2011.

[19] Jeff Z Pan. Resource description framework. In *Handbook on Ontologies*, pages 71–90. Springer, 2009.

[20] Hoifung Poon and Pedro Domingos. Unsupervised ontology induction from text. In *Proceedings of the 48th annual meeting of the Association for Computational Linguistics*, pages 296–305. Association for Computational Linguistics, 2010.

[21] Lev Ratinov and Dan Roth. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155. Association for Computational Linguistics, 2009.

[22] Delia Rusu, Blaz Fortuna, and Dunja Mladenic. Automatically annotating text with linked open data. In *LDOW*, 2011.

[23] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical owl-dl reasoner. *Web Semantics: science, services and agents on the World Wide Web*, 5(2):51–53, 2007.

[24] Charles Sutton and Andrew McCallum. An introduction to conditional random fields. *Foundations and Trends in Machine Learning*, 4(4):267–373, 2012.

Private & Confidential, available to CeADAR Members only

[25] Dmitry Tsarkov and Ian Horrocks. Fact++ description logic reasoner: System description. In *Automated reasoning*, pages 292–297. Springer, 2006.

[26] Paola Velardi, Stefano Faralli, and Roberto Navigli. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707, 2013.

[27] Xiao Hang Wang, Da Qing Zhang, Tao Gu, and Hung Keng Pung. Ontology based context modeling and reasoning using owl. In *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on*, pages 18–22. Ieee, 2004.

[28] Liyang Yu. Linked open data. In *A DeveloperâĂŹs Guide to the Semantic Web*, pages 409–466. Springer, 2011.