FRED: from natural language text to RDF and OWL in one click

Francesco Draicchio[1], Aldo Gangemi[1],
Valentina Presutti[1,2], and Andrea Giovanni Nuzzolese[1,2]

[1] STLab-ISTC Consiglio Nazionale delle Ricerche, Rome, Italy.
[2] Dipartimento di Scienze dell'Informazione, Università di Bologna, Italy.

Abstract. FRED is an online tool for converting text into internally well-connected and quality linked-data-ready ontologies in web-service-acceptable time. It implements a novel approach for ontology design from natural language sentences. In this paper we present a demonstration of such tool combining Discourse Representation Theory (DRT), linguistic frame semantics, and Ontology Design Patterns (ODP). The tool is based on Boxer which implements a DRT-compliant deep parser. The logical output of Boxer enriched with semantic data from Verbnet or Framenet frames is transformed into RDF/OWL by means of a mapping model and a set of heuristics following ODP best-practice [5] of OWL ontologies and RDF data design.

1 Introduction

The problem of knowledge extraction from text is still only partly solved, this is particularly true if we consider it as a means for populating the Semantic Web. Being able to automatically and fastly produce quality linked data and ontologies from an accurate and reasonably complete analysis of natural language text would be a breakthrough: it would enable the development of applications that automatically produce machine-readable information from Web content as soon as it is edited and published by generic Web users.

Existing Ontology Learning and Population (OL&P) approaches can be used as drafting ontology engineering tools, but they show some limitations when used to produce linked data on the Web:

- they usually need a training phase, which can take a long time;
- their output form needs further, non-trivial elaboration to be put in a logical form;
- they only partially exploit OWL expressivity, since they typically focus on specific aspects e.g. taxonomy creation, disjointness axioms, etc.;
- in most cases, they lack implementation of ontology design good practices;
- linking to existing linked data vocabularies and datasets is usually left as a separate task or not considered at all.

In other words, existing tools focus mainly on helping users identifying key elements for ontology drafting, assuming that they would transform and substantially refine and enrich the output. Our approach instead focuses on producing ontologies and linked data ready for the Web.

We present a novel approach implemented in an online tool named FRED, which performs deep parsing of natural language, extracts complex relations based on Discourse Representation Theory (DRT), and produces linked data graphs.

An important aspect of OL&P is the design quality of the resulting ontology: this is related to representing the results in a logical form such as OWL by ensuring that some best-practices are followed.

Based on this consideration, we summarize a set of requirements that FRED follows in order to enable robust OL&P:

- ability to capture accurate semantic structures;
- representing complex relations;
- exploiting general purpose resources;
- no need of large-size domain-specific text corpora and training sessions;
- minimal time of computation;
- ability to map natural language to RDF/OWL representations.

2 Related work

OL&P is concerned with the (semi-)automatic generation of ontologies from textual data (cf. [2]). Typical approaches to OL&P are implemented on top of Natural Language Processing (NLP) techniques, mostly machine learning methods, hence they require large corpora, sometimes manually annotated, in order to induce a set of probabilistic rules. Such rules are defined through a training phase that can take long time. OL&P systems are usually focused on either ontology learning (OL) for TBox production, or ontology population (OP) for ABox production. Examples of OL systems include [6], which describes an ontology-learning framework that extends typical ontology engineering environments by using semiautomatic ontology-construction tools, and Text2Onto [3], which generates a class taxonomy and additional axioms from textual documents. Examples of OP systems include: [9], which presents a GATE plugin that supports the development of OP systems and allows to populate ontologies with domain knowledge as well as NLP-based knowledge; [8] describes a weakly supervised approach that populates an ontology of locations and persons with named entities; [7] introduces a sub-task of OP restricted to textual mentions, and describes challenging aspects related to named entities. The method and tool (FRED) that we present in this paper differs from most existing approaches, because it does not rely on machine learning methods.

3 FRED at work

In this section we present an overview of the pipeline implemented by FRED and an example of FRED execution. FRED reuses Boxer[1], a linguistic tool

based on DRT that generates formal semantic representations of text based on event semantics. Identifying the correct event in a sentence means identifying the frame behind it, which in turns leads to improve the design quality of the resulting ontology, as shown in [4]. This is motivated by the fact that frames match to ontology design patterns.

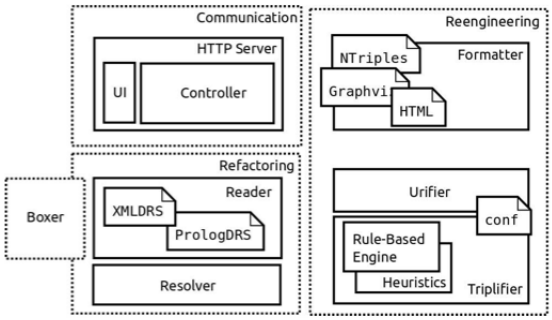

Fig. 1. Block architecture and workflow

Figure 1 depicts FRED architecture that is composed of four main components, implementing the following computational steps:

- capture user input text (e.g., via an HTML interface) and pass it to Boxer;
- retrieve Boxer XML output and transform it into a convenient data structure;
- run a collection of ad-hoc transformation rules and heuristics for producing OWL/RDF triples;
- transform and return triples into human as well as machine readable format.

Although Boxer produces a logical form from natural language - syntactically, lexically, and semantically - this logical form differ from RDF or OWL, and the heuristics that it implements for interpreting a natural language and transforming it to a DRT-based structure can be sometimes awkward when directly translated to ontologies for the Semantic Web. For this reason, FRED implements a set of rules for transforming Boxer output to OWL/RDF ontologies.

DRT construct	**Boxer syntax**	**FOL construct**	**OWL construct**
Predicate	pred(x)	Unary predicate ϕ	`rdf:type`
Relation	rel-name(x,y)	Binary relation	`owl:ObjectProperty`
Eq Rel	eq(x,y)	Identity	`owl:sameAs`
Named Entity	named(<var>, <name>, <type>)	Unary predicate ϕ	`owl:NamedIndividual`
Discourse Referent	(<var>)	Quantified Variable	(generated) `owl:NamedIndividual`
DRS	<drs> with event E	Proposition P with predicate ϕ_E	RDF graph G_P with class `E`
Negated DRS	not(<drs>)	Negated Proposition $\neg P$	G_P with `NotE owl:disjointWith E`

Fig. 2. A sample subset of quality ontology production heuristics

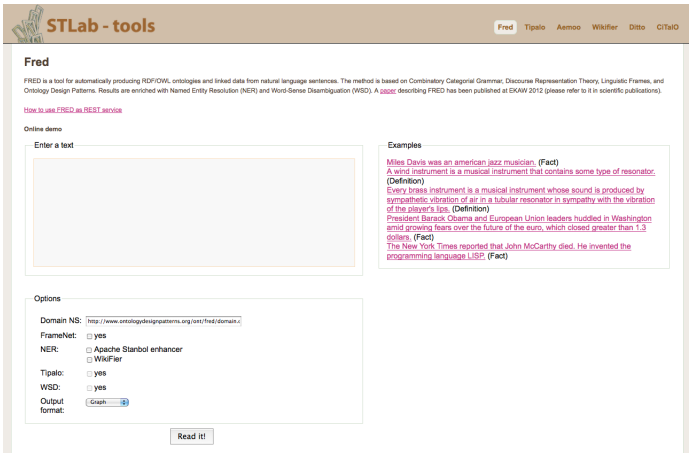

Fig. 3. Screenshot of FRED on-line demo.

We distinguish two set of rules: (i) translation rules, which define global transformations from DRS constructs to OWL constructs (figure 2); and (ii) heuristics rules, which deal with adapting the results to the design needs of a Semantic Web ontology.

Demo. FRED on-line demo[3]. (see Figure 3) provides a simple user interface that accepts a natural language text and returns a RDF/OWL graph either as a graphics or as a RDF serialization. A list of sample sentences are provided (on the right side) as a starting point for testing the tool. For demonstration purpose let us consider the following sentence "*The New York Times reported that John McCarthy died. He invented the language LISP.*". By clicking on the "Read me!" button, a user would obtain the graph shown in Figure 4.

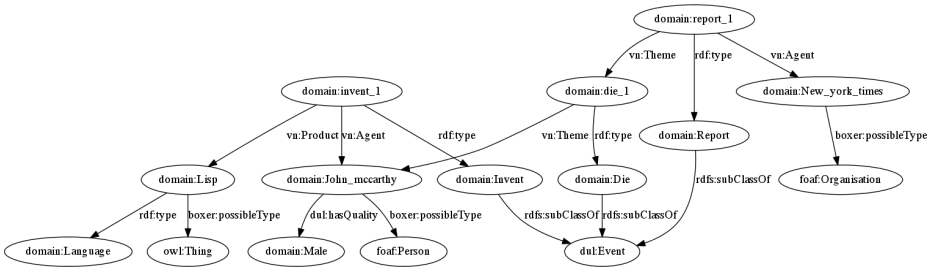

Fig. 4. Resulting graph for the sentence "The New York Times reported that John McCarthy died. He invented the language LISP."

[3] FRED on-line demo `http://wit.istc.cnr.it/stlab-tools/fred`. FRED is exposed as HTTP REST API

Two events (typed as DOLCE events) are correctly recognized: `Report`[4] and `Die`. An individual `John_mccarthy` of type `foaf:Person` is created, which has the role `vn:Theme`[5] in the `Die` event, which in turn is the theme of the `Report` event performed by (role `vn:Agent`) the new created individual `New_york_times` of type `foaf:Organisation`. Additionally, FRED creates an individual for the language `LISP` and is able to recognize a third event `Invent` that involves the individual `John_mccarthy`, as agent this time, and `LISP`, as theme. This last situation is correctly represented by performing co-reference resolution.

4 Conclusion and future work

We have presented FRED, a novel implemented method that transforms natural language texts to OWL/RDF according to a frame-based design approach. Currently we are working at a rigorous evaluation of the resulting ontology and linked data.

References

1. J. Bos. Wide-Coverage Semantic Analysis with Boxer. In J. Bos and R. Delmonte, editors, *Semantics in Text Processing. STEP 2008 Conference Proceedings*, volume 1 of *Research in Computational Semantics*, pages 277–286. College Publications, 2008.
2. P. Cimiano. *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Springer, 2006.
3. P. Cimiano and J. Vlker. Text2onto - a framework for ontology learning and data-driven change discovery, 2005.
4. B. Coppola, A. Gangemi, A. M. Gliozzo, D. Picca, and V. Presutti. Frame detection over the semantic web. In L. Aroyo, P. Traverso, F. Ciravegna, P. Cimiano, T. Heath, E. Hyvönen, R. Mizoguchi, E. Oren, M. Sabou, and E. P. B. Simperl, editors, *ESWC*, volume 5554 of *Lecture Notes in Computer Science*, pages 126–142. Springer, 2009.
5. A. Gangemi and V. Presutti. Ontology Design Patterns. In S. Staab and R. Studer, editors, *Handbook on Ontologies, 2nd Edition*. Springer Verlag, 2009.
6. A. Maedche and S. Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, 16:pp. 72–79, March-April 2001.
7. B. Magnini, E. Pianta, O. Popescu, and M. Speranza. Ontology population from textual mentions: Task definition and benchmark. In *Proceedings of the OLP2 workshop on Ontology Population and Learning, Sidney, Australia*, 2006.
8. H. Tanev and B. Magnini. Weakly supervised approaches for ontology population. In *Proceedings of the 2008 conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge*, pages 129–143, Amsterdam, The Netherlands, The Netherlands, 2008. IOS Press.
9. R. Witte, N. Khamis, and J. Rilling. Flexible ontology population from text: The owlexporter. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *LREC*. European Language Resources Association, 2010.

[4] We omit the prefix `domain:` as it is the local namespace and can be customized according to users needs.

[5] prefix `vn:` refers to VerbNet