A Machine Reader for the Semantic Web

Aldo Gangemi[1][2], Francesco Draicchio[1], Valentina Presutti[1]
, Andrea Giovanni Nuzzolese[1][3], and Diego Reforgiato[1]

[1] STLab-ISTC Consiglio Nazionale delle Ricerche, Rome, Italy.
[2] LIPN, Université Paris13-CNRS-SorbonneCité, France
[3] Dipartimento di Scienze dell'Informazione, Università di Bologna, Italy.

Abstract. FRED is a machine reading 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, combining Discourse Representation Theory (DRT), linguistic frame semantics, and Ontology Design Patterns (ODP). The current version of the tool includes Earmark-based markup, and enrichment with word sense disambiguation (WSD) and named entity resolution (NER) off-the-shelf components.

1 Introduction

The problem of knowledge extraction (KE) from text is still insufficiently addressed from a semantic web (SW) perspective. Being able to automatically produce quality linked data and ontologies from natural language text would be a breakthrough as 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. A rather detailed landscape analysis of the currently available tools for KE, and their exploitation for SW basic tasks is presented in [4]: it shows the substantial lacking of tools for creating RDF graphs that are connected enough to perform application tasks such as event extraction, fact detection, story mining, etc.

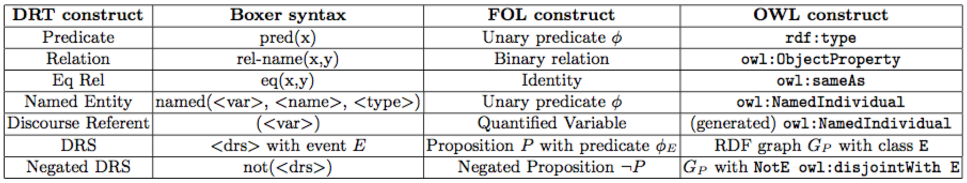
FRED[4] [5] is an exception, since it is intended to produce semantic data and ontologies with a quality closer to what is expected at least from average linked datasets and vocabularies: FRED candidates as a deep version of a machine reader [3] for the Semantic Web. The following requirements have inspired the design of FRED: (i) ability to capture accurate semantic structures (i.e. compliant to formal semantics); (ii) representing complex relations (i.e. n-ary, multigrade relations); (iii) exploitation of sophisticated lexical resources (e.g. VerbNet, FrameNet); (iv) no need of large-size domain-specific text corpora and training sessions (i.e. we address open information extraction); (v) minimal time of computation; (vi) ability to map natural language to RDF/OWL representations; (vii) ability to link the extracted knowledge to both lexical linked data and linked datasets (for maximal interoperability).

[4] http://wit.istc.cnr.it/stlab-tools/fred

Related works and comparison to other tools for knowledge extraction are detailed in [5], and [4], where FRED seems to outperform (though on a limited test) the other tools in sophisticated tasks such as relation and factoid extraction, frame detection, and taxonomy induction.

2 FRED at work

In this section we present an overview of the system and a scenario that shows the output resulting from FRED. [2] shows that detecting the most appropriate frames from the input text leads to improve the design quality of the resulting ontology because frames can be directly mapped to an important variety of ontology design patterns based on *n-ary* relations. On the above consideration, FRED makes use of Boxer [1], a deep semantic parser based on categorial grammar and Discourse Representation Theory (DRT), which generates formal semantic representation of text through an event (neo-Davidsonian) semantics.



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. 1. A sample subset of quality ontology production heuristics.

Boxer frame-based approach supports FRED in automatically design an ontology by following good modeling practices based on ontology patterns. However, Boxer tranforms natural language to a logical form compliant with DRT (substantially a variety of first-order logic) which differs a lot 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 SW, because it obeys pure FOL-oriented design style. For this reason, Boxer DRT-based output is transformed by FRED to OWL/RDF ontologies by means of a set of heuristics, some of them are showed in Figure 1. Figure 2 depicts the main components of FRED: *Communication*: exposes APIs for querying the system; *Refactoring*: transforms Boxer output[5] into a convenient data structure to be passed to the *Reenginering* component; *Reengineering*: applies a collection of ad-hoc mapping rules and heuristics for producing logically consistent OWL/RDF like triples.

In addition, FRED architecture is open to be easily integrated with other components that exploit the text span markup specification supported its current version, i.e. Earmark [6]. This solution, which is similar to architectures such as

[5] Boxer is an external component.

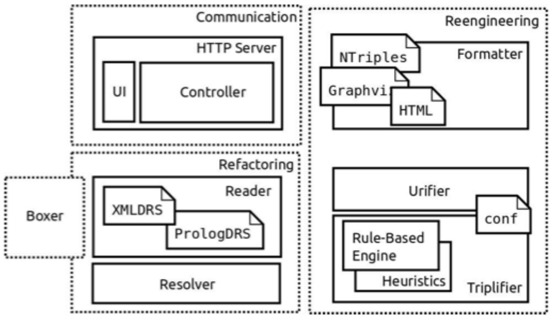


Fig. 2. Block architecture and workflow

NIF and NERD, makes it trivial to augment FRED graphs with off-the-shelf components for e.g. NER, WSD, etc. Our demo, available online[6], allows a user to enter any text (or select one from the list of examples provided), and by simply clicking the *Read It!* button, to receive a RDF/OWL representation of it[7]. Some features can be customized, e.g., type of output, NER or WSD activation, tense-relation between events activation, etc. By default, the output is in the form of Graphviz-like graphs (showing only a core subset of triples), to allow human users to quickly check the OWL/RDF representation.

FRED output consists in RDF triples including either fixed properties: `rdf:type, rdfs:subClassOf, owl:sameAs, dul:associateWith, owl:equivalentTo`, Earmark properties, thematic roles for extracted events, etc., or customized properties produced by the automatic (machine) reading of the sentence.

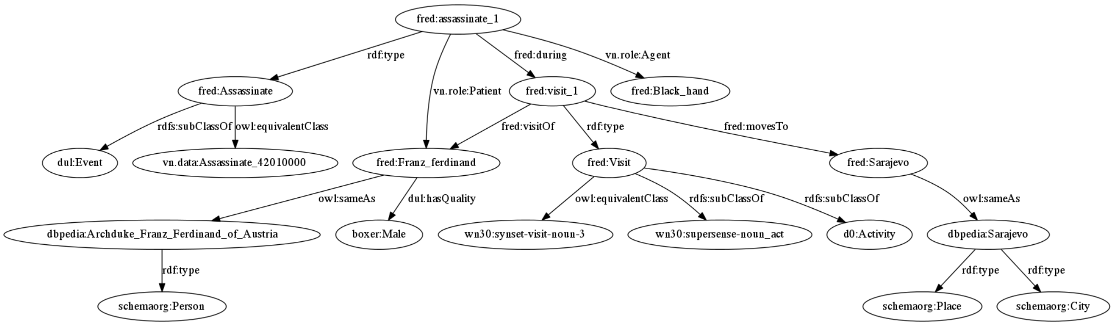


Fig. 3. FRED output for the sentence *“The Black Hand assassinated Franz Ferdinand during his visit to Sarajevo.”*

[6] http://wit.istc.cnr.it/stlab-tools/fred

[7] A graphical output is provided for human users

As an example, consider the sentence *"The Black Hand assassinated Franz Ferdinand during his visit to Sarajevo."* Figure 3 shows FRED output for this sentence: an instance of `dul:Event`, `fred:assassinate_1`, is used to represent the assassination of Franz Ferdinand. It is typed as `fred:Assassinate`, which is disambiguated by the VerbNet frame `vn.data:Assassinate_42010000`. Such an event involves the individual `fred:Black_hand` as agent, and the individual `fred:Franz_ferdinand` (who is recognized and resolved as the same individual as `dbpedia:Archduke_Franz_Ferdinand_of_Austria`) as patient. Furthermore, the RDF graph expresses that such an event happened `fred:during` `fred:visit_1`. FRED heuristically assigns a type `fred:Visit` to `fred:visit_1`; such a type is disambiguated by means of alignments to WordNet, which in turn is aligned to other ontologies, so that FRED can infer e.g., that `fred:Visit` is a `d0:Activity`. The location of the visit is also identified and correctly resolved to `dbpedia:Sarajevo`. Notice that FRED assigns types to all identified individuals either by using classes from existing ontologies e.g., DOLCE, Schema.org, etc., when the entity can be resolved e.g., as a DBpedia entity, or by creating new classes based on the terms used in the input text and disambiguating them on WordNet.

FRED also supports more sophisticated constructs, e.g. propositional referents, called *situations*, full-fledged negation on events or situations, and basic modalities over events.

3 Conclusion

We have presented FRED, a machine reader for the Semantic Web, which automatically extracts rich and connected knowledge from text, represents it as OWL/RDF, and links it to other resources: VerbNet, FrameNet, WordNet, DBpedia, foundational ontologies, etc. The current research is on mainly evaluating it on vertical tasks, and extending its internal components for multilinguality.

References

1. Johan Bos. Wide-Coverage Semantic Analysis with Boxer. In Johan Bos and Rodolfo Delmonte, editors, *Semantics in Text Processing*, pages 277–286. College Publications, 2008.
2. Bonaventura Coppola, Aldo Gangemi, Alfio Massimiliano Gliozzo, Davide Picca, and Valentina Presutti. Frame detection over the semantic web. In Lora Aroyo et al., editor, *ESWC*, volume 5554 of *LNCS*, pages 126–142. Springer, 2009.
3. Oren Etzioni, Michele Banko, and Michael Cafarella. Machine reading. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI)*, 2006.
4. Aldo Gangemi. A comparison of knowledge extraction tools for the semantic web. In *Proceedings of ESWC2013*. Springer, 2013.
5. Valentina Presutti, Francesco Draicchio, and Aldo Gangemi. Knowledge extraction based on discourse representation theory and linguistic frames. In *EKAW: Knowledge Engineering and Knowledge Management that matters*. Springer, 2012.
6. Peroni S., Gangemi A., and Vitali F. Dealing with markup semantics. In *Proceedings of the 7th International Conference on Semantic Systems, Graz, Austria (i-Semantics2011)*. ACM, 2011.