

Using FRED for Named Entity Resolution, Linking and Typing for Knowledge Base population

Sergio Consoli¹ and Diego Reforgiato Recupero¹

STLab-ISTC Consiglio Nazionale delle Ricerche, Catania, Italy

Abstract. FRED is a machine reader for extracting RDF graphs that are linked to LOD and compliant to Semantic Web and Linked Data patterns. We describe the capabilities of FRED as a semantic middleware for semantic web applications. In particular, we will show (i) how FRED recognizes and resolves named entities, (ii) how it links them to existing knowledge base, and (iii) how it gives them a type. Given a sentence in any language, it provides different semantic functionalities (frame detection, topic extraction, named entity recognition, resolution and coreference, terminology extraction, sense tagging and disambiguation, taxonomy induction, semantic role labeling, type induction) by means of a versatile user-interface, which can be recalled as REST Web service. The system can be freely used at <http://wit.istc.cnr.it/stlab-tools/fred>.

1 Introduction

The vision of the Semantic Web (SW) is to populate the Web with machine understandable data so that intelligent agents are able to automatically interpret its content, just like humans do by inspecting Web content, and assist users in performing a significant number of tasks, relieving them of cognitive overload. The Linked Data movement [1] kicked-off the vision by realising a key bootstrap in publishing machine understandable information mainly taken from structured data (typically databases) or semi-structured data (e.g. Wikipedia infoboxes). However, most of the Web content consists of natural language text, hence a main challenge is to extract as much relevant knowledge as possible from this content, and publish them in the form of Semantic Web triples.

In this paper we employ *FRED* [16], a machine reader for the Semantic Web, to address the first two tasks of the Open Knowledge Extraction (OKE) Challenge at the European Semantic Web Conference (ESWC) 2015, which focuses on the production of new knowledge aimed at either populating and enriching existing knowledge bases or creating new ones. The first task defined in the Challenge focuses on extracting concepts, individuals, properties, and statements that not necessarily exist already in a target knowledge base whereas the second task addresses entity typing and class induction. Then results for the two tasks are represented according to Semantic Web standard in order to be directly injected in linked datasets and their ontologies. This is in line with available efforts in

the community¹ to uniform results of existing knowledge extraction (KE) methods to make them directly reusable for populating the SW. Indeed, most of the work addressed so far in the literature on knowledge extraction and discovery are focused on linking extracted facts and entities to concepts already existing on available Knowledge Bases (KB).

The described system, FRED, is a Semantic Web machine reader able to produce a RDF/OWL frame-based representation of a text. Machine reading generally relies on bootstrapped, self-supervised Natural Language Processing (NLP) performed on basic tasks, in order to extract knowledge from text. Machine reading is typically much less accurate than human reading, but can process massive amounts of text in reasonable time, can detect regularities hardly noticeable by humans, and its results can be reused by machines for applied tasks. FRED performs a hybrid (part of the components are trained, part are rule-based), self-supervised variety of machine reading that generates RDF graph representations out of the knowledge extracted from text by tools dedicated to basic NLP tasks. Such graph representations extend and improve NLP output, and are typically customized for application tasks.

FRED integrates, transforms, improves, and abstracts the output of several NLP tools. It performs deep semantic parsing by reusing Boxer [2], which in turn uses a statistical parser (C&C) producing Combinatory Categorical Grammar trees, and thousands of heuristics that exploit existing lexical resources and gazetteers to generate structures according to Discourse Representation Theory (DRT) [11], i.e. a formal semantic representation of text through an event (neo-Davidsonian) semantics.

The basic NLP tasks performed by Boxer, and reused by FRED, include: event detection (FRED uses DOLCE+DnS² [4]), semantic role labeling, first-order logic representation of predicate-argument structures, logical operators scoping (called *boxing*), modality detection, tense representation, entity recognition using TAGME³, word sense disambiguation (the next version is going to use BabelNet⁴), DBpedia for expanding tacit knowledge extracted from text, etc. All is integrated and semantically enriched in order to provide a Semantic Web-oriented reading of a text.

FRED reengineers DRT/Boxing discourse representation structures according to SW and linked data design practices in order to represent events, role labeling, and boxing as typed n -ary logical patterns in RDF/OWL. The main class for typing events in FRED is `dul:Event`⁵. In addition, some variables created by Boxer as discourse referents are reified as individuals when they refer to something that has a role in the formal semantics of the sentence.

Linguistic Frames [13], Ontology Design Patterns [8], open data, and various vocabularies are reused throughout FRED's pipeline in order to resolve, align,

¹ e.g. <http://aksw.org/Projects/GERBIL.html>

² D. U. L. Ontology. <http://www.ontologydesignpatterns.org/ont/dul/dul.owl>

³ <http://tagme.di.unipi.it/>

⁴ <http://babelnet.org/>

⁵ Prefix `dul:` stands for <http://www.ontologydesignpatterns.org/ont/dul/dul.owl#>

or enrich extracted data and ontologies. The most used include: VerbNet⁶, for disambiguation of verb-based events; WordNet-RDF⁷ and OntoWordNet [6] for the alignment of classes to WordNet and DOLCE; DBpedia for the resolution and/or disambiguation of named entities, as well as for enriching the graph with existing facts known to hold between those entities; schema.org (among others) for typing the recognized named entities. For Named Entity Recognition (NER) and Resolution (a.k.a. Entity Linking) FRED relies on TAGME [3], an algorithmic NER resolver to Wikipedia that heavily uses sentence and Wikipedia context to disambiguate named entities.

Besides the graph visualization⁸ displayed using Graphviz⁹, and the triple output, FRED can be recalled also as a REST API with RDF serialization in many syntaxes so that everyone can build online end-user applications that integrate, visualize, analyze, combine and infer the available knowledge at the desired level of granularity.

FRED is also accessible by means of a Python API, namely *fredlib*. It exposes features for retrieving FRED graphs from user-specified sentences, and managing them. More specifically, a simple Python function hides details related to the communication with the FRED service and returns to the user a FRED graph object that is easily manageable. FRED graph objects expose methods for retrieving useful information, including the set of individual and class nodes, equivalences and type information, categories of FRED nodes (events, situations, qualities, general concepts) and categories of edges (roles and non roles). *fredlib* supports *rdflib*¹⁰ (for managing RDF graphs) and *networkx*¹¹ (for managing complex networks) libraries. It can be freely downloaded¹².

Additional visual interfaces to FRED can be experienced in the Sheldon¹³ framework. Potentially, each stakeholder interested in semantic aggregate information for multilingual text could be a customer of the system. As FRED has been successfully applied [16,7,9,18,10,15,12,17,5] in the past in several domains, we want to move forward towards the market uptake exploitation; in fact, the foundation of a start-up exploiting FRED's technology (with only commercially-viable components) as one of its main cutting-edge products is currently on-going.

The rest of this paper is organized as follows: Section 2 introduces FRED and shows how it works. Section 3 discusses how we addressed the first two tasks requirements of the challenge; in particular, Section 3.1 shows the capabilities of FRED that we used to solve task 1 requirements (named entity resolution, linking, typing for knowledge base population) whereas Section 3.2 shows FRED's capabilities for solving task 2 (entity typing and knowledge base enrichment).

⁶ T. V. project. <http://verbs.colorado.edu/~mpalmer/projects/verbnet.html>

⁷ <http://www.w3.org/TR/wordnet-rdf/>

⁸ Available at <http://wit.istc.cnr.it/stlab-tools/fred/>

⁹ Graphviz - Graph Visualization Software, <http://www.graphviz.org/>

¹⁰ <http://code.google.com/p/rdflib/>

¹¹ <https://networkx.github.io/>

¹² <http://wit.istc.cnr.it/stlab-tools/fred/fredlib>

¹³ Sheldon - available at <http://wit.istc.cnr.it/stlab-tools/sheldon/>

Section 4 include the description of the datasets and the evaluation of the challengers’ systems (including the one we propose in this paper) on those datasets for the two tasks mentioned above. Section 5 draws conclusions and sketches out future directions where we are headed.

2 FRED at work

FRED has a practical value to either general Web users or application developers. General Web users can appreciate the graph representation of a given sentence using the visualization tools provided, and semantics expert can analyze the RDF triples in more detail. More important, application developers can use the REST API for empowering applications using FRED capabilities. Developers of semantic technology applications can use FRED by automatically annotating text, by filtering FRED graphs with SPARQL, and by enriching their datasets with FRED graphs, and with the knowledge coming from linkable datasets.

FRED’s user main interface, available at <http://wit.istc.cnr.it/stlab-tools/fred/>, allows users to type a sentence in any language, to specify some optional features, and to decide the format of the output to produce. Available formats include RDF/XML, RDF/JSON, TURTLE, N3, NT, DAG, and the intuitive graph visualization using Graphviz.

The reader will notice that FRED will always provide the results in English, although Bing Translation APIs¹⁴ have been used and embedded within FRED to support users specifying their input sentence in any desired language. If the used language of the sentence is different than English, then the tag `<BING_LANG:lang>` needs to precede the sentence, where *lang* indicates a code for the language of the current sentence¹⁵. For example, the sentence:

<BING_LANG:it> Nel Febbraio 2009 Evile iniziò il processo di pre-produzione per il loro secondo album con Russ Russell.

would be a valid Italian sentence to be processed. The English translation for this sentence is *In February 2009 Evile began the pre-production process for their second album with Russ Russell*. Fig. 1 shows the produced output for this sentence. As shown, FRED produces as output an RDF graph with several associated information (detected DBpedia entities, events and situations mapped within DOLCE, WordNet and VerbNet mapping, pronoun resolution).

Additionally, FRED reuses the Earmark vocabulary and annotation method [14] for annotating text segments with the resources from its graphs¹⁶. For example, in the example sentence of Figure 1, the term “Evil”, starting from the text

¹⁴ <http://www.microsoft.com/web/post/using-the-free-bing-translation-apis>

¹⁵ check <http://msdn.microsoft.com/en-us/library/hh456380.aspx> for the list of language codes.

¹⁶ These triples are not returned in the graph-view result of FRED at <http://wit.istc.cnr.it/stlab-tools/fred/>, they are returned with all other serialization output options.

3 Addressing the Open Knowledge Extraction Challenge

The Open Knowledge Extraction Challenge focuses on the production of new knowledge aimed at either populating and enriching existing knowledge bases or creating new ones. This means that the defined tasks focus on extracting concepts, individuals, properties, and statements that not necessarily exist already in a target knowledge base, and on representing them according to Semantic Web standard in order to be directly injected in linked datasets and their ontologies.

In this direction, the tasks proposed in the OKE Challenge are structured following a common formalisation; the required output is in a standard SW format (specifically the Natural Language Interchange (NIF) format) and the evaluation procedure is then produced in a publicly way by using a standard evaluation framework.

The OKE challenge is opened to everyone from industry and academia and it is aimed at advancing a reference framework for research on Knowledge Extraction from text for the Semantic Web by re-defining a number of tasks (typically from information and knowledge extraction) by taking into account specific SW requirements. Systems are evaluated against a testing dataset for each task. Precision, recall, F-measure for all the tasks are computed automatically by using GERBIL²⁰, a state of the art benchmarking tool. In the following we will show how we have addressed the first two tasks of the challenge that we have considered.

3.1 Task 1: Named Entity Resolution, Linking and Typing for Knowledge Base population

This task consists of:

1. identifying Named Entities in a sentence and create an OWL individual (`owl:Individual`²¹) statement representing it;
2. link (`owl:sameAs` statement) such individual, when possible, to a reference KB (DBpedia);
3. assigning a type to such individual (`rdf:type`²² statement) selected from a set of given types (i.e., a subset of DBpedia).

In this task by “Entity” we mean any discourse referent (the actors and objects around which a story unfolds), either named or anonymous that is an individual of one of the following *DOLCE Ultra Lite classes* [4]:

- Person;
- Place;
- Organization;
- Role.

²⁰ e.g. <http://aksw.org/Projects/GERBIL.html>

²¹ Prefix owl: stands for <http://www.w3.org/2002/07/owl#>

²² Prefix rdf: stands for <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

Entities also include anaphorically related discourse referents²³.

To address this task, we have implemented a web application, available at <http://wit.istc.cnr.it/stlab-tools/oke-challenge/index.php>, relying upon FRED’s capabilities. The system requires to upload a file in NIF format, as requested by the OKE Challenge, containing a set of sentences to process. Each sentence is then processed independently by means of FRED, producing as output a set of triples, again in NIF format. The result includes the offsets of recognized entities for the processed sentence and can be downloaded by the user.

As an example, given the sentence:

Florence May Harding studied at a school in Sydney, and with Douglas Robert Dundas , but in effect had no formal training in either botany or art.

the system recognizes four entities²⁴:

Recognized Entity	Generated URI	Type	SameAs
Florence May Harding	oke:Florence_May_Harding,	dul:Person	dbpedia:Florence_May_Harding
school	oke:School	dul:Organization	
Sydney	oke:Sydney	dul:Place	dbpedia:Sydney
Douglas Robert Dundas	oke:Douglas_Robert_Dundas	dul:Person	

The evaluation of task 1 includes the following three aspects:

- *Ability to recognize entities* - it is checked whether all strings recognizing entities are identified, using the offsets returned by the systems.²⁵
- *Ability to assign the correct type* - this evaluation is carried out only on the selected four target DOLCE types, as already stated above.
- *Ability to link individuals to DBpedia 2014* - entities need to be correctly linked to DBpedia, when possible (in the sentence above, for example, the referred “Douglas Robert Dundas” is not present within DBpedia; therefore, obviously, no linking is possible).

Precision, recall and F1 for these three subtasks of task 1 are calculated by using GERBIL, as already mentioned above. Initial experiments of our tool with the provided Gold Standard showed precision and recall close to 70%.

3.2 Task 2: Class Induction and entity typing for Vocabulary and Knowledge Base enrichment

This task consists in producing `rdf:type` statements, given definition texts. A dataset of sentences are given as input, each defining an entity (known a priori);

²³ Hence, anaphora resolution has to be take into account for addressing the task

²⁴ Prefix oke: stands for <http://www.ontologydesignpatterns.org/data/oke-challenge/task-1/>, dul: stands for <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#> and dbpedia: stands for <http://dbpedia.org/resource/>

²⁵ Only full matches are counted as correct (e.g. if the system returns “Art School” instead of “National Art School” is counted as a miss).

e.g. the entity: `dpedia:Skara_Cathedral`, and its definition: *Skara Cathedral is a church in the Swedish city of Skara.*

Task 2 requires to:

1. identify the type(s) of the given entity as they are expressed in the given definition;
2. create a `owl:Class` statement for defining each of them as a new class in the target knowledge base;
3. create a `rdf:type` statement between the given entity and the new created classes;
4. align the identified types, if a correct alignment is possible, to a set of given types from DBpedia.

In the task we will evaluate the extraction of all strings describing a type and the alignment to any of the subset of *DOLCE+DnS Ultra Lite classes* [4].

As an example, given the sentence:

Brian Banner is a fictional villain from the Marvel Comics Universe created by Bill Mantlo and Mike Mignola and first appearing in print in late 1985.

and the input target entity *Brian Banner*, task 2 requires to recognize any possible type for it. Correct answers include²⁶:

Recognized string for the type	Generated Type	subClassOf
fictional villain	<code>oke:FictionalVillain</code>	<code>dul:Personification</code>
villain	<code>oke:Villain</code>	<code>dul:Person</code>

Again, the results are provided in NIF format, including the offsets of recognized string describing the type. Initial experiments of our tool with the provided Gold Standard showed precision and recall higher than to 67% for task 2.

The evaluation of this task includes the following two aspects:

- *Ability to recognize strings that describe the type of a target entity* - since strings describing types often include adjectives as modifiers (in the example above, “fictional” is a modifier for villain), in the provided Gold Standard²⁷ all possible options are included; the system result is considered correct if at least one of the possibility is returned.
- *Ability to align the identified type with a reference ontology* - which for this evaluation is the subset of DOLCE+DnS Ultra Lite classes.

Precision, recall and F1 for these subtasks of task 2 are calculated by using GERBIL, as already stated in Section 3.

²⁶ Prefix `oke:` stands for <http://www.ontologydesignpatterns.org/data/oke-challenge/task-2/> and `dul:` stands for <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#>

²⁷ https://github.com/anuzzolese/oke-challenge/tree/master/GoldStandard_sampleData

4 Results

Two datasets have been used for the evaluation of the challengers' systems. Each dataset included 100 entities extracted from Wikipedia. Dataset for task 1 included entities extracted from biographies of Nobel prize winners that covered as many as possible the DOLCE types indicated in the guidelines for such a task. The overall number of triples of dataset for task 1 is 6432 whereas that for task 2 is 3686. Table 1 shows the distribution of entities extracted for task 1.

dul	count
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Person>	167
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Organization>	108
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Place>	96
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Role>	56

Table 1: Distribution of entities for task 1.

For task 2 entities have been extracted in order to distribute as much as possible entities to type on DOLCE types indicated in the guidelines for such a task. Table 2 shows the distribution of entities extracted for task 2.

dul	count
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#InformationEntity>	32
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#PhysicalObject>	30
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Person>	29
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Description>	12
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Personification>	11
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Organization>	10
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Situation>	10
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Organism>	9
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Collection>	7
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Abstract>	6
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Relation>	3
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Role>	2
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#TimeInterval>	2
<http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Process>	1

Table 2: Distribution of entities for task 2.

Both datasets can be downloaded from the main pages of the challenge²⁸.

Tables 3 and 4 show the results of six different metrics calculated by GERBIL on the two tasks on the challengers' systems.

Annotator	Micro F1	Micro Precision	Micro Recall	Macro F1	Macro Precision	Macro Recall
Adel	0.6075	0.6938	0.5403	0.6039	0.685	0.54
Fox	0.4988	0.6639	0.4099	0.4807	0.6329	0.4138
FRED	0.3473	0.4667	0.2766	0.2278	0.3061	0.1814

Table 3: Results on task 1.

On one hand FRED has the strength to be flexible (it was the only submitted system used for two tasks without specific tuning); on the other hand it suffers

²⁸ <https://github.com/anuzzolese/oke-challenge/tree/master/evaluation-data>

Annotator	Micro F1	Micro Precision	Micro Recall	Macro F1	Macro Precision	Macro Recall
CETUS	0.4735	0.4455	0.5203	0.4478	0.4182	0.5328
OAK@Sheffield	0.4416	0.5155	0.39	0.3939	0.3965	0.3981
FRED	0.3043	0.2893	0.3211	0.2746	0.2569	0.3173

Table 4: Results on task 2.

from precision with respect to the other competitors. We are already working on this direction (constantly updating, improving and extending FRED) with the goal to obtain a holistic framework that can perform efficiently a huge set of machine reading and semantic web tasks while at the same time remaining flexible and fast.

5 Conclusions

In this paper we have shown how we have employed FRED, a machine reader that we have developed within our lab, to solve the first two tasks of the Open Knowledge Extraction (OKE) Challenge at the European Semantic Web Conference (ESWC) 2015. Our method uses Discourse Representation Theory, Linguistic Frames, Combinatory Categorical Grammar and is provided with several well known base ontologies and lexical resources. FRED is a novel approach and to the best of our knowledge we have not found yet a machine reader that computes similar tasks of it and that can be compared. Its novelty is therefore guaranteed. As future direction we would like to keep improving FRED and transform it in a framework that can be released with some sort of license.

Acknowledgement

The research leading to these results has received funding from the European Union Horizons 2020 the Framework Programme for Research and Innovation (2014-2020) under grant agreement 643808 Project MARIO Managing active and healthy aging with use of caring service robots.

References

1. C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009.
2. J. Bos. Wide-Coverage Semantic Analysis with Boxer. In J. Bos and R. Delmonte, editors, *Semantics in Text Processing*, pages 277–286. College Publications, 2008.
3. P. Ferragina and U. Scaiella. 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, CIKM '10*, pages 1625–1628, New York, NY, USA, 2010. ACM.
4. A. Gangemi. What’s in a schema? *Cambridge University Press, Cambridge, UK*, pages 144–182, 2010.

5. A. Gangemi, F. Draicchio, V. Presutti, A. G. Nuzzolese, and D. R. Recupero. A machine reader for the semantic web. In E. Blomqvist and T. Groza, editors, *International Semantic Web Conference (Posters & Demos)*, volume 1035 of *CEUR Workshop Proceedings*, pages 149–152. CEUR-WS.org, 2013.
6. A. Gangemi, A. G. Nuzzolese, V. Presutti, F. Draicchio, A. Musetti, and P. Ciancarini. Automatic typing of dbpedia entities. In *International Semantic Web Conference (1)*, pages 65–81, 2012.
7. A. Gangemi, A. G. Nuzzolese, V. Presutti, F. Draicchio, A. Musetti, and P. Ciancarini. Automatic typing of dbpedia entities. In *Proceedings of the 11th International Conference on The Semantic Web - Volume Part I*, ISWC’12, pages 65–81, Berlin, Heidelberg, 2012. Springer-Verlag.
8. A. Gangemi and V. Presutti. Ontology Design Patterns. In S. Staab and R. Studer, editors, *Handbook on Ontologies, 2nd Edition*. Springer Verlag, 2009.
9. A. Gangemi, V. Presutti, and D. R. Recupero. Frame-based detection of opinion holders and topics: A model and a tool. *IEEE Comp. Int. Mag.*, 9(1):20–30, 2014.
10. A. D. Iorio, A. G. Nuzzolese, and S. Peroni. Towards the automatic identification of the nature of citations. In A. G. Castro, C. L. 0002, P. W. Lord, and R. Stevens, editors, *SePublica*, volume 994 of *CEUR Workshop Proceedings*, pages 63–74. CEUR-WS.org, 2013.
11. H. Kamp. A theory of truth and semantic representation. In J. A. G. Groenendijk, T. M. V. Janssen, and M. B. J. Stokhof, editors, *Formal Methods in the Study of Language*, volume 1, pages 277–322. Mathematisch Centrum, 1981.
12. A. Musetti, A. G. Nuzzolese, F. Draicchio, V. Presutti, E. Blomqvist, A. Gangemi, and P. Ciancarini. Aemoo: Exploratory search based on knowledge patterns over the semantic web. *Semantic Web Challenge*, 2012.
13. A. G. Nuzzolese, A. Gangemi, and V. Presutti. Gathering Lexical Linked Data and Knowledge Patterns from FrameNet. In *Proc. of the 6th International Conference on Knowledge Capture (K-CAP)*, pages 41–48, Banff, Alberta, Canada, 2011.
14. S. Peroni, A. Gangemi, and F. Vitali. Dealing with markup semantics. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 111–118. ACM, 2011.
15. V. Presutti, S. Consoli, A. G. Nuzzolese, D. Reforgiato Recupero, A. Gangemi, I. Bannour, and H. Zargayouna. Uncovering the semantics of wikipedia wikilinks. 19th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2014), 2014.
16. V. Presutti, F. Draicchio, and A. Gangemi. Knowledge extraction based on discourse representation theory and linguistic frames. In *Knowledge Engineering and Knowledge Management*, volume 7603 of *Lecture Notes in Computer Science*, pages 114–129. Springer, 2012.
17. D. R. Recupero, S. Consoli, A. Gangemi, A. G. Nuzzolese, and D. Spampinato. A semantic web based core engine to efficiently perform sentiment analysis. In *The Semantic Web: ESWC 2014 Satellite Events - ESWC 2014 Satellite Events, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers*, pages 245–248, 2014.
18. D. Reforgiato Recupero, V. Presutti, S. Consoli, A. Gangemi, and A. G. Nuzzolese. Sentilo: Frame-based sentiment analysis. *Cognitive Computation*, 2014.