Semi-automatic Indexing and Parsing Information on the Web with NooJ

Maria Pia di Buono(✉)

Department of Political, Social and Communication Sciences,
University of Salerno, Fisciano (SA), Italy
mdibuono@unisa.it

Abstract. Due to the large amount of data available on the Web, indexing information represents a crucial step to guarantee fast and accurate Information Retrieval (IR). Indexing content allows to find relevant documents on the basis of a user's query. Numerous researches discuss the use of automated indexing, considered faster and cheaper than manual systems. However, in order to produce the index, using algorithms, entails low precision, low recall, and generic results [1]. This is the reason why in this paper we propose a NooJ-based system, by means of which we will develop a search engine able to process online documents, starting from a natural language query, and to return information to users. To do this, and in order to analyze user's request, we will employ software automations to apply NooJ and its Linguistic Resources (LRs).

Keywords: Semantic indexing · Archaeological Italian electronic dictionaries · CIDOC CRM · NooJ linguistic resources

1 Introduction

Indexing information represents a crucial step in order to guarantee fast and accurate Information Retrieval (IR). Actually, according to the definition of [2], IR is composed of different steps which are clearly summarized by [3]:

> "finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)".

Such definition allows us to focus on the main characteristic of IR process, namely the need of dealing with three elements: Web pages, queries, sets of relevant Web pages. Thus, indexing content allows extracting relevant documents from Knowledge Bases (KBs) starting from user's queries.

This process presents some problems related to information fragmentation and the growing complexity of KBs. Most approaches employ a shallow linguistic analysis[1], based on the use of statistical parsers, in order to analyze users' queries and convert them into a machine-readable format.

[1] "The shallow semantic analysis measures only word overlap between text and hypothesis" [6]. This means that starting from tokenization and lemmatization of text and hypothesis, this analysis uses Web documents as corpus and assigns inverse document frequency as weight to each entry in the hypothesis.

© Springer International Publishing Switzerland 2016
T. Okrut et al. (Eds.): NooJ 2015, CCIS 607, pp. 151–161, 2016.
DOI: 10.1007/978-3-319-42471-2_13

In our opinion, applying a deep linguistic analysis to free-texts and queries represents the possibility to overcome boundaries in IR systems, guaranteeing an improvement of results.

2 State of Art

Generally speaking automated indexing is considered faster and cheaper than manual systems, even if results do not seem have a high level of accuracy.

Actually, also according to [1], using stochastic algorithms entails:

- Low precision due to the indexing of common Atomic Linguistic Units (ALUs) or sentences.
- Low recall caused by the presence of synonyms.
- Generic results arising from the use of too board or too narrow terms.

Usually IR systems are based on invert text index, namely an index data structure storing a mapping from content to its locations in a database file, or in a document or a set of documents[2].

Most traditional IR systems process each document separately to retrieve terms in free-text query, which means that they do not compare results provided from different sources.

Such lack of integration in results causes overlapping and decreasing in the positive predictive value, due to the fact that shared content are indexed several times. Various approaches have been proposed to overcome this boundary, increasing recall and precision in results.

[4] propose a mixed approach in order to process queries, involving both tree-based navigation and pattern matching similar to that structured information retrieval domains.

In his presentation [5] from Yahoo! Labs deals with query evaluation strategies, based on Term-at-a-Time (TAAT) and Document-at-a-Time Evaluation (DAAT) processing. TAAT scan postings list one at a time, maintain a set of potential matching documents along with their partial scores. On the other hand, DAAT scan postings lists in parallel, identifying at each point the next potential candidate document and scoring it.

In recent times, various semantic approaches have been proposed in order to outline concept identification methods, able to assign document ALUs to the correct ontological entries [7–9].

Furthermore, different researches employ concept-based in order to process both documents and queries through semantic entities and concepts.

[10] propose an approach for semantic indexing based on concepts identified from a linguistic resource. In their work, authors use WordNet and WordNetDomains lexical databases with the aim to identify concepts and they also apply a concept-based indexing evaluation.

[2] Source: https://en.wikipedia.org/wiki/Inverted_index.

2.1 Framework

Our research activities are based on Lexicon-Grammar (LG) theoretical and practical framework, which is one of the most consistent methods for natural language formalization, automatic textual analysis and parsing. LG, set up by the French linguist Maurice Gross during the '60s [11, 12], was applied to Italian by [13]. The Italian Linguistic Resources have been built by the Computational Linguistic group of University of Salerno, which started its study of language formalization from 1981. Our analysis is based on the Italian module for NooJ [14], which is enriched with specific-domain LRs, namely Archaeological domain LRs.

3 System Overview

We propose a system workflow [Fig. 1] which aims at integrating a semantic annotation process for both query analysis and document retrieval.

Also, we propose an architecture, which takes advantage from semantic information stored both in electronic dictionaries and Finite State Automata and Transducers (FSA/FSTs). Furthermore, this architecture may also map linguistic tags (i.e. POS) and structures (i.e. sentences, ALUs) to domain concepts employing metadata from conceptual schemata.

Therefore, the system workflow is based on a representation model applied to both users' queries and to documents, and on a match between these two elements.

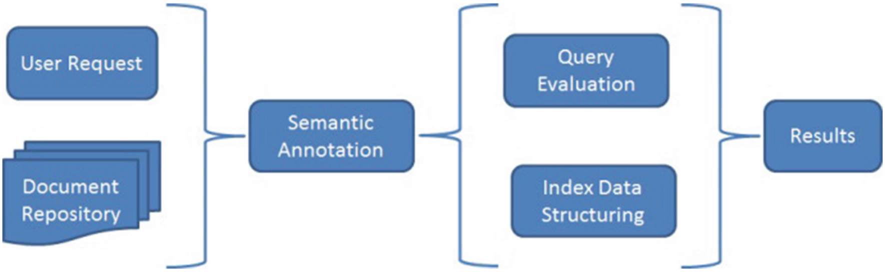


Fig. 1. System workflow.

The representation model proposed is developed on a semantic annotation process to guarantee the interoperability between metadata. In fact, queries may include some restrictions on metadata, such as URL, domain, etc., which are typically different for each document. In order to support these queries, the representation model uses ontological schema to map ALUs with concepts for avoiding overlapping and indexing shared content just once. Semantic association is also used to infer Boolean relationship between elements in a free-text queries and relative meta-data.

In other words, starting from the analysis of users' queries and structured documents, we employ a semantic annotation process in order to create a match among concepts.

In the following paragraphs, we will introduce our methodology based on the use of NooJ and its LRs.

4 Linguistic Resources

In our experiment, we use DBpedia database as knowledge source of structured data in RDF/XML and we test our system outputs using its SPARQL (Protocol and RDF Query Language) Endpoint.[3]

The proposed hybrid approach applies the Lexicon-Grammar framework and its language formalization approach. Thus, firstly we develop Linguistic Resources (LRs), then we model data semantically using two kinds of ontologies: an upper level ontology, namely a cross-domain ontology, and a specific-domain ontology.

The main components of our system are LRs, namely electronic dictionaries, FSA/FSTs [15, 16]. Actually, we develop Italian LRs for the Archaeological Domain, starting from NooJ module for Italian. Such resources have been created and maintained by the research group of University of Salerno, under the LG framework.

As presented in [17, 18], we developed the ARCHAEOLOGICAL ITALIAN Electronic Dictionary (AIED) starting from Thesauri and Guidelines of the Italian Central Institute for the Catalogue and Documentation (ICCD)[4]. Such resources are useful to provide information about the use of terminology and controlled vocabularies for cataloguers and other professionals. It means that they include terms, descriptions and other information needful to objects cataloguing.

In our dictionary, for each entry we indicate:

- Its POS (Category), internal structure and inflectional code (FLX). These information represent a formal and morphological description. In fact, the category and the internal structure indicate that the given ALU is formally a Noun and is formed by different single elements. In Table 1, the tag "NPREPNPREPN" describes how the given ALU, *dinos con anse ad anello*, is formed (i.e. N stands for Noun and PREP for Preposition). At the same time, the tag "FLX = C610" refers to the ALU number and gender recalling a local grammar in order to generate and recognize correspondent forms (e.g. singular/plural, masculine/feminine).
- Its variants (VAR) and synonyms (SYN), if any;
- The type of link (LINK) (RDF and/or HTML), associated to the linguistic resource;
- With reference to a taxonomy, the pertaining knowledge domain (DOM); for our dictionary we have developed a taxonomy, based on ICCD prescriptions, therefore all entries have a terminological and domain label usable for ontology population.
- The use of domain label subset tags is also previewed for those domain sectors which include specific sub-sectors. This is the case with Archaeological Remains,

[3] A SPARQL Endpoint is a service which accept a SPARQL query against a knowledge base and return results. Several tools, able to connect and semi-automatically construct a SPARQL query for a SPARQL endpoint, are available, e.g., ViziQuer.
DBpedia SPARQL Endpoint is available at http://dbpedia.org/isparql/.

[4] http://www.iccd.beniculturali.it/index.php?it/240/vocabolari. Last visit: February 2014.

for which a generic tag «RA1» is used, while more explicit tags are used for Object Type, Subject, Primary Material, Method of Manufacture, Object Description.

- The ICOM International Committee for Documentation (CIDOC) Conceptual Reference Model (CRM) Class (CCL). In AIED we associate the ontology schema, provided by CIDOC[5] and compatible with the Resource Description Framework (RDF), to lexical entries. Actually, the tag CCL allows us to derive definitions and a formal structure for describing the implicit and explicit concepts and relationships used in Cultural Heritage documentation.

Table 1. Sample of AIED entries.

Entry	Category	Internal structure	FLX	VAR	SYN	LINK	DOM	CCL
dinos con anse ad anello	N	NPNPN	C610	dynos/déinos		RDF	RA1SUOCR	E22
kylix a labbro risparmiato	N	NPNA	C611		lip cup	RDF	RA1SUOCR	E22

CCL label and grammatical information with which dictionary entries are tagged, are the basis on which we develop role set matrixes. Such matrixes are useful to identify predicate-argument structures related to sentence contexts and consequently to achieve the semantic annotation process. Context information inserted inside the matrix tables together with NooJ concordances are employed as weighting preferences.

These matrix tables are developed analyzing semantic role sets established on the basis of CIDOC CRM constraints (properties) matched with grammatical and syntactic rules. Also, they indicate if a verb allows active/passive constructions, in order to recognize entities also when analyzing transformed active declarative sentences.

5 Semantic Annotation

In our system, we present a semantic annotation process which works simultaneously on two sides. Actually, it analyses (I) the user's query, and (II) documents in KBs.

Such annotation process is based on a deep Natural Language Analysis, which means that we perform a linguistic analysis of user's queries and documents in order to annotate them.

Semantic annotation represents a key step in our procedure, due to the fact that annotating text requires the capability of matching correctly a natural language formalism and a data model formalism. Actually, as stated in [19], "annotation is the

[5] The CIDOC Conceptual Reference Model (CRM) aims at providing semantic definitions to describe implicit and explicit concepts and relations between Cultural Heritage objects and museum documentations. It is a formal ontology, which allows integration, mediation and interchange of heterogeneous information. CIDOC CRM only defines basic semantics for database schemes and document structures.

inverse of normalization. Just as different strings of characters may have the same meaning, it also happens that identical strings of characters may have different meanings, depending on the context".

Thus, we may divide semantic annotation task into two subtasks: natural language analysis and data representation.

Therefore, during such process we employ two conceptual schemata:

- DBpedia (upper level) ontology which is composed of:
  Classes: 734
  Properties: 2975
- CIDOC CRM (domain) ontology which is composed of:
  Classes: 90
  Properties: 148.

5.1 Natural Language Analysis

The first task concerns the processing of user's queries in order to annotate them, domain-independent semantic data modeling (DBpedia cross-domain ontology) and inferring Boolean relationship among elements in a free-text query and relative meta-data. In our work, we develop NooJ FSA/FSTs in order to process a given query.

Starting from the entries retrieved and from their specific tags, stored in electronic dictionaries and in FSA/FSTs, we use NooJ to write and fill all fields directly using RDF schema and OWL, automatically generating the strings while correctly coupling ontologies and compound words.

In fact, in our FSA-based system we recognize RDF triples in sentence structures.

According to our approach, electronic dictionaries entries (simple words and ALUs) are the subject and the object of the RDF triple.

Also, as regards declarative sentences, RDF gives the possibility to recognize sentences conveying information of the type "X is an element of Y", which also have recursive structures.

All this means that a single FSA/FST can be used to:

- account for all the items of an open list;
- account for all declarative sentences of the type "X is a part of Y", in which X and Y are pre-defined classes;
- allow the matching of POS to RDF triples.

In the following images Figs. 2 and 3[6], we will show a sample of FSA/FST which may be used to analyze users' queries. Such automaton allows us to recognize entities involved in RDF relationships, namely *Person* and *date*. In such RDF triple, the subject, *Person*, and the object, *date,* are trigged by a predicate, namely a Verb Phrase (VP). This VP is represented by a class verb which may co-occur together with the given entities in sentence contexts. Therefore, the VP may hold verbs such as *live* or *born* followed by a

[6] We had to split FSA image in order to improve reading, thus, Figs. 2 and 3 represent respectively the left side and the right side of the automaton. The node *Person,* which stands for the variable "activity2", is repeated twice in order to link the two split figures.

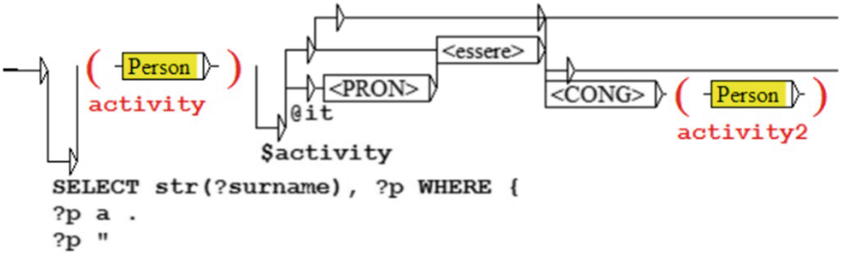


Fig. 2. FSA for annotating users' queries (left side).

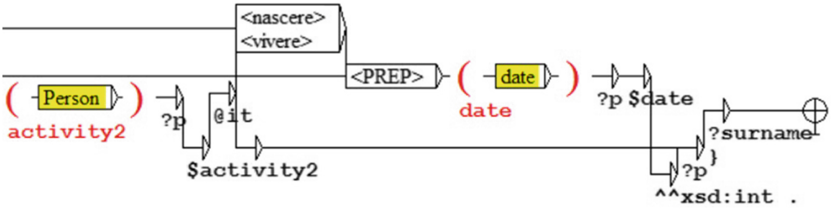


Fig. 3. FSA for annotating users' queries (right side).

preposition. It is worth noticing that in our sample we insert two nodes containing the same entity (*Person*), which stands for two different variables, namely *activity* and *activity2*. Such variables refer to a specific CCL tag, which is used to identify a specific attribute, namely profession, for elements belonging to the generic class *Person.*

Values, produced by variables (*activity, activity2* and *date*), are employed to generate a SPARQL query, able to retrieve surname of such persons which perform a specific activity/job/profession in a determinate interval.

The following sample shows the result of FSA applied to the previous query.

```
SELECT str(?surname), ?p WHERE {
  ?p a .
  ?p "scrittore"@it .
  ?p "archeologo"@it .
  ?p "1900"^^xsd:int .
  ?p ?surname
}
```

[Example of pseudo-code query in SPARQL which may be used into an Endpoint]

Thus, the output of FSA may be used in order to generate a query which may be run against any SPARQL endpoint or repository in which documents are formalized using RDF.

5.2 Data Representation

The second subtask, namely data representation, involves appropriate operations on the RDF-based data layer, mapping OWL concepts to object-oriented classes with methods for interrelations and domain-specific rules used to generate and consolidate processes (e.g. CIDOC CRM ontology).

Such process of data representation aims at analyzing information stored in RDF documents, which means that we may retrieve information from any repository directly. Actually, we use RDF data representation in order to process documents and create a match between users' requests and concepts stored in KBs.

We develop NooJ FSA in order to process information stored in DBpedia KB, matching values of semantic attributes with the ones retrieved from users' queries analysis.

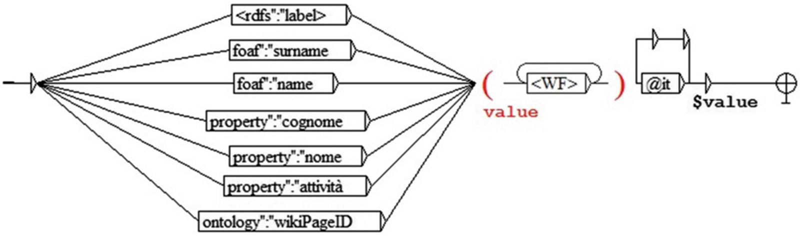
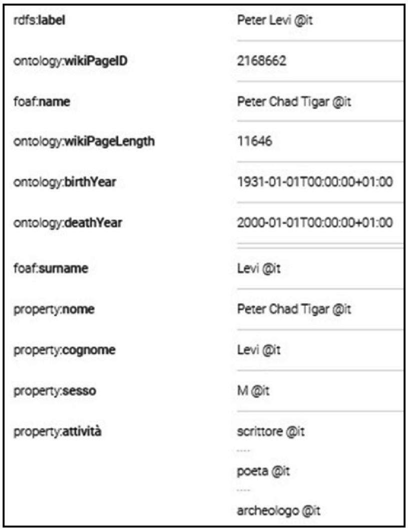


Fig. 4. Sample of FSA for analyzing DBpedia documents.

In the previous FSA (Fig. 4) we use the nodes on the left in order to recognize labels used inside RDF documents, which are stored, for example, in DBpedia KB. It means that first we process tags which describe elements semantically and subsequently we analyze which values are assumed for such descriptions. Actually, in the following node, we insert a generic *WF* class, in order to recognize each word form



rdfs:**label**	Peter Levi @it
ontology:**wikiPageID**	2168662
foaf:**name**	Peter Chad Tigar @it
ontology:**wikiPageLength**	11646
ontology:**birthYear**	1931-01-01T00:00:00+01:00
ontology:**deathYear**	2000-01-01T00:00:00+01:00
foaf:**surname**	Levi @it
property:**nome**	Peter Chad Tigar @it
property:**cognome**	Levi @it
property:**sesso**	M @it
property:**attività**	scrittore @it poeta @it archeologo @it

Fig. 5. Sample of RDF-structured document from Italian DBpedia resources.

which is present inside documents. These word forms represent values stored for each specific semantic descriptive tag, i.e. for *foaf:surname* Levi value (see Fig. 4).

On the other hand, the final node *@it* indicates language tag in resource description schemata (i.e., Italian).

Thus, we may retrieve information structured as the previous sample of RDF-structured document (Fig. 5).

6 Tests and Conclusions

We decide to test our system against DBpedia Knowledge Base (KB)[7], which is structured in RDF. There also is a public SPARQL endpoint over the DBpedia data set[8] and, as reported in the site, users can ask queries against DBpedia using:

- the Leipzig query builder[9];
- the OpenLink Interactive SPARQL Query Builder (iSPARQL)[10];
- the SNORQL query explorer[11]; or
- any other SPARQL-aware client(s).

Therefore, DBpedia endpoints may be accessed just using a query encoded in SPARQL. We test our system outputs, i.e. SPARQL queries and data representations, using Italian DBpedia KB[12].

For example, if we run the following query against a KB:

> *Tutti gli archeologi che sono stati anche scrittori nati nel '900* (Archaeologists that have been also a writer lived in 19th century)[13].

Actually our system displays the results as they are showed in DBpedia Endpoint, after processing the query (Table 2); namely as a table which contains surname value, i.e. Levi, and the specific resource URL.

Thus, for the given query we obtain a list of RDF pages which match with user's information need.

We also test our data representation, obtained through NooJ FSA, on a corpus dumped from the Italian Wikipedia Database.

After being tested and debugged, the LRs described so far are actually under final development and completion and they will be proposed as part of the NooJ Italian module.

[7] DBpedia is a crowd-sourced community effort to extract structured information from Wikipedia and make this information available on the Web. http://wiki.dbpedia.org/.

[8] http://dbpedia.org/OnlineAccess.

[9] http://querybuilder.dbpedia.org.

[10] http://dbpedia.org/isparql.

[11] http://dbpedia.org/snorql (does not work with Internet Explorer).

[12] DBpedia Italian is an open and collaborative project for the extraction and reuse of semantically structured information of the Italian version of Wikipedia. For more information see: http://it.dbpedia.org.

[13] Sample adapted from the one presented in the Italian DBpedia project. http://it.dbpedia.org/esempi/.

Table 2. Results from SPARQL query.

Surname value	Resource
Levi	http://it.dbpedia.org/resource/Peter_Levi
Matthiae	http://it.dbpedia.org/resource/Paolo_Mattiae
Hansen	http://it.dbpedia.org/resource/Thorkild_Hansen
Cooper	http://it.dbpedia.org/resource/Glenn_Cooper
Duggan	http://it.dbpedia.org/resource/Alfred_Duggan
Mallowan	http://it.dbpedia.org/resource/Max_Mallowan
Meomartini	http://it.dbpedia.org/resource/Almerico_Meomartini
Coe	http://it.dbpedia.org/resource/Michael_D._Coe
Kondylis	http://it.dbpedia.org/resource/Thanos_Kondylis
Zecca	http://it.dbpedia.org/resource/Vincenzo_Zecca
Bellis	http://it.dbpedia.org/resource/En_Bellis
Consoli	http://it.dbpedia.org/resource/Sebastaino_Consoli

Subsequently, we will integrate such LRs in our environment, considering that our final aim is to propose the development of a SPARQL endpoint based on NooJ. Furthermore, we will focus on an improvement of result displaying.

Future work also aims at improving both index data structuring and a query evaluation process. It also necessary testing the system in a consistent way on other KBs, in order to propose an independent-domain approach.

References

1. Hjorland, B.: Semantics and knowledge organization. Ann. Rev. Inf. Sci. Technol. **41**, 367–405 (2007)
2. Manning, C.D., Raghavan, P., Schütze, H.: An Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
3. Teufel S.: Lecture 1: Introduction and Overview. Information Retrieval Computer Science Tripos Part II. http://www.cl.cam.ac.uk/teaching/1314/InfoRtrv/lecture1.pdf
4. Halverson, A., Burger, J., Galanis, L., Kini, A., et al.: Mixed mode XML query processing. In: Proceedings of the 29th VLDB Conference, Berlin, German (2003)
5. Lempel, R.: http://webcourse.cs.technion.ac.il/236621/Winter2010-2011/ho/WCFiles/lec4-evaluation.pdf
6. Bos, J., Markert, K.: Marketer combining shallow and deep NLP methods for recognizing textual entailment. In: Proceedings of the First Challenge Workshop, Recognizing Textual Entailment. PASCAL (2005)
7. Baziz, M., Boughanem, M., Aussenac-Gilles, N.: Conceptual indexing based on document content representation. In: Crestani, F., Ruthven, I. (eds.) CoLIS 2005. LNCS, vol. 3507, pp. 171–186. Springer, Heidelberg (2005)
8. Boubekeur, F., Boughanem, M., Tamine, L., Daoud, M.: Using WordNet for concept-based document indexing in information retrieval. In: Fourth International Conference on Semantic Processing (SEMAPRO), Florence, Italy, October 2010

9. Sussna, M.: Word sense disambiguation for free-text indexing using a massive semantic network. In: 2nd International Conference on Information and Knowledge Management (CIKM-1993), pp. 67–74 (1993)
10. Boubekeur, F., Azzoug, W.: Concept-based indexing in text information retrieval. Int. J. Comput. Sci. Inf. Technol. (IJCSIT) **5**(1), 119–136 (2013)
11. Gross, M.: Grammaire transformationnelle du français. Cantilène (1968)
12. Gross, M.: Méthodes en syntaxe. Hermann (1975)
13. Elia, A., Martinelli, M., D'Agostino, E.: Lessico e strutture sintattiche: introduzione alla sintassi del verbo italiano. Liguori (1981)
14. Vietri, S.: The Italian module for NooJ. In: Proceedings of the First Italian Conference on Computational Linguistics, CLiC-it 2014. Pisa University Press (2014)
15. Silberztein, M.: La Formalisation des Langues. L'Approche de NooJ. ISTE Edition (2015)
16. Silberztein, M.: NooJ computational devices. In: Donabédian, A., Khurshudian, V., Silberztein, M. (eds.) Formalising Natural Languages with NooJ: Selected Papers from the NooJ 2012 International Conference. Cambridge Scholars Publishing, Newcastle (2013)
17. di Buono, M.P.: Information extraction for ontology population tasks. An application to the Italian archaeological domain. Int. J. Comput. Sci. Theor. Appl. **3**(2), 40–50 (2015). ORB Academic Publisher
18. di Buono, M.P., Monteleone, M., Elia, A.: Terminology and knowledge representation Italian linguistic resources for the archaeological domain. In: Proceedings of 25th International Conference on Computational Linguistics (COLING 2014) - Workshop on Lexical and Grammatical Resources for Language Processing (LG-LP 2014) (2014)
19. Turney, P.D.: From frequency to meaning: vector space models of semantics. J. Artif. Intell. Res. **37**, 141–188 (2010)