Key-concept Extraction for Ontology Engineering

Marco Rospocher, Sara Tonelli, Luciano Serafini, and Emanuele Pianta*
FBK-irst, Via Sommarive 18 Povo, I-38123,Trento, Italy
{*rospocher,satonelli,serafini,pianta*}*@fbk.eu*

Abstract. We present a framework for supporting ontology engineering by exploiting key-concept extraction. The framework is implemented in an existing wiki-based collaborative platform which has been extended with a component for terminology extraction from domain-specific textual corpora, and with a further step aimed at matching the extracted concepts with pre-existing structured and semi-structured information. Several ontology engineering related tasks can benefit from the availability of this system: ontology construction and extension, ontology terminological validation and ranking, and ontology concepts ranking.

1 Research background

The development of ontologies is a crucial task in many fields in which knowledge needs to be shared and reused, such as knowledge management, e-commerce, database design, educational applications, and so on. Building an ontology is a complex task that requires a considerable amount of human effort.

We claim that the developing of ontologies can benefit from the automatic extraction of knowledge from documents related to the ontology domain and to its linking with available structured and semi-structured resources, such as *WordNet*[1] and *Wikipedia*[2].

We describe an on-line environment in which the ontology development process can be performed *collaboratively* in a Wiki-like fashion. To start the construction (or the extension) of an ontology, the user can exploit a domain corpus, from which the terminological component of the system automatically extracts a set of domain-specific key-concepts. These key-concepts are further disambiguated in order to be linked to existing external resources and obtain additional information such as the concept definition, the synonyms and the hypernyms. Finally, the user can easily select through the interface which concepts should be imported into the ontology.

The system support several ontology engineering tasks, as described in Section 3.

2 System description

We present here the main characteristics of our online environment for key-concept based ontology engineering, focusing on its semantic components. In [1], we presented a preliminary version of the system, in which the evaluation and the disambiguation modules were missing.

* Work partially funded by the European Commission (contract FP7-248594, PESCaDO).

[1] `http://wordnet.princeton.edu`

[2] `http://www.wikipedia.org`

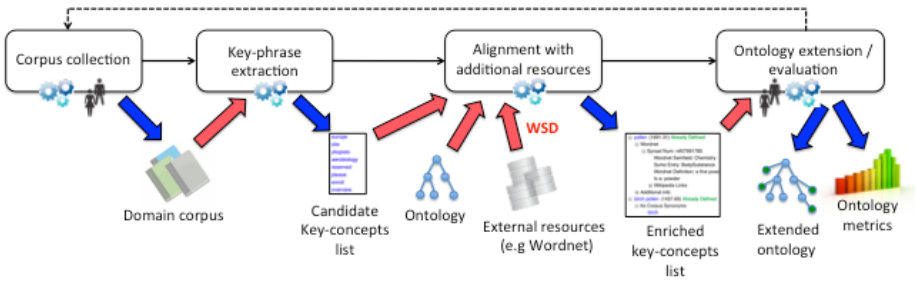

Fig. 1: System workflow for ontology building / extension

Starting from a domain-specific textual corpus, the system first extracts a list of key-concepts, then retrieves additional information from available lexical resources, and finally enables the user, through a user-friendly graphical interface, to (i) add new concepts to the ontology, or (ii) check the terminological coverage of an ontology with respect to the domain described in the text corpus.

The workflow has been made available in an on-line collaborative environment which can be potentially used for any kind of ontology domain. We have taken advantage of the existing infrastructure for conceptual modelling provided by MoKi [2][3], a collaborative Mediawiki-based[4] tool for modeling ontological and procedural knowledge in an integrated manner. The main idea behind MoKi is to associate a wiki page to each basic entity of the ontology, i.e., concepts, object and datatype properties, and individuals. The environment has been enriched with a terminology-extraction component and with external lexical resources[5].

The overall workflow is displayed in Figure 1. As a preliminary step, the user should collect a domain corpus in one of the languages supported by the tool (currently, either English or Italian). On this corpus, the module for terminological extraction is first run to obtain a set of domain-specific terms that are seen as candidate concepts for the integration into the ontology. Such terms are obtained using KX [3], a robust system for keyphrase-extraction that first extracts n-grams from a corpus, then recognizes the multi-words by combining lexical-based filtering and black lists, and finally ranks the candidate key-concepts by relevance based on different ranking parameters. Hence, the output is a list of key-concepts ranked by relevance. The tool can be easily configured by the user through the online interface in order to define some basic selection criteria for the key-concepts. For instance, longer[6] key-concepts may be preferred over shorter ones. Note that the system is completely unsupervised and can be easily extended to handle multilingual documents.

[3] See `http://moki.fbk.eu`

[4] See `http://www.mediawiki.org`

[5] A demo of the tool is available at `http://moki.fbk.eu/moki/tryitout2.0/`. After logging in, go to *Import Functionalities*, and then *Extract new concepts from textual resources*

[6] In terms of number of words that compose a key-concept

In the following step, the key-concepts are linked to external resources to enrich the information provided to the user while building, extending or evaluating an ontology. To this purpose, the key-concepts are disambiguated by assigning them a WordNet synset. Word sense disambiguation (WSD) is performed using WordNet::SenseRelate::WordToSet library[4], which has been developed in order to disambiguate a word given the textual context in which it appears. We apply this disambiguation procedure in a slightly different way: we build a context by considering the 10 top-ranked key-concepts returned by KX, and disambiguate all other key-concepts in the list based on them. The intuition behind this is that the top-ranked concepts should be the most representative for the domain, and therefore they should build the most appropriate context to disambiguate other concepts from the same domain.

Once a key-concept is associated to a unique WordNet sysnet, additional information from WordNet can be retrieved, such as the gloss, the hypernyms and the synonym list for the given key-concept. All this information is displayed to the user after the extraction process. An example is displayed in Fig. 2, showing the top-ranked key-concepts extracted from a corpus about pollens. In addition, we exploit *BabelNet*[5] (an automatic alignment between Wikipedia pages and WordNet synsets), in order to provide for each synset a link to the corresponding Wikipedia page.

Once a set of key-concepts has been extracted from a domain corpus and enriched with additional information from WordNet and Wikipedia, the platform relies on this information to perform different processes. If a pre-defined ontology has been uploaded, each concept and its WordNet synonyms (if any) are matched against the concepts currently defined in the ontology, in order to understand if the concept (or a synonym of it) is already defined in the ontology under development. If a match is recognized, the key-concept is marked with an X (see Fig. 2), thus reducing the chance that unnecessary duplicated content is added to the ontology. In the current version of the system, the matching is performed by applying normalized matching techniques on concepts (and synonyms) labels.

The matching is exploited in two ways: for *ontology extension*, the user can decide whether to automatically add some of the extracted un-matched concepts to the ontology. For *ontology evaluation*, some metrics derived from standard precision, recall, and F1 measures can be computed based on the matching between ontology concepts and corpus key-concepts extracted in the previous step. These measures aim at estimating how well a given ontology terminologically covers the domain described by the corpus, i.e. to which extent the concepts described in the ontology are appropriate and complete with respect to the domain.

3 System Demonstration

We will demonstrate the support provided by our online environment for ontology engineering in the three following tasks:

Boosting of ontology construction and extension This is the typical situation occurring when users want to build a domain ontology and no structured resources are already available, or when they want to iteratively extend an existing ontology. A sample corpus will be uploaded by the system, together with an ontology (optional). We will show the

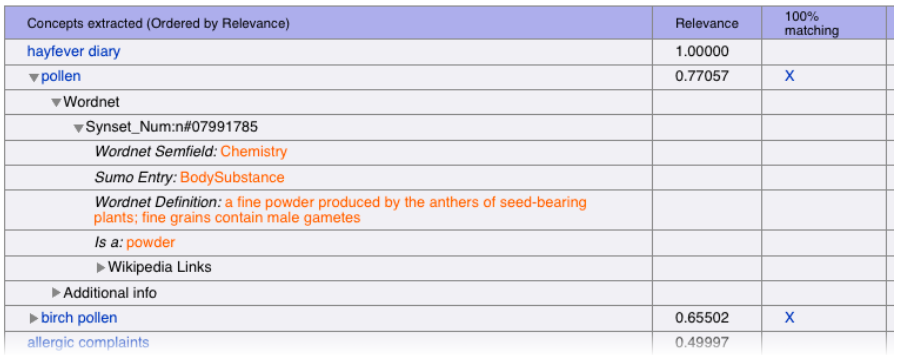

Fig. 2: Screenshot of matching output

support provided by the tool in identifying and adding new domain concepts extracted from the corpus to the given ontology.

Ontology terminological evaluation and ranking Based on the matching between an ontology and corpus key-concepts, the system can compute a set of metrics inspired by precision/recall/F1 to terminologically evaluate an already existing ontology against the domain described by the corpus, or to rank different candidate ontologies according to how well they terminologically cover the domain. This may be helpful when deciding whether to reuse ontologies already available on the web. We will upload a domain corpus, and compute the ontology metrics for different candidate ontologies, showing how to interpret the results obtained.

Ranking of ontology concepts Finally, we will show how to evaluate the relevance of the concepts in an ontology with respect to the domain described by a corpus. Given that an ontology can include thousands of concepts, it is not always easy to understand if some of them are more relevant than others, thus representing a sort of core knowledge of the domain ontology (information which usually is not explicitly encoded in ontologies). We will upload a domain corpus and an ontology, showing which are the most relevant domain-wise concepts of the ontology according to the tool.

References

1. Tonelli, S., Rospocher, M., Pianta, E., Serafini, L.: Boosting collaborative ontology building with key-concept extraction. In: Proceedings of 5th IEEE International Conference on Semantic Computing. (2011)
2. Ghidini, C., Rospocher, M., Serafini, L.: MoKi: A Wiki-Based Conceptual Modeling Tool. In: Proc. of ISWC 2010, Posters and Demonstrations Track, Shanghai, China (2010)
3. Pianta, E., Tonelli, S.: KX: A flexible system for Keyphrase eXtraction. In: Proc. of SemEval 2010, Task 5: Keyword extraction from Scientific Articles, Uppsala, Sweden (2010)
4. Pedersen, T., Banerjee, S., Patwardhan, S.: Maximizing semantic relatedness to perform word sense disambiguation. Technical Report UMSI 2005/25, University of Minnesota (2005)
5. Navigli, R., Ponzetto, S.P.: Babelnet: Building a very large multilingual semantic network. In: Proc. of the 48^{th} Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden (2010)