OntoRich - A Support Tool for Semi-Automatic Ontology Enrichment and Evaluation

Gabriel Barbur, Bogdan Blaga and Adrian Groza
Computer Science Department
Technical University of Cluj Napoca, Romania
adrian.groza@cs.utcluj.ro

***Abstract*—This paper presents the OntoRich framework, a support tool for semi-automatic ontology enrichment and evaluation. The WordNet is used to extract candidates for dynamic ontology enrichment from RSS streams. With the integration of OpenNLP the system gains access to syntactic analysis of the RSS news. The enriched ontologies are evaluated against several qualitative metrics.**

I. Introduction

In recent years, much effort has been put in ontology learning as an imperative for the concept of Semantic Web. The migration from HTML Web to Semantic Web [1] is still considered only a theoretical approach mainly because of the effort that this transformation would imply. Information Extraction methods by means of domain specific templates and the lightweight use of Natural Languages Processing techniques (NLP) have been already proposed [2]. Another good heuristic is to use a search engine to find web pages with relevant content. However, current search engines retrieve web pages, not the information itself [3]. After the information is retrieved, a system for term extraction is needed in order to obtain candidates for ontology enrichment.

The first consideration was to provide an automatic way for information extraction from the web and the considered solution is based on RSS feeds that more and more websites provide nowadays. Because of the standard format a single RSS Reader system is enough to fetch information from many websites that are related to a certain domain.

Several approaches for extracting concepts, instances and relationships exploit separately or integrate statistical methods, semantic repositories such as WordNet, natural language processing libraries such as OpenNLP, or lexicon-syntactic patterns in form of regular expressions [4]. The developed system provide users with the capability to choose among and mix these methods in order to obtain potential candidates for ontology enrichment.

Ontology evaluation is an important task in real life scenarious. When creating an application based on semantic knowledge it is necessary to guarantee that the considered ontology meets the application requirements. Ontology evaluation is also important in cases where the ontology is automatically populated from different resources that might not be homogeneous, leading to duplicate instances, or instances that are clustered according to their sources in the same ontology [5].

II. Theoretical background

a) Statistical methods: The first category of term extraction methods is based on two statistical methods absolute term frequency and TF-IDF weight.

Definition 1. *Absolute term frequency tf_i is defined by $tf_i = n_i / \sum_{i=1} n_i$, where n_i represents the number of times term i appears.*

Definition 2. *Term frequency - inverse document frequency metric (TF-IDF weight) evaluates how important a word is to a document in a collection or corpus, defined by: $(tf-idf)_{i,j} = tf_{i,j} * idf_i$, where $tf_{i,j}$ are the absolute term frequency of term i in document j and and idf_i the inverse document frequency: $idf_i = \log |D|/(j : t_i \in d_j)$ where $|D|$ is the total number of documents in the corpus and $j : t_i \in d_j$ the number of documents where term t_i appears.*

The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus.

b) WordNet based methods: Using the stemming function provided by RiTa WordNet, each word in the text is reduced to its stem form. For example, the stem of 'friendships' is 'friendship', to which the inflectional suffix '-s' is attached. Using this approach many forms of basically the same word can be found and counted in computing the statistical values.

The semantic power of WordNet is exploited by using the methods for retrieving hyponyms and meronyms that RitaWordNet provides. In linguistics, a hyponym is a word or phrase whose semantic field is included within that of another word. For example, 'scarlet', 'vermilion', 'carmine', and 'crimson' are all hyponyms of red (their hypernym), which is, in turn, a hyponym of 'color'. More terms can be obtained by using the hyponym tree provided by WordNet to which RiTa WordNet offers a simple access. In many ways, meronymy is significantly more complicated than hyponymy. The Wordnet databases specify three types of meronym relationships: part meronym (a 'processor' is part of a 'computer'), member meronym and substance (stuff) meronym.

c) NLP based methods: In many cases the text corpus could be easier to use if a syntactic analysis could be applied. With the use of the OpenNLP library the OntoRich system provides users the possibility to: i) split the text into sentences and tag each word with the correct POS (part of speech)

978-1-4577-1481-8/11/$26.00 ©2011 IEEE

within the sentence; ii) use OpenNLP built-in models to extract well-known organization names, person names and date references(e.g. today, Monday, July, etc); and iii) extract potential relations between concepts using the syntactic role that words have within sentences;

d) Metric based evaluation: The first considered type of evaluation is from the design point of view. This kind of metrics are known as *schema metrics*. Metrics in this category indicate the richness, width, depth, and inheritance of an ontology schema design. The implemented schema metrics are *Relationship Richness*, *Inheritance Richness* and *Attribute Richness*. *Relationship Richness*, for example, is defined as:

Definition 3. *Relationship Richness (RR) represents the ratio of the number of non-inheritance relationships (P), divided by the total number of relationships defined for the ontology, inheritance relationships (H) and non-inheritance relationships:* $RR = |P|/(|H| + |P|)$.

The RR metric gives information about the diversity of the types of relations in the ontology; Ontologies can also be evaluated considering the way data is placed within the ontology or in other words, the amount of real-world knowledge represented by the ontology. These metrics are refereed to as knowledge base metrics: *Class Richness*, *Class Connectivity*, *Class Importance*, *Cohesion* and *Relationship Richness*.

Definition 4. *Class importance ($Imp(C_i)$ of a class is defined as the percentage of the number of instances that belong to the inheritance subtree rooted at this class ($inst(C_i)$) in the ontology compared to the total number of class instances in the ontology (CI):* $Imp(C_i) = |Inst(C_i)|/KB(CI)$

It helps to identify which areas of the schema are in focus when the instances are added to the ontology.

III. System Architecture

The proposed OntoRich method for ontology engineering is based on dotNetRDF, an Open Source .Net Library using the latest versions of .Net Framework to provide a powerful and easy to use API for working with Resource Description Framework (RDF). The main components of the system are the $RSS\ Reader$, the $Ontology\ Engineering$ component, the $Ontology\ Enrichment$ component and the $Ontology\ Evaluation$ module (see figure 1).

The $RSS\ Reader$ is a web application created in PHP that distinguishes between two main users: the administrator and the normal user. An advantage of using RSS is that the information provided is always updated, so new concepts or instances that appear in a domain and are useful to be considered for the managed ontology can be found faster.

The $Ontology\ Engineering$ component is the one dealing with loading, displaying, editing and saving ontologies. It is based on the dotNetRDF open source API. dotNetRDF is a .Net library written in C# designed to provide a simple but powerful API for working with RDF data. It provides a large variety of classes for performing all the common tasks from reading and writing RDF data to querying over it.

The $Ontology\ Enrichment$ module deals with extracting new terms that can be added as concepts, instances or relations to the ontology. It is based on RiTa WordNet Java API and OpenNLP Java API. RiTa WordNet is a WordNet library that offers a simple access to the WordNet ontology and also provides distance metrics between ontology terms. OpenNLP is an organizational center for open source projects related to natural language processing. OpenNLP also hosts a variety of Java-based NLP tools which perform sentence detection, tokenization, pos-tagging, chunking and parsing and named-entity detection. The $OntologyEvaluation$ provides to users the option of testing the loaded ontology against some defined ontology metrics and also offers some interesting features such as assessing the evolution in time of an ontology, comparing two ontologies or checking an ontology consistency using the Pellet reasoner. The major approaches currently in use for the evaluation and validation of ontologies using metric-based ontology quality analysis are available.

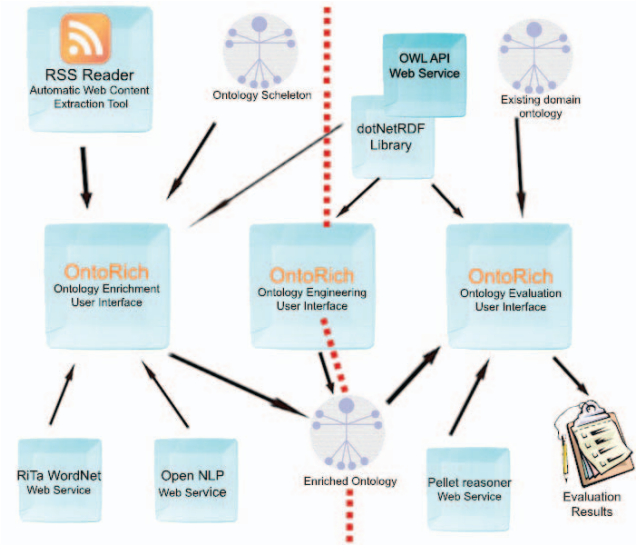

Fig. 1. OntoRich system architecture.

IV. Framework Capabilities

The main features of the OntoRich tool[1] are illustrated with the help of two testing ontologies: the well-known 'Wine' ontology and an IT ontology skeleton created using Protégé.

A. Ontology Engineering

The sytem provides specific features aiming at supporting the management of an ontology. In order to graphically display the ontology, a tree structure is used with nodes representing classes. The instances of every class can be seen in a separate window as well as the relationships defined in the schema. The main features that the ontology engineering component provides are: loading ontologies from a local file or URI, displaying ontologies in the form of a tree view or

[1]The sytems is available at http://cs-gw.utcluj.ro/∼adrian/ontorich

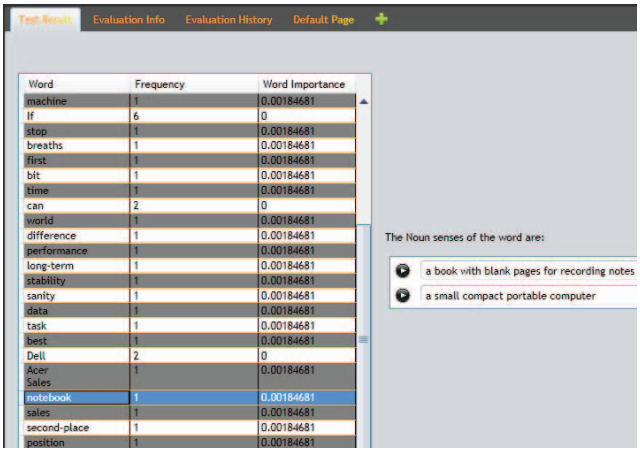

Fig. 2. Term extraction results.

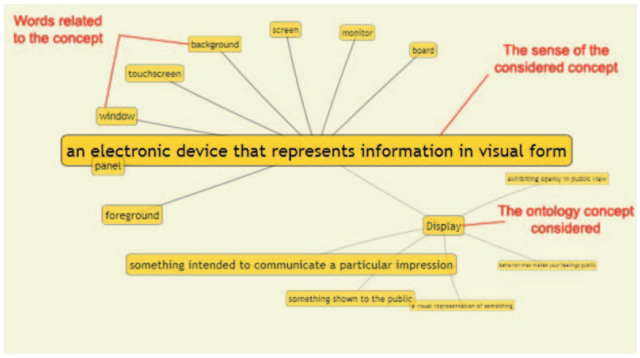

Fig. 3. Hyponym Tree for the term 'computer'.

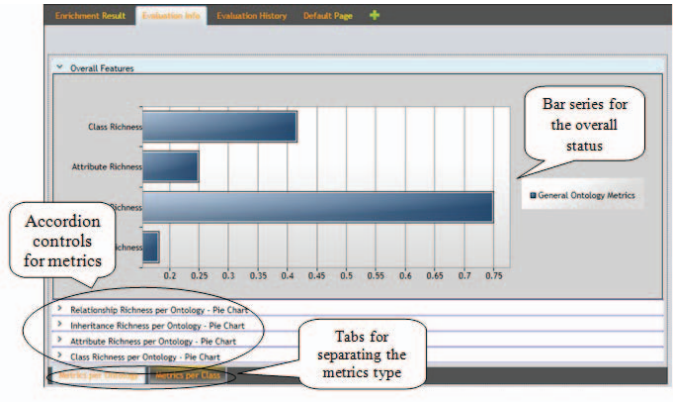

Fig. 4. Overall evaluation result.

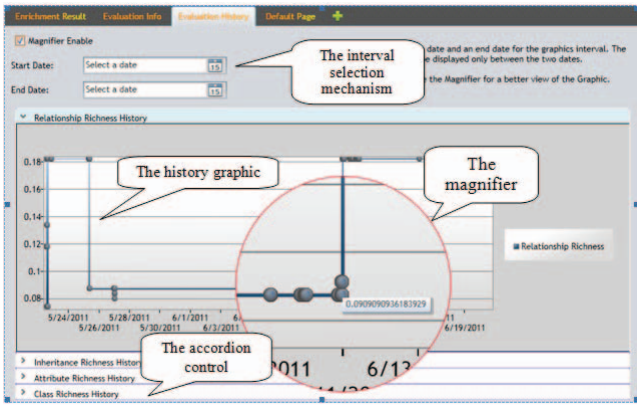

Fig. 5. History chart of ontology metrics

in the RDF/OWL format, displaying ontology relationships and instances in separate windows, adding concepts, roles, and instances to the ontology.

B. Ontology Enrichment

The OntoRich tool uses domain categorized web content extracted by our RSS Reader and sent to the user in the form of an e-mail. After having a document (or more) added in the corpus the user has several methods for text processing and term extraction. The first method implies using statistical methods described in section II-0a (see figure 2).

The existing concepts in the ontology are using together with the semantic power of WordNet in order to extracting 'partOf', 'membeOf', 'madeFrom' and 'isKindOf' relations. Every word displayed in the hyponym tree can be selected and added to the ontology as sub-class of a specified concept. Results for the IT considered ontology are shown in figure 3. Using Open NLP models the OntoRich system is able to discover well known categories of concepts or instances (person names, organization names or time references), but also possible instances that are represented by proper nouns identified in the text by POS tagging.

C. Ontology Evaluation

This module provides methods for evaluating the ontology as a whole or evaluating a specified class from the ontology. Each metric result is presented with the help of intuitive graphics and user aids, as depicted in figure 4. Ontology metrics evolution over time has been also included.

The user has the opportunity to store multiple evaluation results on the same ontology and then request for an evaluation chart in order to observe the changes that the ontology has subject to during a certain period (see figure 5). When an ontology is evaluated for several times, the system keeps information about the evolution of the ontology from the first time it was loaded by the system. This feature allows to create an evolution-based evaluation by showing how the metrics described above vary in time for an ontology. Another important feature of the evaluation component is the ability to compare the considered ontology with another ontology from the same domain. The two ontologies are evaluated and the results are presented in a comparative manner so that the user can decide which ontology is better for his own needs.

D. System Performance

Terms were found using statistical methods and NLP based methods. The changed ontology was successfully saved to its

original location. The term extraction process took about 10 seconds because the large amount of text content loaded in the text corpus. This delay is due to the amount of computation done in order to test each possible term against the input parameters given by the user (minimum appearance frequency and maximum number of words in a term). An improvement to this problem could be an approach in which extracted terms are returned to the user as they are found and not only after the whole term extraction process completed. Another conclusion was that the application can scale well for loading ontologies up to 1 MB in size but works harder when the size of the ontology goes over this threshold (see figure 6).

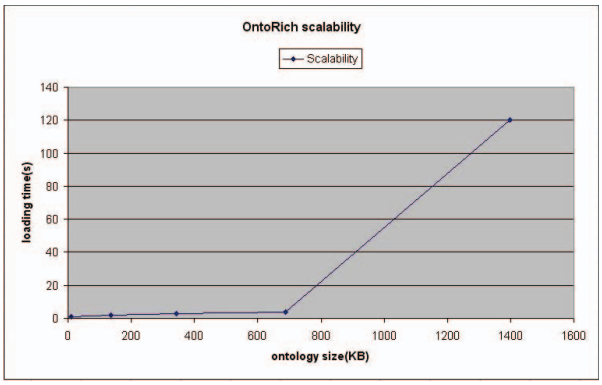

Fig. 6. OntoRich System scalability

A comparison between OntoRich and the four major existing systems for ontology enrichment and evaluation Kaon, Neon, OntoQA, ROMEO can be seen in table I.

Feature	Enrichment		Evaluation		OntoRich
	KAON	NEON	OntoQA	ROMEO	
Load Ontology	✓	✓	✓	✓	✓
Search Ontology Online	✗	✗	✗	✓	✓
Edit/Manage Ontology	✓	✓	✗	✗	✓
Add/Remove Concept	✓	✓	✗	✗	✓
Add/Remove Property	✓	✓	✗	✗	✓
Instantiate Properties	✓	✓	✗	✗	✗
Ontology Merging	✗	✓	✗	✗	✓
Query and Searching	✗	✓	✓	✓	✗
Automated Information Extraction	✗	✓	✗	✗	✓
Customized Information Extraction	✓	✗	✗	✗	✓
Instance Extraction	✓	✓	✗	✗	✓
Relation Extraction	✓	✓	✗	✗	✓
Relation Learning	✓	✓	✗	✗	✗
Ontology Matching	✗	✓	✗	✗	✗
Ontology Evaluation	✗	✓	✓	✓	✓
Metrics Definition	✗	✗	✗	✓	✓
Metrics Customization	✗	✗	✗	✗	✓
Customized Result Interpretation	✗	✗	✗	✓	✓
Graphic View of Results	✗	✗	✓	✓	✓
Evaluation History	✗	✗	✗	✗	✓

TABLE I
COMPARISON BETWEEN ONTORICH AND EXISTING SOFTWARE.

V. CONCLUSIONS

In this paper the main idea presented is that of using together a set of tools and methods already known in the domain of Semantic Web in order to create a powerful tool for both ontology enrichment and evaluation. An RSS Reader is the considered automatic web content extraction method. RSS feeds are an important source of information as they provide constantly updated web content. So, new instances of some already existing ontology are easily found within the content of domain specialized RSS feeds. In order to extract new concepts, relationships and instances for an ontology statistical methods were used. RiTa Wordnet API and OpenNLP API provided also an important backup. The WordNet ontology is used in order to examine and extract candidates for ontology enrichment taking advantage of various features such as word stems, word hyponyms or word meronyms, whilst with integration of the OpenNLP API the system gained access to syntactical analysis of a text in order to improve the quality of discovered terms in relation to the context where they appeared. Options for evaluating the ontology from the design point of view and also from the knowledge base perspective were added to the OntoRich system. Metrics for evaluating the entire ontology schema or for evaluating a specific classes from the ontology are provided. Comparative evaluation of the new ontology against the old one is also presented to facilitate the quality assessment of an ontology.

Ongoing work regards refinement of the ontology population algorithms and evaluation components. WordNet ontology can be exploited even more, and with the help of OpenNLP, relationships between concepts from the ontology or new domain concepts could be discovered even when the context of use causes word ambiguity.

ACKNOWLEDGMENT

We are grateful to the anonymous reviewers for their useful comments. This work was supported by the grant ID 160/672 from the National Research Council of Romanian Ministry of Education and Research, and POSDRU/89/1.5/S/62557/EXCEL.

REFERENCES

[1] T. C. Du, F. Li, and I. King, "Managing knowledge on the web - extracting ontology from html web," *Decision Support Systems*, vol. 47, no. 4, pp. 319–331, 2009.

[2] M. Vargas-Vera, J. Domingue, Y. Kalfoglou, E. Motta, and S. B. Shum, "Template driven information extraction for populating ontologies," in *Workshop on Ontology Learning*, ser. CEUR Workshop Proceedings, A. Maedche, S. Staab, C. Nedellec, and E. H. Hovy, Eds., vol. 38, 2001.

[3] G. Geleijnse and J. H. M. Korst, "Creating a dead poets society: Extracting a social network of historical persons from the web," in *ISWC/ASWC*, ser. Lecture Notes in Computer Science, K. Aberer, Ed., vol. 4825. Springer, 2007, pp. 156–168.

[4] S. Wang and E. Chen, "An instance learning approach for automatic semantic annotation," in *CIS*, 2004, pp. 962–968.

[5] J. Yu, J. A. Thom, and A. M. Tam, "Requirements-oriented methodology for evaluating ontologies," *Inf. Syst.*, vol. 34, no. 8, pp. 766–791, 2009.