Semantic annotation and retrieval of services in the Cloud.

Miguel Ángel Rodríguez-García[1], Rafael Valencia-García[1], Francisco García-Sánchez[1], J. Javier Samper-Zapater[2], and Isidoro Gil-Leiva[1]

[1]Departamento de Informática y Sistemas,

Universidad de Murcia. Campus de Espinardo 30100 Murcia, Spain

{miguelangel.rodriguez,valencia,frgarcia, isgil}@um.es

http://www.um.es

[2]Departament d'Informàtica,

Escola Tècnica Superior d'Enginyeria, Universitat de València, Avda. de la Universidad, s/n, 46100 Burjassot, Valencia (Spain)

jose.j.samper@uv.es

http://www.uv.es

Abstract. Recently, the economy has taken a downturn, which has forced many companies to reduce their costs in IT. This fact has, conversely, benefited the adoption of innovative computing models such as cloud computing, which allow businesses to reduce their fixed IT costs through outsourcing. As the number of cloud services available on the Internet grows, it is more and more difficult for companies to find those that can meet their needs. Under these circumstances, enabling a semantically-enriched search engine for cloud solutions can be a major breakthrough. In this paper, we present a fully-fledged platform based on semantics that (1) assist in generating a semantic description of cloud services, and (2) provide a cloud-focused search tool that makes use of such semantic descriptions to get accurate results from keyword-based searches. The proposed platform has been tested in the ICT domain with promising results.

1. Introduction

The future Internet will be based on services and this new trend will have significant impact on domains such as e-Science, education and e-Commerce. Consequently, the Web is evolving from a mere repository of information to a new platform for business transactions and information interchange. Large organizations are increasingly exposing their business processes through Web services technology for the large-scale de-

velopment of software, as well as for the sharing of their services within and outside the organization. New paradigms for software and services engineering, such as Software-as-a-Service (SaaS) and the cloud computing model, promise to create new levels of efficiency through large-scale sharing of functionality and computing resources.

Cloud computing is a technological paradigm that permits to offer computing services over the Internet [Zhang et al., 2010]. In the current socio-economic climate, the affordability of cloud computing has gained popularity among today's innovations. Under these circumstances, more and more cloud services become available. Consequently, it is becoming increasingly difficult for service consumers to find and access those cloud services that fulfill their requirements. Semantic approaches have proven to be very effective in improving search processes [Vidoni et al., 2011]. However, providing semantic descriptions for all the cloud solutions currently available on the Internet is a very time-consuming task. Natural language processing (NLP) tools can help in automating the translation of the existent cloud-related natural language descriptions into semantically equivalent ones. In this paper, we present a semantic-based platform to annotate and retrieve services in the cloud.

In last decade several semantic annotation systems have been developed. However, as of today there is still not a standard approach for semantic annotation [Uren et al., 2006]. For this reason, semantic annotation systems have been classified based on some parameters such as 'standard format', 'ontology support', 'document evolution' and 'automation' [Uren et al., 2006]. Concerning 'standard formats', several formats are recommended by the World Wide Web Consortium to build ontologies. The most extended formats in the context of semantic annotation are RDF, RDF Schema, and OWL. The two former formats are used by the following approaches Armadillo [Chapman et al., 2005], CREAM [Handschuh and Staab, 2003]. OWL, on the other hand, is supported by others tools such as CERNO [Kiyavitskaya et al., 2009], EVONTO [Tissaoui et al., 2011], and KIM [Popov et al., 2003]. The application proposed here is also based on OWL.

Additionally, one property that is often desired in the scope of semantic annotation is multiple ontologies support, which allows to expand the knowledge to cover different domains. There are several tools such as KIM, CREAM or Armadillo that have been developed to support the use of multiple ontologies. In contrast, CERNO, S-CREAM [Handschuh et al, 2002] or EVONTO do not include this feature.

In the annotation context, there are a number of constraints related to computational cost guiding the way to process multiples ontologies, as follows: (i) the ontologies that are to be used must be merged, or (ii) annotations have to explicitly declare to which ontology they refer. Given performance and computational costs constraints, it is more appropriate to have several mid-size ontologies than a big merged ontology. In fact, some techniques have been proposed that split huge ontologies into several modules to make them more manageable for computers [Cuenca-Grau et al., 2007].

A further interesting property of ontology-based systems is that of ontology evolution. It refers to the process of changing the ontologies over time by, for example, adding or modifying new classes or individuals, or removing knowledge and ensuring the consistency of the annotations against the ontologies that are being modified.

EVONTO, KIM, S-CREAM and CREAM implement an ontology evolution approach. Other semantic annotation tools such as CERNO or Armadillo do not cover this feature. In our work, support for both multiple ontology and ontology evolution is provided.

Almost all the current semantic annotation tools provide support for document evolution. For example, while Armadillo, CREAM, KIM and EVONTO update the annotations if a change is made in one or more documents, S-CREAM and CERNO do not.

Three kinds of semantic annotation systems can be distinguished: manual, fully automated and semi-automated. Manually annotating documents with semantic content is a very time-consuming task [Cravegna et al., 2002]. Therefore, the tendency is toward providing semi-automated tools within the current ontology-based annotation systems. Examples of this trend are CERNO and S-CREAM. There are also some fully-automated tools such as Armadillo, KIM and EVONTO.

The rest of the paper is organized as follows. The components that take part in the platform and their overall architecture are described in Section 2. In Section 3, a case use scenario in the information and communications technologies (ICT) domain and its evaluation is shown. Finally, conclusions and future work are put forward in Section 4.

2. Platform architecture

The focus of the work described here is the development of a fully-fledged application for the semantically-enhanced search of services in the cloud. The architecture of the proposed approach is shown in Fig. 1. The approach is based on three main modules: (i) the semantic annotation module, (ii) the semantic indexing module, and (iii) the semantic search engine. In a nutshell the system works as follows: First, natural language descriptions of the services in the cloud are semantically represented and annotated. Then, from these annotations a semantic index is created using the classic vector space model. Finally, a semantic search engine permits to retrieve the matching services from keyword-based searches. Next, these components are described in detail.

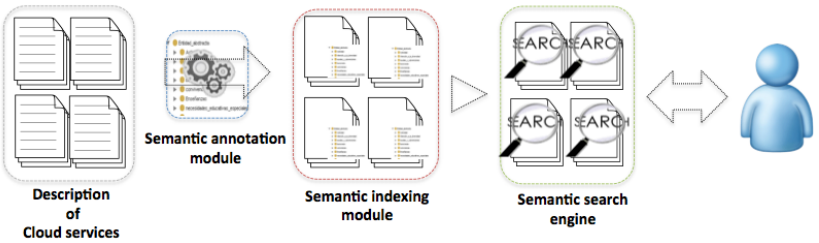

Fig. 1. System architecture

2.1. Semantic annotation module

This tool receives both domain ontologies and a natural language description of cloud services as inputs. Then, using a set of natural language processing (NLP) tools, it obtains a semantic annotation for the analyzed cloud services descriptions in accordance with the domain ontologies and Wordnet. This module is based on the methodology presented in [Valencia-García et al., 2008] and is composed of two main phases: the NLP phase and the semantic annotation phase.

The main aim of the NLP stage is the extraction of the morphosyntactic structure of each sentence. For this purpose, a set of NLP software tools, including a sentence detection component, a tokenizer, a set of POS (Part-Of-Speech) taggers, a set of lemmatizers and a set of syntactic parsers, have been developed. The GATE framework[1] has been employed to build some of the components required for the NLP phase. GATE is an infrastructure for developing and deploying software components that process human language.

During the second phase, the cloud services descriptions are annotated with the classes and individuals of the domain ontologies by following the process described next. First, the most important linguistic expressions are identified by means of statistical approaches based on the syntactic structure of the text. Then, for each linguistic expression, the system tries to determine whether such expression is an individual of a class of the domain ontology.

The outcome of the semantic annotation module is a list of semantic annotations defined in terms of the ontology. The classes and individuals in the annotations represent terms that have been extracted from the cloud services descriptions.

2.2 Semantic indexing module

In this module, the system retrieves all the annotated knowledge from the previous module and tries to create fully-filled annotations with this knowledge. This step is based on the work presented in [Castells et al., 2007]. Each annotation of each document is stored in a database and has a weight assigned. The annotation weight reflects how relevant the ontological entity is for the document meaning. Weights are calculated by using the TF-IDF algorithm [Salton and McGill, 1983], which uses the following equation (see equation 1).

$$(tf - idf)_{i,d} = \frac{n_{i,d}}{\sum_k n_{k,d}} \times \log \frac{|D|}{N_i} \qquad (1)$$

where ni,d is the number of occurrences of the ontological entity i in the document d, $\sum_k n_{k,d}$ is the sum of the occurrences of all the ontological entities identified in

[1] http://gate.ac.uk/

the document d, |D| is the set of all documents and Ni is the number of all documents annotated with i.

In this scenario, the cloud services descriptions are the documents to be analyzed. For each description, an index is calculated based on the adaptation of the classic vector space model presented in [Castells et al., 2007]. Each service is represented as a vector in which each dimension corresponds to a separate ontological concept of the domain ontology. The value of each ontological concept dimension is calculated as follows (see equation 2).

$$(v_1, v_2, \dots, v_n)_d \; where \; v_i = \sum_{j=1}^{n} \frac{tf - idf_{j,d}}{e^{dist(i,j)}} \qquad (2)$$

where dist(i,j) is the semantic distance between the concept i and concept j in the domain ontology. This distance is calculated by using the taxonomic (subclass_of) relationships of concepts in the domain ontology. So, the distance between a concept and itself is 0, the distance between a concept and its taxonomic parent or child is 1 and so on.

The outcome of the semantic indexing module is a list of semantic concepts sorted according to equation 2. Each assigned value represents both the relevance of the corresponding concept in all the analyzed descriptions and its relationships with other concepts in the domain ontology.

2.3 Semantic search engine

This module is responsible for finding services in the cloud from a keywords-based query. This process takes advantage of the semantic content and annotations previously gathered by the system.

First, users introduce a series of keywords and the system identifies which concepts in the domain ontology are referred by them. As it has been explained in the previous section, each service is represented as a vector in which each dimension corresponds to a separate concept of the domain ontology. Then, the semantic search engine calculates a similarity value between the query q and each service s. In order to do that, the cosine similarity is used (see equation 3):

$$sim(q,s) = cos\theta = \frac{q \times s}{|q| \times |s|} \qquad (3)$$

A ranking of the most relevant cloud services that are related to the topics referenced in the query is then defined by using the similarity function showed in equation 3. The ‘s’ vector is the one calculated by equation 2 for each service description. The

'q' vector, on the other hand, is the one created from the concepts extracted from the search engine query. The θ symbol represents the angle that separates both vectors, and describes the similitude grade between two documents.

3. Case use: Annotation and retrieval of ICT services in the Cloud

The platform described in the previous section has been implemented and tested in the ICT domain. For this, in the first place, an ontology of the ICT domain has been developed. Next, around 100 different services with their description in natural language have been selected to be annotated by the system.

3.1 ICT ontology

In this work, ontologies that semantically describe the functional properties of ICT applications have been studied. A representative example within this area is shown in [Lasheras et al., 2009], where an OWL (Web Ontology Language) ontology for requirements specification documents is developed and used for modeling reusable security requirements. The semantic description of the functionality of software components has been addressed in [Happel et al., 2006]. Here, the KOntoR system allows semantic descriptions of components to be stored in a knowledge base and semantic queries to be run on it (using the SPARQL language). The OWL ontology-based DESWAP system is presented in [Hartig et al., 2008]. In the context of this project, a knowledge base with comprehensive se-mantic descriptions of software and its functionalities was developed. Thus, by taking into account the shortcomings of developing a new ontology from scratch, the ontologies developed under the scope of the DESWAP project have been reused in this work to represent the features and functional properties of software projects.

3.2 Evaluation

During a first stage, representatives of an ICT organization are required to input a set of interesting services in the cloud with their descriptions. Then they are semantically annotated and stored in the ontology repository. The Sesame RDF repository, backed up by a MySQL database, has been used to implement the ontology repository.

Once the semantic indexes have been created, the experiment starts. This experimental evaluation aims at elucidating whether the semantic search engine module of the proposed platform is useful. Ten topic-based queries were issued. For each query, a set of cloud services was manually selected. At the same time, the semantic search

engine was asked to perform the same task, in an automatic way. These results were then compared to those produced by the manual selection.

The average time taken for the human expert to complete each search throughout the cloud services repository, which contains 106 services, was 180,98 seconds. In contrast, the tool proposed in this paper executed each query at a rate of 0,78 seconds.

The final results of the experiment are shown in Table 1. The system obtained the best scores for queries of the topic "Databases", with a precision of 0.92, a recall of 0.89, and a F1 measure of 0.90. In general, the system obtains better results in precision (88% on average) than the results of recall (82% on average). Hence, these results are promising.

Table 1. Precision, recall and F1 of the experiment.

Topics	Precision	Recall	F1
J2EE	0,89	0,81	0,85
application server	0,82	0,76	0,79
Databases	0,92	0,89	0,90
Enterprise information systems	0,9	0,83	0,86

4. Discussion and Conclusion

Semantic annotation and retrieval of cloud services is a challenging task and addresses the issue of finding the service or services with the functionality that meets the users' needs. In this paper, a semantic platform for the annotation of cloud services from their natural language descriptions and their retrieval from key-word based searches has been proposed. The system presented here automatically annotates different cloud services from their natural language description, which can be available in a number of document formats such as XML, HTML or PDF. Besides, the proposed platform has been implemented taking into account a multiontology environment (with OWL 2 ontologies) to be able to cope with several domains. Moreover, it supports the evolution of the source documents, thus maintaining the coherence between the natural language descriptions and the annotations, which are stored using a semantic Web-based model.

An experiment has been carried out with the objective of checking whether the system is useful for semantically annotating and retrieving services in the cloud. The results of the experiment are promising. However, they do not reflect the actual potential of the approach, since the experiment has been performed at a very small scale. Thus, a more complete and thorough validation of the system is planned by applying the system to a larger set of services and by using statistical methods for analyzing the results obtained.

Several issues remain open for future work. So far, the services have been analyzed by exploring their natural language descriptions. It could be beneficial to also use semantic information about their functionality by using ontologies that can de-

scribe these services as shown in [Ortegón-Cortázar et al., 2012]. Additionally, we are currently working on upgrading this system and converting it into a recommendation system in which users could set their preferences and the system would return only those services that are relevant to them in a particular domain. Finally, we plan to study the possibility of offering a search service also including an opinion mining engine, such as the one presented in [Peñalver-Martínez et al., 2011], which permits to obtain the sentimental classification of the services in order to provide information about their non-functional properties.

Acknowledgements: This work has been supported by the Spanish Ministry for Science and Innovation and the European Commission (FEDER / ERDF) through project SeCloud (TIN2010-18650).

References

1. Castells, Pablo et al (2007) 'An adaptation of the vector-space model for ontology-based information retrieval'. IEEE Transactions on Knowledge and Data Engineering, 19(2), 261–272.
2. Chapman Sam et al (2005) 'Armadillo: Integrating knowledge for the semantic web'. In Proceedings of the Dagstuhl Seminar in Machine Learning for the Semantic Web.
3. Ciravegna, Fabio et al (2002) 'User-system cooperation in document annotation based on information extraction'. In Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management, EKAW02. Springer Verlag.
4. Cuenca-Grau, Bernardo et al (2007) 'Extracting Modules from Ontologies: A Logic-based Approach'. Proc. of the 3rd OWL Experiences and Directions Workshop, n 258 in CEUR.
5. Handschuh Siegfried et al (2002) 'S-CREAM – Semi-automatic CREAtion of Metadata'. The 13th International Conference on Knowledge Engineering and Management (EKAW 2002), ed. Gomez-Perez, A., Springer Verlag.
6. Handschuh, Siegfried & Staab, Steffen (2003) 'Cream: Creating metadata for the semantic web'. Comput. Networks, 42(5): 579–598.
7. Happel, Hans-Jörg et al (2006) 'KOntoR: An Ontology-enabled Approach to Software Reuse'. Proc. of the 18th Int. Conf. on Software Engineering and Knowledge Engineering (SEKE), San Francisco.
8. Hartig, Olaf et al (2008) 'Designing Component-Based Semantic Web Applications with DESWAP'.Proceedings of the Poster and Demonstration Session at the 7th International Semantic Web Conference (ISWC), Karlsruhe, Germany.
9. Kiyavitskaya, Nadzeya, et al (2009) 'Cerno: Light-weight tool support for semantic annotation of textual documents'. Data Knowl. Eng. 68(12): 1470-1492.
10. Lasheras, Joaquín et al (2009) 'Modelling Reusable Security Requirements based on an Ontology Framework'. Journal of Research and Practice in Information Technology, 41(2): 119-133.
11. Ortegón-Cortázar, Giovanni et al (2012) 'Adding semantics to cloud computing to enhance service discovery and access'. In Proceedings of the 6th Euro American Conference on Telematics and Information Systems (EATIS '12), Rogerio Patricio Chagas do Nascimento (Ed.). ACM, New York, NY, USA, 231-236. DOI=10.1145/2261605.2261639 http://doi.acm.org/10.1145/2261605.2261639

12. Popov, Borislav et al (2003) ‘KIM - Semantic Annotation Platform’. In: Proceedings of the 2nd International Semantic Web Conference. doi:10.1007/b14287.
13. Peñalver-Martínez, Isidro et al (2011) 'Ontology-guided approach for Feature-Based Opinion Mining', NLDB 2011 pp 193-200 Alicante Spain.
14. Salton, Gerald & McGill, Michael J. (1983) ‘Introduction to modern information retrieval’. McGraw-Hill. ISBN 0070544840.
15. Tissaoui, Anis et al (2011) ‘EVONTO: Joint evolution of ontologies and semantic annotations’. (short paper). Dans: International Conference on Knowledge Engineering and Ontology Development (KEOD 2011), Jan Dietz (Eds.), INSTICC - Institute for Systems and Technologies of Information, Control and Communication, p. 1-6.
16. Uren, Victoria et al (2006) ‘Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art’, Journal of Web Semantics, 2006, 4(1): 14-28.
17. Valencia-García, Rafael et al (2008) ‘A knowledge acquisition methodology to ontology construction for information retrieval from medical documents’. Expert Systems, 25(3): 314–334.
18. Vidoni, Renato et al (2011) ‘An intelligent framework to manage robotic autonomous agents’. Expert Systems with Applications, 38(6): 7430-7439.
19. Zhang, Qi et al (2010) ‘Cloud computing: state-of-the-art and research challenges’. J Internet Serv Appl 1(1): 7-18.