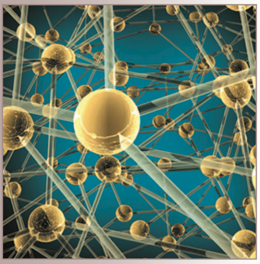

Semantic Heterogeneity Issues on the Web

To operate effectively, the Semantic Web must be able to make explicit the semantics of Web resources via ontologies, which software agents use to automatically process these resources. The Web's natural semantic heterogeneity presents problems, however — namely, redundancy and ambiguity. The authors' ontology matching, clustering, and disambiguation techniques aim to bridge the gap between syntax and semantics for Semantic Web construction. Their approach discovers and represents the intended meaning of words in Web applications in a nonredundant way, while considering the context in which those words appear.

Jorge Gracia
Universidad Politécnica de Madrid

Eduardo Mena
University of Zaragoza

Although it's far from fully deployed, the Semantic Web constitutes an enormous source of distributed and heterogeneous information, and is evolving quickly as users add new knowledge. To realize the Semantic Web vision, we must clearly define Web resources' semantics using ontologies. However, expecting the volume of annotated resources to reach the critical mass that the Semantic Web requires via manual annotation alone is unrealistic. Rather, we need methods that can automatically determine the semantics of textual resources on the Web.

The availability of online ontologies and semantic content benefits Web interoperability. Because of the Web's open nature, however, online semantics can be defined by different people, for different domains, and can vary significantly in expressiveness, richness, coverage, and quality, leading to increasing semantic heterogeneity. This leads to various issues, the most significant of which are

- *semantic ambiguity*, in which many intended meanings are associated with the same word; and
- *semantic redundancy*, in which many semantic descriptions are available to represent the same intended meaning.

These two problems can hamper Semantic Web applications when they must determine the correct meaning of certain textual resources (data, keywords, entities, and so on) using online ontologies.

Here, we discuss a set of techniques that can help reduce such redundancy and ambiguity issues. In particular, using these techniques in combination lets us process any word in a Web context whose sense we must discover by

1089-7801/12/$31.00 © 2012 IEEE

retrieving its set of possible senses, expressed as concise ontology terms, and indicating which one is the most relevant for the current situation. We focus specifically on discovering the meaning of words in unstructured texts (such as search keywords or folksonomy tags). Such a scenario maximizes the ambiguity problems due to a lack of syntactical or structural information in context, preventing us from applying traditional disambiguation methods.

Semantic Heterogeneity

Imagine we want to query the Semantic Web as follows: "Give me a list, ordered by calories, of recipes containing apple." This problem has two dimensions: semantic querying, or how to clearly specify this query's semantics so that a software agent can understand it; and semantic annotation, or how to add semantic content to the Web beforehand to satisfy such queries. We don't discuss the details of semantic querying and annotation here, but in both cases the semantics of the involved terms must be clearly defined for the query to be answered successfully.

In our example, only when the semantics of "apple" are clearly defined in the query — by grounding it into a certain ontology term (see http://dbpedia.org/resource/Apple) — can a software agent process the query to retrieve Web data that contains the same term (such as recipes from a webpage in which the word "apple" has been semantically annotated with http://dbpedia.org/resource/Apple). The sense-selection problem gets even more complicated because "apple" is polysemous and can be interpreted, depending on the context, as "a fruit," "a tree," or "a company," for instance. Thus, determining its intended meaning is necessary to provide a suitable semantic description. This exemplifies the ambiguity problem. Furthermore, given a particular interpretation of "apple" (for example, as "a fruit"), the word might be annotated with different but semantically equivalent ontology terms in websites or datasets about recipes. This illustrates the redundancy problem.

In our example, we must determine the most suitable meaning of the word "apple" in its context (a Web search about recipes). In a Semantic Web-based scenario, we can search online ontologies to get possible semantic descriptions for "apple." Nevertheless, we must deal with the term's ambiguity and try to select, among all possible semantic descriptions, the one that best represents the intended meaning. Deciding which sense is the intended one is relatively easy for humans — we simply inspect the context in which the ambiguous word appears. If "apple" appears as a cooking ingredient on a webpage about recipes, it likely refers to "a fruit," whereas if it's in a document about electronic equipment, it probably refers to the company. This selection task is difficult for a computer program, however. To help computers decide a word's correct meaning, we can apply word sense disambiguation (WSD) techniques, which try to pick out the most suitable sense of an ambiguous word according to the context (usually the surrounding text) in which it appears.

As mentioned, we can access online ontologies to discover possible semantic descriptions for "apple." However, this approach might return many redundant terms. For instance, at the time of writing, the Swoogle Semantic Web search engine (http://swoogle.umbc.edu) retrieved 445 different ontology terms associated with "apple." Obviously, this number is considerably higher than this word's real polysemy. This problem can hamper the disambiguation task because we must analyze and choose from all 445 terms, even though a simple inspection confirms that most of them fall into one of the three possible interpretations we mentioned previously.

We can solve this redundancy problem by conducting a sense clustering process that groups terms referring to the same meanings. In addition to reducing the number of meanings to facilitate disambiguation algorithms, this process also allows for richer integrated semantic descriptions that combine information from different sources.

Techniques for Semantic Heterogeneity Reduction

Our proposed approach (see Figure 1) is grounded in a study of semantic measures that numerically compute the degree of similarity and relatedness among different semantic descriptions. To overcome redundancy and ambiguity on the Semantic Web, we've developed a set of techniques based on these measures: *ontology matching*, *sense clustering*, and *sense disambiguation*.

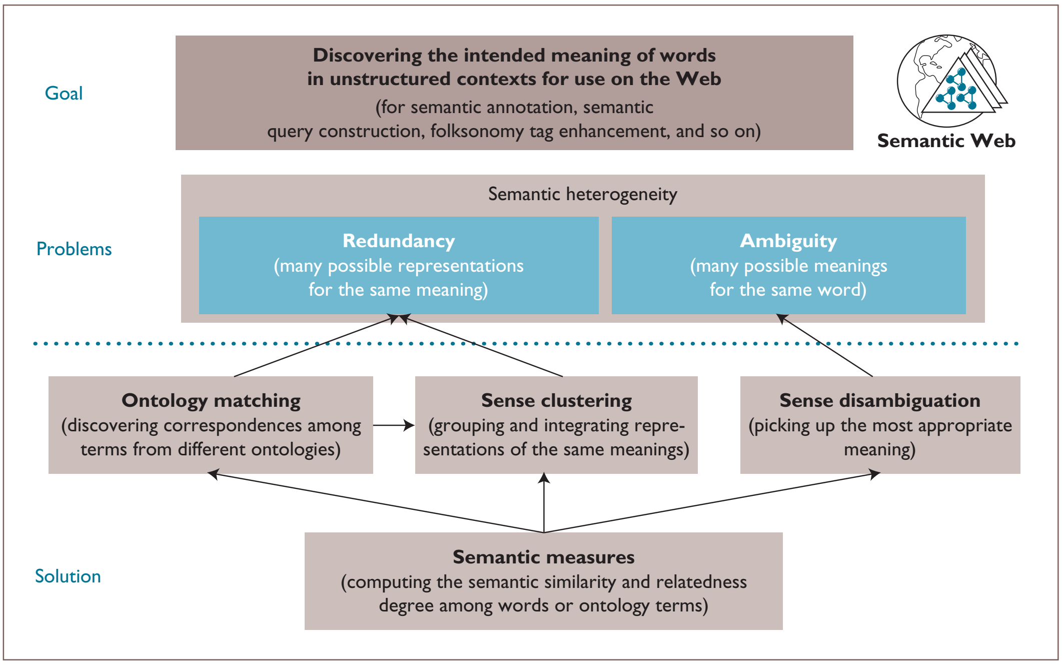
Goal
Discovering the intended meaning of words in unstructured contexts for use on the Web
(for semantic annotation, semantic query construction, folksonomy tag enhancement, and so on)
Semantic Web
Problems
Semantic heterogeneity
Redundancy
(many possible representations for the same meaning)
Ambiguity
(many possible meanings for the same word)
Ontology matching
(discovering correspondences among terms from different ontologies)
Sense clustering
(grouping and integrating representations of the same meanings)
Sense disambiguation
(picking up the most appropriate meaning)
Solution
Semantic measures
(computing the semantic similarity and relatedness degree among words or ontology terms)

Figure 1. Techniques for addressing semantic heterogeneity issues on the Web. These techniques are based on semantic measures and help solve the redundancy and ambiguity issues that heterogeneity introduces.

Semantic Measures

Semantic measures evaluate how semantically related two terms are. *Semantic relatedness measures* account for all possible semantic relationships, whereas *semantic similarity measures* consider only those relationships that involve similitude or likeness between the two compared terms.[1] Based on these definitions, we can create various semantic measures suitable for use on the Web. Such measures should have certain characteristics:

- *Maximum coverage.* In many scenarios, we don't know in advance what users have in mind when they choose certain words to interact with Web applications. To maximize the chances of inferring the correct user meanings, semantic measures should consider as many interpretations of the measured terms or words as possible, relying not on a single knowledge source or annotated corpus but on as many as possible.
- *Dynamic knowledge selection.* Expecting to treat all the accessible knowledge on the Semantic Web as local is unrealistic. On the contrary, semantic measures should be able to work among any dynamically discovered ontology term, coming from any pool of online or local ontologies.
- *Universality.* Semantic measures, in the Web's highly dynamic context, should be defined independent of their final application.

These features have motivated the design principles we adopt to define semantic measures. Our proposal reuses some previous work in the field,[1–3] but adds functionalities such as using the Web as a corpus for relatedness computation and applying lightweight inference for schema-based similarity computation. We propose two types of measures.

Context- and inference-based semantic similarity. This measure combines different techniques to compare the ontological context of two terms — that is, labels, hyper/hyponyms, domains, roles, and so on. In addition, we apply semantic reasoning techniques to these ontological contexts to give rise to inferred facts that aren't present in the asserted ontologies. After extracting the ontological contexts, this approach compares them by combining different elementary techniques such as linguistic similarities and vector space modeling.[4] The result is a value in [0,1] representing the similarity of the compared ontology terms' contexts. We compare two entities a and b as follows:

$$\begin{aligned} sim(a,b) = {} & w_{label} \cdot sim_{str}(a^{syn}, b^{syn}) \\ & + w_{descr} \cdot vsm(a^{descr}, a^{descr}) \\ & + w_{attr} \cdot vsm(a^{attr}, b^{attr}) \\ & + w_{sup} \cdot vsm(a^{sup}, b^{sup}) \\ & + w_{sub} \cdot vsm(a^{sub}, b^{sub}), \end{aligned} \tag{1}$$

where x^{syn}, x^{descr}, x^{attr}, x^{sup}, and x^{sub} denote the set of synonym labels, the textual description, the set of attributes characterizing an entity x (that is, properties if x is a class; domains and ranges if x is a property; and associated classes and property values if x is an individual), and the hierarchical graph's super- and subterms, respectively. Here, vsm is a comparison based on vector space models; sim_{str} is a string-based similarity; and w_i is the weight of the ith component. In our prototype, we inferred such weights empirically after experimenting with the Ontology Alignment Evaluation Initiative (OAEI; http://oaei.ontologymatching.org) benchmark track and chose the weights that led to the best performance.

Web-based semantic relatedness. To compute a Web-based relatedness measure between plain words, we chose the normalized Google distance (NGD), a well-founded semantic measure.[2] NGD uses the Web as a knowledge source and is based on counting the co-occurrence of words on webpages. Relatedness between two words x and y is given by

$$relWeb(x, y) = e^{-2NWD(x,y)}, \tag{2}$$

where NWD is a generalization of NGD to any Web search engine. Based on Equation 2, we define a relatedness measure between ontological terms a and b as follows:[5]

$$\begin{aligned} rel_0(a,b) &= \frac{\sum_{i,j} relWeb(a_i^{syn}, b_j^{syn})}{|\, a^{syn} \,| \cdot |\, b^{syn} \,|} \\ rel_1(a,b) &= \frac{\sum_{i,j} rel_0(a_i^{ctx}, b_j^{ctx})}{|\, a^{ctx} \,| \cdot |\, b^{ctx} \,|} \\ rel(a,b) &= w_0 \cdot rel_0(a,b) + w_1 \cdot rel_1(a,b) \end{aligned} \tag{3}$$

where x^{syn} and x^{ctx} are the set of synonym labels and the minimum ontological context of x, respectively; x^{ctx} contains direct superclasses, domains, or associated classes (if x is a class, property, or instance, respectively); and w_i are empirically inferred weights ($w_0 = w_1 = 0.5$ in our prototype).

These measures fulfill the previously mentioned requirements for operating on the Web. Let's look at how they're applied to the tasks identified in Figure 1.

Ontology Matching

Ontology matching is the task of determining relationships among terms from two different ontologies. The Context and Inference-based Ontology Aligner (Cider) is an ontology-matching tool for discovering semantic equivalence relationships.[6] Cider inputs are ontologies expressed in OWL or RDF. When it receives input, Cider extracts the compared terms' ontological context and applies a lightweight inference mechanism to add semantic information that isn't explicit in the asserted ontologies. Next, it computes semantic similarities between each pair of terms using Equation 1 and obtains a matrix with all the similarities. Finally, it extracts the final alignment, finding the highest-rated one-to-one relationships among terms and filtering out those below a certain threshold. In addition to aligning whole ontologies, Cider can also serve individual similarity computations and is thus easily adaptable to other uses, such as sense clustering.

Sense Clustering

To tackle redundancy reduction on the Semantic Web, we define a clustering technique that we apply to the base of ontological terms collected by Watson (a system that crawls the Web and indexes available semantic resources[7]) and creates groups of ontological terms having similar meanings. Step 1 initially groups all ontology terms we can find in Watson that are associated with the same keywords (or synonym labels). We call these sets of ontology terms *synonym maps*.

Step 2 is extraction and similarity computation. An iterative algorithm takes each ontology term from a synonym map and computes its similarity degree (Equation 1) with respect to the other terms in the map. Step 3 is integration. When the obtained similarity value

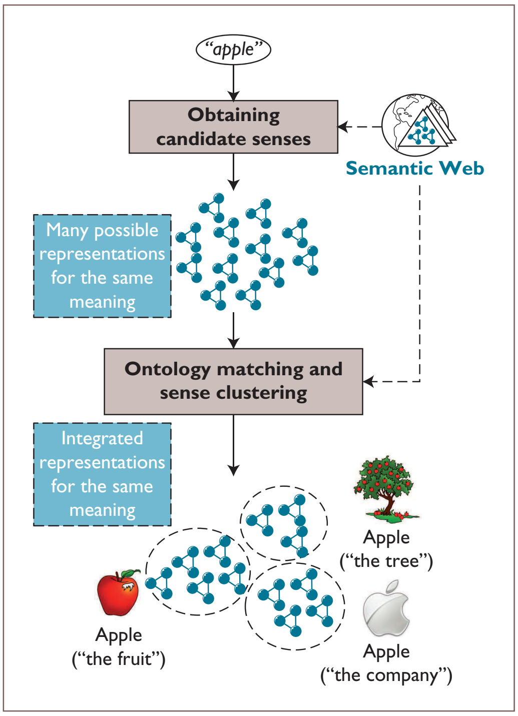
"apple"
Obtaining candidate senses
Semantic Web
Many possible representations for the same meaning
Ontology matching and sense clustering
Integrated representations for the same meaning
Apple ("the tree")
Apple ("the fruit")
Apple ("the company")

Figure 2. Clustering example. The sense clustering method, applied to a search of "apple," returns all the ontology terms that refer to the meanings "the fruit," "the tree," and "the company," grouped together as three single integrated senses.

is less than a given threshold, we consider both to be different senses, and the algorithm continues comparing other terms. If, on the contrary, the similarity is high enough, the algorithm integrates both terms into a single sense, and the comparison process is reinitiated among the new integrated sense and the rest of terms in the synonym map. We discuss a method for selecting a suitable threshold elsewhere.[8]

Steps 2 and 3 constitute an agglomerative clustering algorithm that produces, for each synonym map, a set of integrated senses (called *sense maps*) as output. The clustering process repeats with the remaining synonym maps to ultimately create a pool of integrated senses that covers all ontology terms in the Watson indexes. Each sense map groups the ontology terms that correspond to the same intended meaning.

Figure 2 illustrates the sense clustering process. The method, applied to a search of "apple," returns all the ontology terms that refer to the meanings "the fruit," "the tree," and "the company," grouped together as three single integrated senses, respectively.

Sense Disambiguation

Figure 3 illustrates how we can solve the ambiguity problem by applying a disambiguation technique that explores the context words (such as "recipe, calories") and determines the ambiguous word's possible senses to deduce the latter's correct meaning in that context.

We focus on disambiguation in unstructured Web contexts, such as sets of user keywords or user tags, where traditional WSD techniques have difficulty operating. In fact, user tags themselves constitute a highly heterogeneous context (usually free text) that hampers disambiguation. Because tags aren't in well-formed sentences, we can't apply syntactic analysis and similar techniques. Furthermore, many tags refer to users' subjective impressions (such as "my favorite" or "amazing") or technical details (for example, "Nikon" or "photo"), which leads to contexts in which many words are useless (or even harmful) for disambiguation.

Figure 4 illustrates the overall disambiguation process, used in combination with the sense clustering technique. The system receives a keyword and its context words as input and gives its most suitable sense as output. The first step in this process is context selection. We hypothesize that the most significant words in the disambiguation context are those most highly related to the word we want to disambiguate. Based on this, we compute Equation 2 between each context word and the keyword for disambiguation, filtering out the context words that score below a certain threshold. We empirically infer this threshold, which depends on the search engine used to compute Equation 2 (we use a 0.22 threshold for Yahoo in our current prototype). The resultant subset of context words is called *active context*.

After context selection, the system accesses online and local resources to provide a set of candidate senses for the keyword. This output describes the possible meanings of the keyword we want to disambiguate. As a result of applying the sense clustering process, each sense corresponds to an ontology term or to the integration of various ontology terms.

Finally, the system runs a disambiguation algorithm and weight the senses according to their likeliness of being the most suitable for the given context.[9] The disambiguation algorithm explores the semantic relatedness among the keyword senses and the words in the context, the overlap between the words appearing in the context and the words that appear in the sense's semantic definition,[3] and how frequently each sense is used according to annotated corpora or semantic data (if available).

Experiments

We evaluated our techniques to assess how well they performed their respective tasks. For instance, the Cider system participated in OAEI,[6] where it performed well in the benchmark track (97 percent precision and 62 percent recall, which is considerably higher than the 43 percent precision and 59 percent recall of the string matching-based baseline). The results in the directory track were the second best in the competition that year (60 percent precision and 38 percent recall).

We also studied our proposed large-scale method for ontology terms.[8] Our intention was to confirm empirically whether the method scales up and is feasible when applied to Watson. We answered this question positively after applying our technique to a pool of 73,169 ontology terms, obtaining a strong linear dependence of the time response with respect to the keyword maps' size (linear correlation coefficient $R = 0.97$).

Additionally, we explored our disambiguation algorithm's behavior to determine the sense of a set of ambiguous words associated with 350 pictures on Flickr, comparing them to human opinion.[9] The resultant 58 percent accuracy beat both the random and the "most frequent sense" (MFS) baselines in this experiment (20 percent and 43 percent accuracy, respectively). The state of the art indicates that nonsupervised disambiguation techniques rarely score higher than the MFS baseline, so this is a remarkable achievement.

Our semantic measures clearly perform well for the different tasks we've considered. Additionally, the principles in which their design is based let us use them with any ontology no matter the domain, and no matter whether the ontologies were predefined or discovered at runtime. More details about these experiments and their results are available at http://sid.cps.unizar.es/SEMANTICWEB/EXPERIMENTS/.

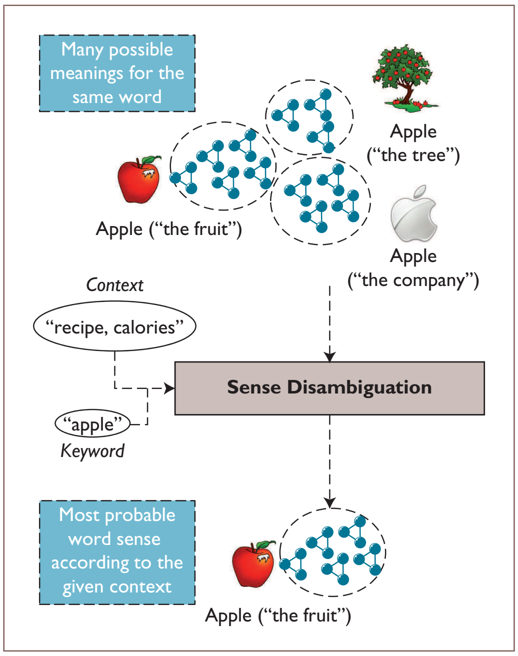

Figure 3. Disambiguation example. We can apply this disambiguation technique to explore context words and determine an ambiguous word's possible senses to deduce its correct meaning.

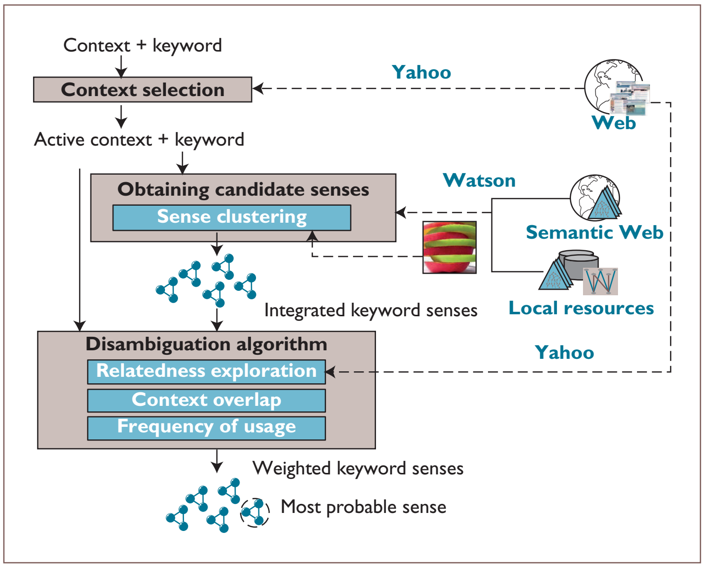

Figure 4. Disambiguation scheme. The system receives a keyword and its context words as input, gets possible candidate senses for the keyword, and uses the most significant words in the context (active context) to disambiguate. Finally, the most probable sense for the keyword in the given context is returned.

Related Work in Redundancy Reduction and Sense Discovery

To tackle the redundancy problem we discuss in the main text, some researchers have proposed a coreference resolution service (CRS)[1] in the context of linked data.[2] This service aims to determine equivalent Web identifiers (URIs) that refer to the same concept or entity. The system is targeted at storing, manipulating, and reusing coreference information. It stores equivalent identifiers in bundles, similar to how we store them in sense maps using our approach. CRS also treats the knowledge regarding coreference and equivalence separately from the semantic data sources, thus avoiding overusing the `owl:sameAs` clause to identify duplicate entities. Nevertheless, CRS doesn't propose specific algorithms to identify semantically equivalent groups of ontology terms, as we do. Furthermore, our approach deals with ontology terms, whereas CRS deals with resource identifiers, whether or not they constitute semantic descriptions.

PowerAqua, a Semantic Web-based question-answering system,[3] shares our aim to discover words' semantics in a Web-based context. It receives a simple question posed in natural language as input and obtains the data that satisfy this question as output. It first analyzes the input question linguistically, transforming it into a set of possible query triples. Then, it accesses Watson[4] to retrieve online semantic documents describing the involved entities. PowerAqua applies a mapping algorithm to find the most suitable correspondence between the query triples and the information in the candidate ontologies, finally deriving the searched information. Both our approach and PowerAqua exploit knowledge from dynamically selected online ontologies. Their respective inputs are different, however: whereas PowerAqua processes well-formed sentences, we deal with keywords in unstructured contexts, and don't rely specifically on linguistic analysis.

References

1. H. Glaser, A. Jaffri, and I. Millard, "Managing Coreference on the Semantic Web," *Proc. Linked Data on the Web Workshop* (LDOW 09), vol. 538, CEUR-WS, Apr. 2009.
2. C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data — the Story So Far," *Int'l J. Semantic Web and Information Systems* (IJSWIS), vol. 5, no. 3, 2009, pp. 1–22.
3. V. López et al., "PowerAqua: Supporting Users in Querying and Exploring the Semantic Web Content," *Semantic Web J.*, 2011; http://iospress.metapress.com/content/f2346411255409u5/fulltext.pdf.
4. M. d'Aquin et al., "Characterizing Knowledge on the Semantic Web with Watson," *Proc. 5th Int'l Evaluation of Ontologies and Ontology-Based Tools Workshop* (EON 07), 2007; http://km.aifb.kit.edu/ws/eon2007/EON2007_Proceedings.pdf.

For the sake of semantic interoperability and more precise Web data recovery, we must deal with semantic heterogeneity issues on the Web. Although redundancy and ambiguity are treated locally in particular domains and systems, few overall strategies exist for solving these issues when dynamically harvesting the Semantic Web. Our approach is a step in that direction, intended to enable expressing concisely the meaning of terms that appear in unstructured contexts on the Web. Discovering the meaning of keywords can assist semantic query construction, semantic webpage annotation, semantic classification of tagged resources, and so on. Our future work will focus on creating and enriching these kinds of systems, facilitating a practical realization of the Semantic Web.

References

1. A. Budanitsky and G. Hirst, "Evaluating WordNet-Based Measures of Lexical Semantic Relatedness," *Computational Linguistics*, vol. 32, no. 1, 2006, pp. 13–47.
2. R.L. Cilibrasi and P.M.B. Vitányi, "The Google Similarity Distance," *IEEE Trans. Knowledge and Data Eng.*, vol. 19, no. 3, 2007, pp. 370–383.
3. S. Banerjee and T. Pedersen, "Extended Gloss Overlaps as a Measure of Semantic Relatedness," *Proc. 18th Int'l Joint Conf. Artificial Intelligence*, 2003, pp. 805–810.
4. V.V. Raghavan and M.S.K. Wong, "A Critical Analysis of Vector Space Model for Information Retrieval," *J. Am. Soc. for Information Science*, vol. 37, no. 5, 1986, pp. 279–287.
5. J. Gracia and E. Mena, "Web-Based Measure of Semantic Relatedness," *Proc. 9th Int'l Conf. Web Information Systems Eng.* (WISE 2008), LNCS 5175, Springer, 2008, pp. 136–150.
6. J. Gracia and E. Mena, "Ontology Matching with CIDER: Evaluation Report for the OAEI 2008," *Proc. 3rd Ontology Matching Workshop* (OM 08), vol. 431, CEUR-WS, 2008, pp. 140–146.
7. M. d'Aquin et al., "Characterizing Knowledge on the Semantic Web with Watson," *Proc. 5th Int'l Evaluation of Ontologies and Ontology-Based Tools Workshop* (EON 07), 2007; http://km.aifb.kit.edu/ws/eon2007/EON2007_Proceedings.pdf.
8. J. Gracia, M. d'Aquin, and E. Mena, "Large Scale Integration of Senses for the Semantic Web," *Proc. 18th*

Int'l Conf. World Wide Web (WWW 09), ACM, 2009, pp. 611–620.

9. J. Gracia and E. Mena, "Multiontology Semantic Disambiguation in Unstructured Web Contexts," *Proc. Workshop on Collective Knowledge Capturing and Representation* (CKCaR 09), 2009; www.uni-koblenz.de/confsec/CKSM09/ckcar09_submission_1.pdf.

Acknowledgments

This work was supported by CICYT (Comisión Interministerial de Ciencia y Tecnología) projects TIN2010-21387-C02-02 and TIN2010-17550, and by the Spanish Ministry of Economy and Competitiveness within the Juan de la Cierva program.

Jorge Gracia is a postdoctoral researcher in the Artificial Intelligence Department at the Universidad Politécnica de Madrid, Spain. His research interests include ontology matching, semantic disambiguation, semantic measures, and multilingualism in the Semantic Web. Gracia has a PhD in computer science from the University of Zaragoza. Contact him at jgracia@fi.upm.es.

Eduardo Mena is an associate professor at the University of Zaragoza, Spain, where he leads the Distributed Information Systems research group. His research interests include interoperable, heterogeneous, and distributed information systems, the Semantic Web, and mobile computing. Mena has a PhD degree in computer science from the University of Zaragoza. Contact him at emena@unizar.es.