Computing the semantic distance between terms: An Ontology-based approach

Alicia Martinez[1], Fernando Pech[1], Noe Castro[1], Dante Mujica[1], Hugo Estrada[2], and Ilse Caspeta[1]
[1]Computer Science Department, CENIDET, Cuernavaca, Morelos, Mexico
{amartinez, fpech, ncastro, dantemv, ilselanda11c}@cenidet.edu.mx
[2]Research Area, INFOTEC, Mexico City, Mexico
hugo.estrada@infotec.com.mx

Abstract—*The semantic measure determines how they relate two terms or concepts. The challenge of calculating the similarity between terms has become a research area important and has many application in several fields such as artificial intelligence. The development of efficient measures for the computation of semantic similarity is fundamental for computational semantics. Semantic distance is a measure that identifies the strength of relationship between two concepts in an ontology.*

This paper presents the development of novel method (called NaoBig) that expresses the semantic distance between concepts of a knowledge base based on ontologies through a numerical factor. The semantic distance between concepts is shown graphically by a directed graph. Also, BigData RDF is used as search engines and indexing triplets.

Keywords: Semantic distance, semantic similarity measure, path based measure.

1. Introduction

The aim of the Semantic Web is to help automate tasks that require a level of conceptual understanding of the objects involved, and enabling software programs to automatically find and combine information and resources in consistent ways. The core of these new technologies are ontologies [10], which are key to represent formal knowledge so that it can be understood, used and shared between distributed application components.

Ontology is a description (formal knowledge) of concepts and their relationships. However, the information represented in ontology is not always reliable, because there may be two concepts in the same ontology, which are taxonomically distant. For example, given two concepts “heart”and “blood” where “heart” is a subclass of “cardiovascular system” and “blood” is subclasses of “body fluids”. Both concepts have no direct relationship within ontology. However, a person might consider that relationship concepts “heart” and “blood” is strong and should have a direct relationship within the ontology under the assumption that the heart pumps blood. To solve this problem, we propose to visualize the ontology to measure the semantic distance between concepts that the user wants to know.

The semantic measures are explored in various fields of research and has various direct and relevant applications such as natural language processing (disambiguation of words [14], synonym detection [13], automatic spelling error detection and correction [3]), knowledge management (thesauri generation [6], information extraction [2], semantic annotation [20], biomedical domain [19], ontologies [7], learning [18], etc.), information retrieval, etc.

The purpose of this paper is to present the development of a novel method (called NaoBig) that expresses the semantic distance between concepts of a knowledge base, which is based on ontologies through a numerical factor. The semantic distance between concepts is shown graphically by a directed graph. Also, BigData RDF is used as search engines and indexing triplets.

The rest of the paper is structured as follows: Section 2 presents related works.. Section 3 describes the method proposed for obtaining the Semantic distance, while Section 4 presents the results of the evaluation conducted during the case studies. Finally Section 5 concludes and briefly discusses future work.

2. Related work

2.1. Semantic measures

The three major semantic measures considered in the literature are: semantic similarity, semantic relation and semantic distance [5]. The semantic similarity is defined taking into account the lexical relations of synonymy (e.g., <car> and <automobile>) and hiperonimia, this measure evaluates the similarity between two concepts of a major subset of semantic links (e.g., is-a and part-of). The semantic relation indicates how distant semantically are two concepts in a network or taxonomy, by using all relations between them (e.g., hyponym, antonyms, meronymy or any functional relation including is-made-of, is-an-attribute-of). The semantic distance is a measure that identifies the strength of the relation between two concepts or terms. If the measure of semantic distance is less, there will be more semantic relationship between the two terms. The similarity measure can be classified by the ontological structure and the content of the information as follows:

- *Path length based measure*. It is based on the distance of the route that separates the concepts or terms. The quantification of similarity is based on the ontology or taxonomic structure [15], [11], [12].
- *Depth relative measure*. It is based on the shortest path approach, considering the depth of the edges of the two concepts in the general structure of the ontology [21], [8].
- *Information content based measure*. It uses both the path length and the depth to determine the similarity between concepts [16].
- *Hybrid measure*. It combines the knowledge derived from several sources of information (such as the path length, local density and some other approaches).
- *Feature based Measure*. It exploits the properties of the ontology to obtain the similarity values and is based on the assumption that each concept is described by a set of words that indicate their properties or characteristics.

In the context of these semantic measures some important contributions have been done.

In [15], Rada proposed an intuitive way to calculate the semantic similarity (also known as taxonomic or attributional measures, it states that the ontologies can be seen as direct graphs in which the concepts are interrelated among them. To calculate the similarity between two nodes/concepts is necessary to count the number of edges in the shortest path between two nodes. This means that the semantic distance of two concepts are correlated with the length of the shortest path.

Given a path $(c_1, c_2) = l_1, ...l_k$ as a set of links that connect the concepts c_1 and c_2 in a taxonomy, and cosidering all possible paths from c_1 to c_2, the semantic distance could be expressed by:

$$dist_{rad}(c_1, c_2) = len(c_1, c_2)$$

where len is the length of the shortest path between c_1 y c_2 with respect to the number of edges. However, this measure is based on the assumption that each edge carries the same amount of information, which does not apply in most ontologies [16].

In [12], Hirst and St-Onge defined the similarity as a distance between the path of two concepts, expressed as follows:

$$sim_{HS}(c_1, c_2) = C - \ path\ length - \ k \text{ x } d$$

where d is the number of changes of direction in the path, C and k are constant parameters (the authors use C=8, K=1); if there is no path, $sim_{HS}(c_1, c_2)$ is zero and concepts are not related. Hirst and St-Onge considered the following address path: up (as hiperonimia and meronymy), down (as hyponymy and holonym) and horizontal (as antonymy).

In [8], Ge and Qiu defined the mapping of the similarity of terms based on semantic distance considering the hierarchical relations, relations of semantic distance between terms and degree of measurement mapping between terms through semantic similarity. The algorithm takes two concepts as input and calculates the similarity in four stages: assigning weights between relations, generation of routes or paths between nodes, semantic distance calculation and calculation of semantic similarity. So, given two concepts c_1 and c_2 the expression to compute the weights is next.

$$w[sub(c_1, c_2)] = 1 + \frac{1}{k^{depth}(c_2)}$$

where $depth(c)$ represents the depth of the concept c regard to the concept of the root of node C in the ontology, k is a predefined factor greater than 1 that indicates the rate values of the hierarchy of the ontology.

Wu and Palmer in [21] proposed a strategy to measure the semantic representation of verbs and analyzes the impact on the problems of lexical selection in automatic translations. Since the concepts c_1 and c_2 the similarity measure is calculated with the following expression:

$$Sim_{WP}(c_1, c_2) = \frac{2H}{N_1 + N_2 + 2H}$$

where N_1 and N_2 are the number of "is-a" links from c_1 and c_2, respectively to the Lowest Common Subsumer (LCS) c, and H to the number of "is-a" links from c to the root of the taxonomy.

In [11], Hao et al. used the semantic distance between two concepts by calculating the length of the shortest path c_1 and c_2, as well as the depth under LCS in the tree of lexical hierarchy based on Wordnet to represent different points and calculate the semantic similarity of terms. They propose the following equation to calculate the similarity between two terms:

$$Sim(c_1, c_2) = \left(1 - \frac{\mid path(c_1, c_2) \mid}{\mid path(c_1, c_2) \mid + Depth(LCS(c_1, c_2)) + \beta}\right) \times \left(\frac{Depth(LCS(c_1, c_2))}{\mid path(c_1, c_2) \mid + Depth(LCS(c_1, c_2))/2 + \alpha}\right)$$

where α and β are smoothing factors. When $Depth(LCS(c_1, c_2)) = 0$, both terms have attributes less common and their similarity is 0.

3. NaoBig: Semantic Distance among terms in an Ontology

The semantic distance is a measure that identifies the strength of the relationship between two concepts or terms. You may disagree with the structure of ontology indicating that two terms that are far in the ontology should have a direct relationship. In Figure 1a an ontology that has the words "Aspirin" and "Child" is displayed and can be seen that there is no relationship between the words; however a user can say that there is a close relationship because a “Child” can take an “Aspirin”.

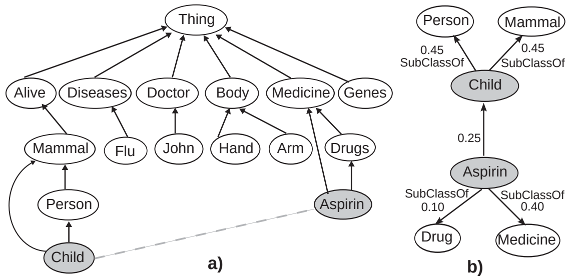
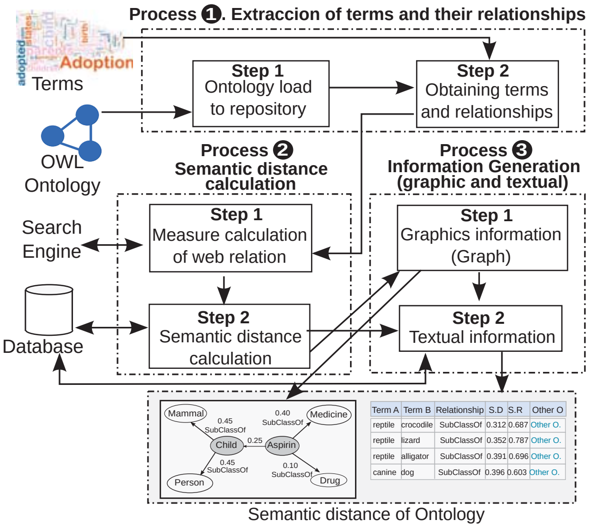


Fig. 1: Partial view of the terms or concepts “child” and “Aspirin”a) in an ontology, b) in an ontology shown by NaoBig.

In this paper a methodology called NaoBig is proposed for the visualization and navigation of a graph showing the semantic distance between concepts from ontology terms entered by the user. Presenting the information in this way, would allow a better decision making. For indexing ontologies and consulting information into NaoBig a BigData RDF API was implemented. Figure 1b shows a partial view of the concepts “Child” and “Aspirin”generated by our method NaoBig, where it can be seen the semantic distance existing between among the concepts.

Figure 2 shows the NaoBig methodology architecture which is made up of three processes: 1) terms and relationship extraction, 2) calculation of semantic distance and 3) generation of graphical and textual information.



Term A	Term B	Relationship	S.D	S.R	Other O
reptile	crocodile	SubClassOf	0.312	0.687	Other O.
reptile	lizard	SubClassOf	0.352	0.787	Other O.
reptile	alligator	SubClassOf	0.391	0.696	Other O.
canine	dog	SubClassOf	0.396	0.603	Other O.

Fig. 2: NaoBig architecture.

3.1. Terms extraction and relationships

This process allows the extraction of all data that can be obtained from an ontology in order to show to the end user those terms related to his query. The process is carried out with two inputs, the terms (to be used in knowledge representation) and an ontology (in OWL). Pairs of terms are extracted from the ontology (they can be superclasses, subclasses, instances or term properties) and type of the relations. The process 1 is made up of two stages:

1. *Loading the ontology in the repository*. OWL Ontology is loaded and saved in a repository in order to access the loaded information trough BigData RDF Database, because of it contains the information that will be used for displaying the semantic distance of terms.
2. *Getting terms and relations*. This stage consists in the retrieving data from the repository containing the ontology; superclasses, subclasses, instances, relations and / or properties that have the terms that the user enters the system are extracted. It can be seen in Figure 1 that the term “child” is related to the terms “person” and “mammal” (both as superclasses). When the user enters any term queries are executed for obtaining the following terms:
   - Retrieving terms of lower level (instances or subclasses).
     $SELECT\ ?z\ ?y\ WHERE\ \{\ ?z\ ?y\ <\ ''+term+''\ >\}$
     where *term* is entered by the user. The query returns all those terms $?z$ and the kind of relationship $?y$ that exist with *term*.
   - High level term extraction if the term is an instance
     $SELECT\ ?y\ ?z\ WHERE\ \{<''+term+''>\ ?y\ ?z\ \}$
     term may be an instance in the ontology. The query returns terms that can be classes or instances (?z) and the type of relationship (?y) that exists with term.
   - Superclass terms extraction
     $SELECT\ ?z\ WHERE\ \{<\ ''+term+''\ >\ rdfs:subClassOf\ ?z\ \}$

   The results of previous queries are saved in an array that will be used in the process 2.

3.2. Calculating the semantic distance

In this stage a numerical value indicating the degree of relationship (semantic distance) among the terms according to the paternship by frequency is calculated. The pairs of terms are extracted in the process 1. This process is necessary in various APIs that provide data to calculate the semantic distance such as Google Custom Search[1] and Watson[2].

For this process it is necessary a connection to our Database “NaoBig” and Google Custom Search API to

[1] https://developers.google.com/custom-search/
[2] https://developer.ibm.com/watson/

obtain data that help us to calculate the semantic distance. The inputs of this process are the terms and relationships extracted from Process 1 using values obtained with Google Custom Search and Watson. The outputs are the pairs of terms and the semantic distance between terms and their relationships. This process is made up of two stages:

1. *Calculation of Web relationship measure*. The Web relationship among terms is calculated using the Garcia and Mena's formula [9]. The two-term Web relation (from the process 1) is calculated with the measure and frequency of both terms using the formula of the normalized distance of Google $NGD(x, y)$, also known as Normalized Web Distance $NWD(x, y)$ defined in [5] and which is shown above:

   $$NGD(x,y) = \frac{max\{log\ f(x), log\ f(y)\} - log\ f(x,y)}{log\ M - max\{log\ f(x),\ log\ f(y)\}}$$

   where M is the total number of pages indexed by google or the number of ontologies and semantic Web documents retrieved by watson [17], $f(x)$ and $f(y)$ are the frequencies of terms x and y obtained with the Google Custom Search API and Watson, respectively. Note that Google Custom Search only allows 100 queries per day for free. All frequencies of the query terms in both servers are stored in our database NaoBig so that when you want to know the frequency of a term, it is first checked in our DB; if not found, Google Custom Search and Watson are used.

   NGD has a range of values from 0 to ∞, however Gracia and Mena [9] make use of an improvement to this formula to obtain a range from 0 to 1. To obtain the Web relationship next formula is applied:

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

   Ontological context (*OC(t)*) is also taking into account, which is a set of ontological terms extracted from the repository containing the ontology, in order to disambiguate the query terms:
   OC(t) is defined as the minimum set of ontological terms located on an ontology.

   - If *t* is a class then *OC(t)* is the set of direct hyperonyms and is obtained with the following query:
     $SELECT\ ?z\ WHERE\{<\ "+term+"\ >$ $rdfs: subClassOf\ ?z\}$
   - If *t* is a instance then *OC(t)* is the class to which it belongs and it is returned with the query:
     $SELECT\ ?z\ WHERE\{<\ "+term+"\ >$ $rdf: type\ ?z\}$
   - If *t* is a property then *OC(t)* is the set of classes of its domain and is obtained with the query:
     $SELECT\ ?z\ WHERE\{<\ "+term+"\ >$ $rdfs: domain\ ?z\}$

   To calculate the Web relationship of the ontological context the following formula is used:
   $relWebOC(x,y)\ =\ e^{-2NWD(OC(x),OC(y))}$
   where *OC(x)* and *OC(y)* are the ontological context of the term1 and term2, respectively.
   The results obtained from *relWeb(x,y)* and *relWebOC(OC(x), OC(y))* are used for calculating the semantic distance.
2. *Calculating the semantic distance*. In this stage the Semantic distance is calculated using the Web relationship calculated in the previous stage. To achieve this, it is necessary to first calculate the semantic relationship between $Term1(x)$ and $Term2(y)$ by applying a weighting w_0 and w_1 to *relWeb(x,y)* and to *relWebOC(OC(x), OC(y))* of the obtained values in Google and Watson.
   To calculate the Semantic distance with Google the formula is:
   $RSGoogle(x,y)\ =\ w_0\ *\ relWeb(x,y)\ +\ w_1\ *\ relWebOC(OC(x),\ OC(y))$
   Semantic distance in Watson is calculated with:
   $RSWatson(x,y)\ =\ w_0\ *\ relWeb(x,y)\ +\ w_1\ *\ relWebOC(OC(x),\ OC(y))$

   where w_0 and w_1[3] are weighting values that must be higher than 0 and the sum must be equal to 1. After obtaining the values of the semantic relationship using Google and Watson, a combination of these two results is performed to obtain the semantic relationship of *x* and *y*, which a weighting to each of the results is again applied, as shown below:

   $$RelSem(x,y)\ = \\ wt_0 * RSgoogle(x,y)\ +\ wt_1 * RSwatson(x,y)$$

   Where wt_0 y wt_1[4] are weighting values which must be higher than 0 y and the sum of the values must be equal to 1.

   According to [9], "the semantic distance is the inverse of the semantic relationship. The two terms more related semantically are the closest to each other". Being 1 the largest value in the semantic relationship and 0 the closest value in the semantic distance, so:
   If *Relsem(x,y)=1* then *DistSem(x,y)=0*
   If *Relsem(x,y)=0* then *DistSem(x,y)=1*

   Therefore, the formula considered for obtaining the semantic distance is as follows:

   $$DistSem(x,y) = 1\ -\ RelSem(x,y)$$

   where x and y are the pair of terms obtained from the Process 1. The result of this step is the calculation

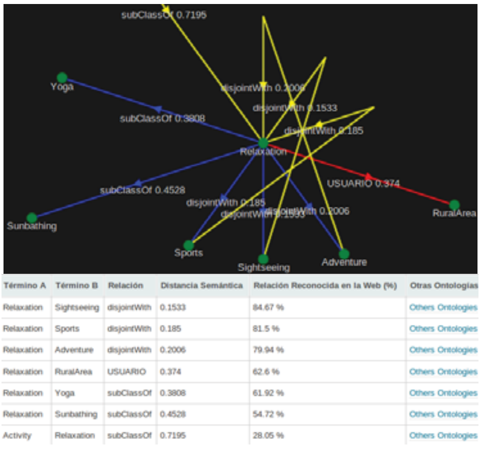
[3] The value of w_0 y w_1 applied in this work is 0. 5

[4] The values used in this work are: to wt_0 = 0,7 and wt_1 = 0,3

of the distance semantics. This numerical value is the input to the next process that is described below.

3.3. Process Information generation

NaoBig interface for graphical and textual display (see Figure 3) of semantic distance was developed in the third process of our methodology proposed. The JavaScript InfoVis Toolkit[5] was used because provides tools for creating Interactive Data Visualizations for the Web.



Término A	Término B	Relación	Distancia Semántica	Relación Reconocida en la Web (%)	Otras Ontologías
Relaxation	Sightseeing	disjointWith	0.1533	84.67 %	Others Ontologies
Relaxation	Sports	disjointWith	0.185	81.5 %	Others Ontologies
Relaxation	Adventure	disjointWith	0.2006	79.94 %	Others Ontologies
Relaxation	RuralArea	USUARIO	0.374	62.6 %	Others Ontologies
Relaxation	Yoga	subClassOf	0.3808	61.92 %	Others Ontologies
Relaxation	Sunbathing	subClassOf	0.4528	54.72 %	Others Ontologies
Activity	Relaxation	subClassOf	0.7195	28.05 %	Others Ontologies

Fig. 3: Textual and graphical representation of NaoBig.

This process requires the JSON data exchange file (it generated in the process 1) and textual information generated in the process 2. This process consists of two steps:

1. *Graphics information (Graph)*. The JSON data exchange file is generated to graphically display the user's query. It receives as input pairs of terms, relationships and semantics distance calculated in the process 2. The output is both graphic and textual representation of the semantic distance. It also generates a data exchange file that contains the nodes and arrows that compose the graphical representation.
   The data exchange file is composed of three objects, these are: main nodes (these are generated from the terms obtained from the process 2), auxiliary nodes (these are generated from the semantic distance -See Table 1) and relations (these are generated for joining the main nodes and the auxiliary nodes). Infovis JavaScript Toolkit displays objects as Figure 4a.
   For example, if the user creates the relationship between the terms: "Reptile" and "Dog" with a semantic distance of 0.45. This can be seen as in Figure 4b. Thus, 21 auxiliary nodes will be created, as shown in Table 1. Figure 4c shows the terms "Reptile" and "Lizard", with the "SubClassOf" relationship, with a semantic distance of 0.35. therefore auxiliary nodes 17 are created, as shown in Table 1.

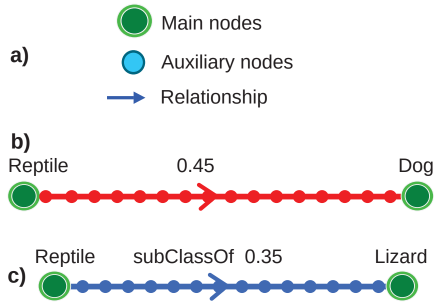


Fig. 4: Representation of the terms on JavaScript Infovis Toolkit.

   The output of this step is a graph, which displays the terms related to "reptile" and "dog" and its semantic distance.

Table 1: Number of auxiliary nodes by semantic distance.

S.D.	#nodes	S.D.	#nodes
0.01	3	0.325	16
0.025	4	0.35	17
0.05	5	0.375	18
0.075	6	0.4	19
0.1	7	0.425	20
0.125	8	0.45	21
0.15	9	0.475	22
0.175	10	0.5	23
0.2	11	0.525	24
0.225	12	0.55	25
0.25	13	0.575	26
0.275	14	...	...
0.3	15	1	43

2. *Textual information*. This step generates the textual representation of graphic content in order to display a table with the information contained in the graph. The input of this step is: the pairs of terms, relationships and semantics distance calculated in the process 2.

   The generation of graphic representation required to store the values retrieved in the process 2 in a table (pairs of terms, the relationship and semantic distance) in order to have all the information generated from the user's query. The textual information generated is extracted from the local database.

4. Tests and results

The semantic distance obtained with our methodology was evaluated with two different approaches: correlation and efficiency.

4.1. Correlation approach

The results of the semantic distance obtained with NaoBig are compared with a gold standar WordSim353 [4], [1]. The correlation between the values NaoBig and gold standard

[5] http://thejit.org

WordSim353 is measured with Spearman correlation (ρ) [1], the following formula is used:

$$\rho = 1 - \frac{6\sum D^2}{N(N^2-1)}$$

The interpretation scale of the correlation coefficient is between -1 and 1, the value 0 indicates no linear association between the two variables of study.

4.2. Efficiency approach

It is for the evaluation of the performance of NaoBig. Pairs of words in different ontologies were taken, a range of values was determined to determining whether two words were closely related, and another range for words with little relationship, the ranges are:

- If the semantic distance >0 and <= 0.5, then, are closely related words (high relationship).
- If the semantic distance >a 0.5 y <1, then, have little related words (low relationship).

The criteria for these measures is based on a heuristic. Since the measure of semantic distance is subjective, we chose to have two ranges; a pair words with high relationship and a pair of words with low relationship, both in the same size range, in this case 0.5. To measure the efficiency of NaoBig, precision metrics, recall and F-measure (efficiency) were used [4].

- To measure the efficiency of NaoBig regarding words with high relationship (*Frel*), the following expression was used:

  $$Frel = \frac{2 \times Prel \times Rrel}{Prel\ +\ Rrel}$$

  Where (*Prel*) is the precision of words with high relationship, *Rrel* is coverage of words with high relationship.
- To measure the efficiency of NaoBig regarding words with low relationship (*FrelB*), the following expression was used:

  $$FrelB = \frac{2 \times PrelB \times RrelB}{PrelB\ +\ RrelB}$$

  where (*PrelB*) is the precision of words with low relationship, *RrelB* is coverage of words with low relationship.

The tests were conducted with three ontologies, OntoSem[6] and OpenCyc[7].

For both approaches, 3 distinct formulas were applied to calculate the semantic distance: 1) using the google search engine, 2) using Watson y 3) using NaoBig (combination of both search engines, giving a weight of 0.7 to google and 0.3 to Watson). For reasons of space only two case studies are described:

1. Using the pair of terms "computer" and "software" with OntoSem. This ontology contains 60 pairs words of WordSim353 gold standard used for correlation and efficiency. Table 2 shows that there is a better correlation using Google search engine formula (close to 1). Table 3 shows results in the words with high relationship, and Table 4 shows results in the words with low relationship; in both tables shows that the best results in accuracy, coverage and efficiency (F-measure) are obtained by google.

Table 2: Result of Spearman correlation with OntoSem and OpenCyc.

	Ontosem			OpenCyc		
	Google	Watson	NaoBig	Google	Watson	NaoBig
Spearman	**0.4371**	-0.0427	0.3700	-0.0473	0	**0.13054**

Table 3: Result of precision, coverage, and efficiency in terms (high relationship) with OntoSem.

	Google	Watson	NaoBig
P.T with S.D. <0.5 classifieds manually	39	39	39
Correct number of P.T with S.D.<0.5	26	13	19
P.T <0.5 returned	33	20	37
Precision (high relationship)	**0.7878**	0.65	0.7030
Coverage (high relationship)	**0.7222**	0.4406	0.5757
F-measure (high relationship)	**0.7748**	0.51948	0.5952

P.T. = Pair of terms S.D.= Semantic distance

Table 4: Result of precision, coverage, and efficiency in terms (low relationship) with OntoSem.

	Google	Watson	NaoBig
P.T with S.D. >0.5 classifieds manually	21	21	21
Correct number of P.T with S.D. >0.5	14	14	13
P.T >0.5 returned	27	40	33
Precision (low relationship)	**0.5185**	0.35	0.3939
Coverage (low relationship)	**0.6666**	6666	0.6190
F-measure (low relationship)	**0.5833**	0.4590	0.4814

P.T. = Pair of terms S.D.= Semantic distance

2. Using the terms pair "planet" and "astronomer" with OpenCyc. This ontology contains 29 pairs of words in the gold standard WordSim353. Table 2 shows that the correlation with NaoBig is the nearest to 1. Table 5 shows results with respect to the set of pairs words with high relationship; therefore, the better accuracy and coverage are of Watson, and F-measure by NaoBig. Table 6 displays the results in the words with low relationship: coverage by Watson, precision and F-measure by NaoBig.

Of the various tests with gold standard, the better correlation was with NaoBig. With an efficiency of almost 60 % regard to the semantic distance, precision of up to 80 % in the pairs of terms with high relationship, and the low relationship terms was obtained less than 70 %.

[6] http://morpheus.cs.umbc.edu/aks1/ontosem.owl

[7] http://www.OpenCyc.com/platform/openOpenCyc/downloads

Table 5: Result of precision, coverage, and efficiency in terms (high relationship) with OpenCyc.

	Google	Watson	NaoBig
P.T with S.D. <0.5 classifieds manually	20	20	20
Correct number of P.T with S.D. <0.5	3	1	8
P.T <0.5 returned	5	1	9
Precision (high relationship)	0.6	**1**	0.8889
Coverage (high relationship)	0.15	0.05	**0.4**
F-measure (high relationship)	0.2400	0.09524	**0.5517**

P.T. = Pair of terms S.D.= Semantic distance

Table 6: Result of precision, coverage, and efficiency in terms (low relationship) with OpenCyc.

	Google	Watson	NaoBig
P.T with S.D. >0.5 classifieds manually	9	9	9
Correct number of P.T with S.D.>0.5	7	9	8
P.T >0.5 returned	24	28	20
Precision (low relationship)	0.29167	0.32143	**0.4**
Coverage (low relationship)	0.7778	**1**	0.88889
F-measure (low relationship)	0.4242	0.48649	**0.5517**

P.T. = Pair of terms S.D.= Semantic distance

5. Discussion and future work

The semantic distance calculation depends on the information that is retrieved by the search engines (google and Watson). This is because the process of semantic distance calculation is based on the relationship of association frequency of terms in the corpus. NaoBig, a methodology based on the combination of values obtained with the search motors Normal Web and Semantic Web was developed in this paper. As we have experimentally demonstrated, the results obtained with our approach were better than those obtained with each search motor evaluated in separately way. Quantitatively, the proposed scheme obtained an efficiency close to 60 % with respect to semantic distance, an accuracy of 80 % to recognize terms with high relation, a coverage of 76 % to recognize terms with low relation. Also, it obtained the best correlation coefficient in comparision with Google and Watson. To extend the tool to future,be advisable to perform different tests with a larger set of pairs of terms from different ontologies. Furthermore, it would be interesting to add another numerical factor to calculate the semantic distance, as the frequency of terms from a corpus.

References

[1] Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 19–27, Stroudsburg, PA, USA, 2009.

[2] John Atkinson, Anita Ferreira, and Elvis Aravena. Discovering implicit intention-level knowledge from natural-language texts. *Knowledge-Based Systems*, 22(7):502 – 508, 2009.

[3] Alexander Budanitsky and Graeme Hirst. Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In *Proceedings of the Workshop on WordNet and Other Lexical Resources, Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2001)*, pages 29–24, Pittsburgh, PA, 2001.

[4] Alexander Budanitsky and Graeme Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Comput. Linguist.*, 32(1):13–47, March 2006.

[5] Rudi L. Cilibrasi and Paul M. B. Vitanyi. The google similarity distance. *IEEE Trans. on Knowl. and Data Eng.*, 19(3):370–383, March 2007.

[6] James R. Curran. Ensemble methods for automatic thesaurus extraction. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, EMNLP '02, pages 222–229, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[7] Anna Formica. Concept similarity in formal concept analysis: An information content approach. *Knowledge-Based Systems*, 21(1):80 – 87, 2008.

[8] Jike Ge and Yuhui Qiu. Concept similarity matching based on semantic distance. In *Proceedings of the 2008 Fourth International Conference on Semantics, Knowledge and Grid*, SKG '08, pages 380–383, Washington, DC, USA, 2008. IEEE Computer Society.

[9] Jorge Gracia and Eduardo Mena. Web-based measure of semantic relatedness. In *In Proc. of 9th International Conference on Web Information Systems Engineering (WISE 2008), Auckland (New Zealand*, pages 136–150. Springer, 2008.

[10] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220, June 1993.

[11] Dou Hao, Wanli Zuo, Tao Peng, and Fengling He. An approach for calculating semantic similarity between words using wordnet. In *ICDMA*, pages 177–180. IEEE, 2011.

[12] Graeme Hirst and David St-Onge. Lexical chains as representations of context for the detection and correction of malapropisms, 1997.

[13] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 296–304, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[14] Siddharth Patwardhan, Satanjeev Banerjee, and Ted Pedersen. Using measures of semantic relatedness for word sense disambiguation. In *Proceedings of the 4th International Conference on Computational Linguistics and Intelligent Text Processing*, pages 241–257, Berlin, Heidelberg, 2003. Springer-Verlag.

[15] Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(1):17–30, 1989.

[16] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'95, pages 448–453, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[17] Marta Sabou, Miriam Fernández, and Enrico Motta. Evaluating semantic relations by exploring ontologies on the semantic web. In *Natural Language Processing and Information Systems, 14th International Conference on Applications of Natural Language to Information Systems, NLDB 2009, Saarbrücken, Germany, June 24-26, 2009. Revised Papers*, pages 269–280, 2009.

[18] David Sánchez. A methodology to learn ontological attributes from the web. *Data & Knowledge Engineering*, 69(6):573 – 597, 2010.

[19] David Sánchez and Montserrat Batet. Semantic similarity estimation in the biomedical domain: An ontology-based information-theoretic perspective. *Journal of Biomedical Informatics*, 44(5):749 – 759, 2011.

[20] David Sánchez, Montserrat Batet, and David Isern. Ontology-based information content computation. *Knowledge-Based Systems*, 24(2):297 – 303, 2011.

[21] Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, Stroudsburg, PA, USA, 1994. Association for Computational Linguistics.