KNOWLEDGE MEDIA

KMi

INSTITUTE

An Ontology-Driven Similarity Algorithm

Tech Report kmi-04-16

Maria Vargas-Vera and Enrico Motta

The Open University

An Ontology-Driven Similarity Algorithm

Maria Vargas-Vera and Enrico Motta
Knowledge Media Institute (KMi),
The Open University, Walton Hall,
Milton Keynes, MK7 6AA,
United Kingdom
{m.vargas-vera@open.ac.uk}

Abstract. This paper presents our similarity algorithm between relations in a user query written in FOL (first order logic) and ontological relations. Our similarity algorithm takes two graphs and produces a mapping between elements of the two graphs (i.e. graphs associated to the query, a subsection of ontology relevant to the query). The algorithm assesses structural similarity and concept similarity. An evaluation of our algorithm using the KMi Planet ontology[1] is presented. We also carried out an experiment to test the human judgment about similarity using context and without context. Our similarity algorithm has been manly used in AQUA, our question answering, in the query reformulation process.

1 Introduction

Similarity ha been an important research topic in several fields such linguistic, Artificial intelligence (in particular in the field of Natural Language Processing and Fuzzy Logic). The range of application of measure of similarity ranges from word sense disambiguation, text summarization, information extraction and retrieval, question answering, automatic indexing and automatic correction of codes.
There are two types of similarity: syntactically and semantic similarity. Syntactic similarity can be defined as functions over terms. For instance the hamming distance (used in Information Theory). This similarity is defined as the number of positions with different characters in two terms with the same length. Whilst semantic similarity can be defined as Miller and Charles (Miller et al 1991) as a continuous variable that describes the degree of synonymy between two words.

When evaluating similarity in a taxonomy the most natural way to access similarity is to evaluate the distances between the two concepts being compared. Therefore the shorter is the path from one to another means that they are more similar. This approach has been used as measure of similarity. However, one of the main drawbacks is that it relies on the notion that links in a taxonomy represent uniform distances (Resnik 1995; 1998). Our own view is that similar entities are assumed to have common features[2]. For instance (university, research_institute) but it is also the case that dissimilar entities may also be semantically related by the relation meronym or holonym such as (student –person; bicycle-wheel) .

Our similarity algorithm assess concept similarity and relation similarity. It compares extended graph obtained from the user query (using entities in the query itself plus informative classes from ontology) and a graph which represents a subset of the ontology (relevant to the query). As an inter-media stage it creates the intersection upon nodes (described in section 3) between the two graphs (the **graph of the query** and the **graph obtained from the ontology)**. In short, our similarity algorithm assess concept similarity and relation similarity using the Dice Coefficient using the most informative classes from the ontology.

The main application of our similarity algorithm is to be part of AQUA a question answering system (Vargas-Vera et al 2003a; 2003b, 2004). The goal of AQUA is to answer a question using several

[1] The KMi Planet ontology describes academic life at KMi (KMi is a short name for Knowledge Media Institute).
[2] The common features in two entities might not be so discriminative as the features which are different in them.

resources like databases, ontologies and knowledge bases. However, each of these resources use their own vocabulary (shows a viewpoint of its creator). Therefore, it could be a mismatch between name of relations in the user query and the name of relations in resources. Then query needs to be reformulated before it can be executed.

The paper is organized as follows: Section 2 describes the AQUA our logic-based question answering system. Section 3 describes the similarity algorithm embedded in AQUA. Section 4 describes an evaluation of the SimilarityBase algorithm and Similarity procedure. Section 5 gives an experiment carried out to assess human judgment about similarity between two terms (verbs). Section 6 describes a section of related work. Finally, section 7 gives conclusions and directions for future work.

2 AQUA a Question Answering System

AQUA, is a question answering system developed at the Open University in England UK (Vargas-Vera et al 2003a; 2003b, 2004). AQUA combines Natural Language processing (NLP), Ontologies, Logic, and Information Retrieval technologies in a uniform framework. AQUA makes intensive use of an ontology in several parts of the question answering system. The ontology is used in the refinement of the initial query, the reasoning process (a generalization/specialization process using classes and subclasses from the ontology), and in the similarity algorithm.

Queries are formulated in plain English are translated by AQUA into logic formulae using the grammatical components obtained by our parser. AQUA then looks for an answer in different resources such as databases, a populated ontology (or knowledge base) and the Web. Currently, AQUA makes use of an inference engine which is based on the Resolution algorithm. AQUA uses also our similarity algorithm which is based on both the ontological structures and instances of the ontology, a WordNet thesaurus and the Dice coefficient. The similarity algorithm, is a key feature of AQUA. It is used to find similarities between relations/concepts in the translated query and relations/concepts in the ontological structures. The similarities detected then allow the interchange of concepts or relations in a logic formula corresponding to the user query. In this way, we make the mapping between user 's queries and ontological spaces.

AQUA also has facilities for analyzing and explaining proofs. The explanation is provided both in pseudo-natural language and as visualization. AQUA is coupled with the KMi ontology in a first instance. Therefore, questions about academic life in our institute can be answered when AQUA works as closed-domain question answering system. When AQUA works as open-domain question answering system uses the Web as a resource. While in the latest mode questions such as **who killed John Kennedy?** are answered by AQUA using information retrieval techniques.

3 Similarity Algorithm

The success of the attempt to satisfy a query depends on the existence of a good mapping between the names of relations used in the query and names of relations used in the knowledge base. Therefore, we have embedded in AQUA a similarity algorithm. Our similarity algorithm uses both ontological structures and instances in the selected ontology, the Dice coefficient[3] (Frakes et al. 1992, Manning et al. 1999) and the WordNet thesaurus.
Our similarity algorithm differs from other similarity algorithms in that it uses ontological structures and also instances. Instances provide evidential information about the relations being analyzed. This is an important distinction between the kind of similarity which can be achieved using only WordNet or the similarity which can be achieved using distances to super-classes followed by Wu et al. (Wu et al 1994). In the first one WordNet brings all the synsets found even the ones which are not applicable to the problem

---

[3] As a reminder, the Dice coefficient measuring the similarity of two vectors X, and Y is defined by

$$\frac{2\,|X \cap Y|}{|X| + |Y|}$$

It is normalized for length by dividing by the total number of non-zero entries, and doubled so that we get a measure that ranges from 0.0 to 1.0. A coefficient of 1.0 indicates identical vectors.

being solved. Whilst in the latest the idea of a common super-class between concepts is required. In our case is not a necessary condition.

We present an explanation of our algorithm when arguments in the user query are grounded (instantiated terms) and they match exactly (at the level of strings) with instances in the ontology. A detailed description of the algorithm and example can be found in (Vargas-Vera et al. 2003b).

The algorithm uses all grounded terms in the user query. It tries to find them as instances in the ontology. Once they are located a portion of the ontology (G_2) is examined including neighborhood classes. Then an intersection [4] (G_3) between augmented query (using knowledge from the ontology) G_1 and G_2 is performed to assess structural similarity. It could be the case that in the intersection G_3 several relations can include the grounded arguments. Then the similarity is computed for all relations (containing elements of the user query) using Dice Coefficient. Finally, the relation with the maximum Dice Coefficient value is selected as the most similar relation.

AQUA reformulates the query using the most similar relation and it then tries to prove the reformulated query. If no similarity is achieved using our similarity algorithm then AQUA offers to the user the synsets obtained from WordNet. From this offered set of sysnsets the user selects the suitable one. In this paper we show the case when both arguments, on the user query, match exactly with terms in the KMi ontology. Note that the similarity between concepts is not presented in this paper.

The similarity algorithm for relations is defined as follows:

SimilarityBase algorithm:

Case 1: X_1 and X_2 are grounded arguments.

1. Translate the question to First Order Logic i. e. predicate_name(X_1, X_2)

2. $\exists$ relation connecting C_1 and $C_2 \wedge \exists C_1 \supset X_1 \wedge \exists C_2 \supset X_2$ such that

relation(C_1,C_2) where relation is an ontological relation between C_1 and C_2.

3. $\exists$ relation connecting C_1 and $C_2 \wedge \exists C_1 \supset X_1 \wedge \exists C_2 \supset X_2$ such that

relation(C_1, C_2) $\wedge \ \exists S_1 \supset (U_{11} \subset U_{12} \subset \ldots \subset U_{1n}) \supset C_1 \wedge \exists S_1 \wedge (U_{11} \subset U_{12} \subset \ldots \subset U_{1n}) \supset C_2$

4. $\exists$ relation connecting C_1 and $C_2 \wedge \exists C_1 \supset X_1 \wedge \exists C_2 \supset X_2 \wedge S_1 \supset X_1 \wedge S_2 \supset X_2 \wedge \exists (U_{11} \subset U_{12} \subset \ldots \subset U_{1n}) \subset C_1 \wedge \exists (U_{21} \subset \ldots \subset U_{2n}) \subset C_2$
where Uij is a subclass of U_{ij+1}

5. Find the intersection, G_3, of G_1 and G_2 based upon the node labels.

6. Let be A and B vectors containing the features used to compare similarity.

Compute:

Concept _similarity = 0 no common concepts

---

[4] Intersection means to find a sub-graph in G_2 which contains all concepts contained in graph G_1using subsumption relation.

Concept _similarity = 1 same set of concepts , otherwise

Concept_similarity = sim_dice(A,B) = $2*\sum_1^n aibi / \sum_1^n ai^2 + \sum_1^n bi^2$
vectors A and B are filled with the number of concept nodes of graph G_1 and G_3 respectively.

7. Compute Relation_ similarity = d_i = sim_dice(A,B) = $2*\sum_1^n aibi / \sum_1^n ai^2 + \sum_1^n bi^2$
vector A and B are filled with the number of arcs in the immediate neighborhood of the graph G_1 and G_2 respectively.

8. maximum(d_i) where i=1,n

The algorithm builds graphs from the user query G_1, obtain a fragment from the ontology containing the relevant nodes G_2 and build an intersection (G_3) between G_1 and G_2.
Step 2 describes the construction of the graph G_1 for the query. This is created using subject (X_1), relation, object (X_2) and the most representative classes for X_1 and X_2 respectively.
Step 3 describes how the algorithm finds sub-hierarchy containing grounded[5] arguments/concepts from the user question (i.e., the neighborhood containing the grounded arguments).
Step 6. the similarity between G_1 and G_3 using the Dice coefficient is computed. The vectors A and B are filled with the number of concept nodes of graphs G_1 and G_3 respectively then

Concept _similarity = 0 no common concepts

Concept _similarity = 1 same set of concepts , otherwise

Concept _similarity = $2*\sum_1^n aibi / \sum_1^n ai^2 + \sum_1^n bi^2$

Step 7. the similarity between G_1 and G_2 using the Dice coefficient is computed.

Relation_similarity = $d_i = 2*\sum_1^n aibi / \sum_1^n ai^2 + \sum_1^n bi^2$

vector A and B are filled with the number of arcs in the immediate neighborhood of the graph G_1 and G_2 respectively.

A procedure called SimilarityTop uses the SimilarityBase algorithm (defined above), WordNet synsets, and feedback from the user. The SimilarityTop procedure checks if there is similarity between the name of the relation/concept in the query and the relation/concept in the selected ontology. If there is no similarity then it offers all the senses which are found in the WordNet thesaurus to the user. It is SimilarityTop that AQUA uses, with the selected sense, to rewrite the logic formulae. The main steps are defined as follows:

SimilarityTop procedure:

- Call our **SimilarityBase** algorithm (defined above)

- If ontological_relation $\neq \Theta$ then

  evaluate_query(ontological_relation(β_1,β_2))

[5] Grounded argument means instantiated argument.

- Else
  Obtain synsets for relation_question using WordNet thesaurus
  Ask user to select sense from the ones that WordNet thesaurus provided
  Call evaluate_queryl(selected_sense(β_1, β_2))

To illustrate how our similarity algorithm works we present an example. Note that in more complex examples the graph G_2 could have several relations which could be assessed for similarity.

3.3.1 Working Example

In this section we illustrate how our similarity algorithm works by presenting a working example. Note that in more complex examples the graph G_2 could have several relations which could be assessed for similarity.

Let us imagine that someone asks the question: "Does Enrico Motta work on AKT?"
work(enrico-motta,akt).

By refining our query using the KMi ontology we obtain the following formula:

project(akt) & researcher(enrico-motta) & work(enrico-motta,akt).

If AQUA evaluates this query over the knowledge base, it is likely that the question will fail. The problem is that the name of the relations in the knowledge base and the names of relations in the question might be completely different. AQUA will ask the user if they want to use similarity. If the answer to this question is ``yes" then AQUA builds three graphs: the graph associated with the question (G_1), the graph using ontological structures (G_2) and the intersection graph of G_1 and G_2. These graphs are shown in Figure 2.

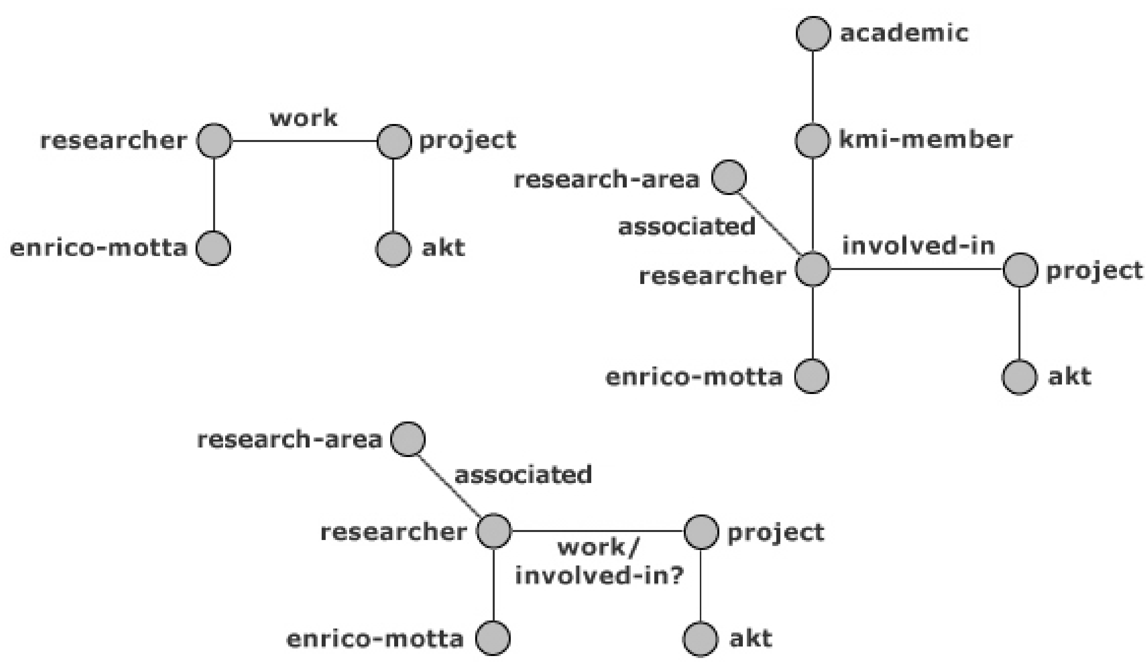

Figure 2. Graph G_1 and G_2 and intersection G_3

The relation similarity is computed as follows:

In the example shown in Figure 2 using G_1 and G_3.

Concept _similarity = 1

In order to compute Relation_similarity we use G_1 and G_2 graphs and vectors A=(2,2) and B=(3,2)

Then, Relation_similarity = $d_i = 2^* \sum aibi / \sum_1{}^2 ai^2 + \sum_1{}^2 bi^2$

$d_i = (A,B) = (2 *(6+4)) / (4+4) +(9+4) = (2*10)/21 = 20/21=0.9$

The outputs of the similarity algorithm is 0.9 and the name of the relation between concept *researcher* and concept project is "involved-in", Then the question is re-written as follows:

project(akt) & researcher(enrico-motta) & involved-in(enrico-motta,akt).

Once that a reformulation is proposed by AQUA or by user selecting the correct one. Then, the question is re-evaluated by the interpreter. By using the similarity algorithm AQUA tries to reduce to a minimum the possibility of failure because of mismatches between relation names. Since, there are several techniques for assessing similarity (Doan et al. 2002, Noy et al. 2000, Wu et al. 1994). a future implementation of AQUA will contain several algorithms for similarity.

4 Evaluation

We evaluated our similarity algorithm using queries to the KMi ontology (KMi is a short name for Knowledge Media Institute). Some of the typical queries used in the experiment are shown in Table I. However, a more comprehensive set of questions used in our experiment can be found at the AQUA home page: http://kmi.open.ac.uk/projects/akt/aqua/
The evaluation has been performed using the KMi ontology which describes academic life at Knowledge Media Institute (KMi).

AQUA as closed domain question answering system:

NLP query	Query expressed as logic predicate	Reformulated predicate using SimilarityBase algorithm	Concept Similarity value	Relation Similarit y value
1. Does Enrico Motta work in akt?	work(enrico-motta, akt) & project(akt)	involved (enrico-motta,akt) & project(akt)	1	0.9
2. Is David Celjuska associated to akt?	associated(david-celjuska,akt) & project(akt)	involved(david-celjuska,akt) & project(akt)	1	0.9

Table I. query and its reformulation using the **SimilarityBase** algorithm

Question 2 (Is David Celjuska associated to akt?) can be reformulated as suggested in Table I by using our **SimilarityBase algorithm**. However, if we use the suggested WordNet senses as first resource then the query can be reformulated as is shown Table II. The latest is not an optimal solution for AQUA since it will try to evaluate (as logic predicates) each of the reformulation. Therefore, our goal of having a question answering system in real time it might suffers of the problem of slow time response. The question 2 example (from Table I) illustrates that by using a domain specific ontology queries can have less alternative reformulations.

Query	Reformulations using WordNet senses
Is David Celjuska associated to akt?	linked(david-celjuska,akt) & project(akt) related(david-celjuska,akt) & project(akt) connected(david-celjuska,akt) & project(akt) affiliated(david-celjuska,akt) & project(akt) linked-up(david-celjuska,akt) & project(akt)

Table II. query and its reformulation using WordNet suggested senses

When AQUA fails to find a candidate for similarity using SimilarityBase (i.e. null value is returned). Then, it tries to find possible reformulations using the synsets of WordNet. Table III shows an examples of queries and possible reformulations offered to the user by AQUA.

AQUA as closed domain question answering system:

NLP query	Query expressed as logic predicate	Reformulated predicate using SimilarityBase algorithm	Reformulated predicate using SimilarityTop procedure
Who is D. Fensel ?	is(d-fensel,X)	Null	have-the-quality(d-fensel,X) be-identical(d-fensel,X) be-somewhere(d-fensel,X) cost-of-be(d-fensel, X) remain (d-fensel, X) live(d-fensel, X) exist(d-fensel, X) equal(d-fensel, X) constitute(d-fensel, X) embody(d-fensel, X) take(d-fensel, X) occupy(d-fensel, X) represent(d-fensel, X) comprise(d-fensel, X) make-up(d-fensel, X)

Table III. query and its reformulation using the **SimilarityTop** procedure

From Table III we can see that once that **SimilarityBase algorithm** failed to find similarity using the AKT ontology. AQUA offers 15 reformulations using WordNet senses. However, some of them are not relevant to the context of the query.

5 Experiment

Our goal was to carry out an experiment to test the human judgment about similarity. We set up our experiment involving 10 people which was asked to pick related words from a list. We asked to indicate similarity between verbs. The task was performed without and with context. Our participants were 10 people ranging in edge from 25 to 60 years old. Some of them were native speakers. None of people who participated in our experiment had a significant background on Linguistics. Table IV shows the options selected by each person in the first part of our experiment (without context).

5.1 Experiment without context

In the first part of our experiment we try to find out which option is the most similar to the given item "across the board" (i.e. in all contexts where the given item may possibly appear). The experiment showed that without context people found hard to find a similar work without the context where the word appeared. Results from this first part of our experiment are shown in IV.

Question	Pair of terms	Human similarity ratings	Similarity value
1	locate – situate locate-site	(6/10) (3/10)	0.6 0.33
2	loosen-relax loosen-untie loosen-undo	(4/10) (4/10) (1/10)	0.4 0.4 0.1
3	work-for- employ work-for –exercise work-for-involve	(5/10) (2/10) (2/10)	0.5 0.2 0.2
4	unfold-spread unfold- extend	(6/10) (4/10)	0.6 0.4
5	relate- associate relate-connect	(8/10) (2/10)	0.8 0.2
6	participate- take-part	(10/10)	1
7	employ-use employ-utilize employ-hire	(5/10) (2/10) (1/10)	0.5 0.2 0.1
8	publish-issue publish- release publish-write	(3/10) (6/10) (1/10)	0.3 0.6 0.1
9	associate-relate associate-link-up associate-connect	(6/10) (1/10) (1/10)	0.6 0.1 0.1
10	award-present award-grant award-allocate award-reward	(2/10) (2/10) (1/10) (5/10)	0.2 0.2 0.1 0.5

Table IV. Summary of human similarity ratings

5.2 Experiment using context

We also performed the experiment with context. Results from this second part of our experiment are shown in Table V. This time people found easier than the first part of the experiment.

Question	Pair of terms	Human similarity ratings	Similarity value

1	locate – situate locate-site locate-settle	(5/10) (4/10) (1/10)	0.5 0.4 0.1
2	loosen-relax loosen-untie	(9/10) (1/10)	0.9 0.1
3	work-for - employ work-for –exercise work-for --body-of-work	(5/10) (2/10) (2/10)	0.1 0.8 0.1
4	unfold-spread	(10/10)	1
5	relate- associate relate-connect	(4/10) (6/10)	0.4 0.6
6	participate- take-part participate-enter	(9/10) (1/10)	0.9 0.1
7	employ-utilize employ-hire	(1/10) (9/10)	0.1 0.9
8	publish-issue publish-release publish-write	(2/10) (4/10) (4/10)	0.2 0.4 0.4
9	associate-relate associate-link-up associate-connect	(7/10) (2/10) (1/10)	0.7 0.2 0.1
10	award-grant	(10/10)	1

Table V. Summary of human similarity ratings when context was given

The outcome from the experiment was that the context helped people to select the correct answer. The set of possibilities was reduced by using context.

6 Related Work

In this section we describe only the most relevant work tha we fpound in similarities measures. (i.e. the closest in spirit to our algorithm of similarity driven by an ontology).

Bulskov et al. (Bulskov et al 2002) proposed a similarity measurement which is based on distance in ontology. The graph of ontology has weights associated to the edges. Then, the measure of similarity between two concepts was defined as the product of weights on the paths connecting the two atomic concepts.

The similarity defined in (Wu et al. 1994) relies in the structure of the ontology (it requires has to be a lattice). The similarity of two concepts is computed by how closely they are related in the hierarchy. In our case we do not require as a necessary condition that the taxonomy be a lattice.

The word reported by Resnik (Resnik 1995) describes semantic similarity in a taxonomy based in the idea of information content. The author claimed that method is sensitive to the problem of varying link distances. A similarity measure which takes in account such each class contribute information instead content of a single concept that maximize information content.

Later Resnik and Diag (Resnik and Diab 2000) reported a verb similarity which is different from the problem of noun similarity because verbs are generally viewed as possessing properties such as subcategorization restriction, selectional preferences and event structure. In our algorithm presented in section 2 we also have assessed similarity in relations which carry some semantic meaning and have constraint about the kind of arguments which can take as arguments.

7 Conclusions and Future Work

We had presented a similarity algorithm for relations. The main role of our similarity algorithm is to be part of a question answering system called AQUA developed at the Open University in England, UK. Our similarity algorithm plays an important role in query reformulation. We want to ensure that the query succeed even if the name in relation (obtained from the user query) is different form the relations in the ontologies. Then query is reformulated using the most similar relation which is suggested by our **similarityBase algorithm**. The similarity algorithm uses contextual neighbourhood and evidential information about arguments in the query. Future work includes to provide AQUA with a library of similarity algorithms. Our similiratyBase algorithm only works if AQUA is used as a closed-domain question answering system. However, if we use AQUA as an open-domain question answering we find similar words using WordNet. The main drawback in using WordNet is that WordNet offers all synsets even the one which are not related to the context.

We assessed the human ratings for similarity. The experimented was done in two parts. In the first part of our experiment we try to find out which option is the most similar to the given item "across the board" (i.e. in all contexts where the given item may possibly appear). Whilst in the second part context was provided. By presenting the first item in a sentence actually help disambiguate the task.

Ongoing work (at the Knowledge Media Institute, KMi) is in the direction of the use of LSA to assess similarity between concepts (according to common features). Preliminary results seems to be confirming the human judgment about similarity between two concepts (Burek et al. 2004). However, more research need to be done in this direction .

Acknowledgments

This work was funded by the Advanced Knowledge Technologies (AKT) Interdisciplinary Research Collaboration (IRC), which is sponsored by the UK Engineering and Physical Sciences Research Council under grant number GR/N157764/01. The authors would like to thank Mark Gaved for his invaluable help in reviewing the first draft of this paper. We want to thank Gaston Burek for his help in the formalization of the similarity algorithm and Harriet Cornish for improving pictures in this paper.

References

Budanitsky A. and Hirst G. (2001) Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. Workshop on WordNet and Other Lexical Resources, Second meeting of the North American Chapter of the Association for Computational Linguistics, Pittsburgh, June 2001.

H. Bulskov, R. Knappe, and T. Andreasen (2002): On Measuring Similarity for Conceptual Querying. Proceedings of Flexible Query Answering Systems 5th International Conference, FQAS 2002. LNAI 2522, pp. 100-111 (Eds.) T. Andreasen, A. Motro, H. Christiansen, H.L. Larsen (Eds.)
Copenhagen, Denmark, October 27-29, 2002.

Dekang L. An information-Theoretic Definition of Similarity (1998): Proceedings of International Conference on Machine Learning, Madison, Wisconsin, 1998.

Frakes W. and Baeza-Yates R. (1992): Information Retrieval: Data Structures & Algorithms, Prentice Hall, 1992.

Burek G., Vargas-Vera M. and Moreale E. (2004): Indexing Student Essays Paragraphs Using LSA Over an Integrated Ontological Space. Workshop on eLearning for Computational Linguistics and Computational Linguistic for eLearning. COLING 2004, Geneve 28th August 2004.

Knappe R. and Bulskov H. and Andreasen T. (2003): On Dimilarity Measures for Concept –based Querying. International Fuzzy Systems Association IFSA-2003 , LNAI , Istanbul, Turkey June29-July 2, 2003.

Miller G. A. and Charles W. G. (1991): Contextual correlates of semantic similarity. Language and Cognitive Processes , 6(1), 1-28.

Resnik P. (1995): Using information content to evaluate semantic similarity in a taxonomy. In Proceedings of IJCAI-95, pages 448-453, Montreal, Canada.

Resnik P. (1998): Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. Journal of Artificial Intelligence research.

Resnik P and Diab M. (2000): Measuring Verb Similarity. Proceedings of the 22nd Annual Meeting of the Cognitive Science Society (COGSCI2000), Philadelphia, USA, 2000.

Tversky A. (1977): Feautures of similarity. Psychological Review, 84, 327-352.

Vargas-Vera M. and Motta E. (2004): AQUA - Ontology-based Question Answering System. Accepted to be presented at Third International Mexican Conference on Artificial Intelligence (MICAI-2004), LNCS 2972, Springer Verlag, Editor R. Monroy et al. April 26-30, 2004, ISBN 3-540-21459-3.

Vargas-Vera M. and Motta E and Domingue J. (2003a): AQUA: An Ontology-Driven Question Answering System. AAAI Spring Symposium, New Directions in Question Answering, Stanford University, March 24-26, 2003.

Vargas-Vera M. and Motta E. and Domingue J. (2003b): An Ontology-Driven Question Answering System (AQUA). KMI-TR-129, Knowledge Media Institute, The Open University, 2003.

Wu Z. and Palmer M. (1994): Verb semantics and Lexical Selection. 32nd Annual Meetings of the Association for Computational Linguistics, 1994.

Appendix I

Exercise. Given the word at the top of each section, select the option which is the most similar to it with respect to meaning. Table A shows the set of questions used in our experiment without context described in section 5.

Question	Offered options
Question 1 **Locate**	a) settle b) turn-up c) site d) situate
Question 2 **Loosen**	a) relax b) untie c) undo d) tease
Question 3 **work-for**	a) employ b) body-of-work c) exercise d) involve
Question 4 **Unfold**	a) spread b) stretch c) extend d) divorce
Question 5 **Relate**	a) colligate b) associate c) connect d) concern
Question 6	a) enter

Participate	b) take-part c) act d) move
Question 7 **Employ**	a) use b) utilize c) apply d) hire
Question 8 **Publish**	a) bring-up b) issue c) release d) write
Question 9 **associate**	a) link-up b) relate c) affiliate d) connect
Question 10 **Award**	a) present b) grant c) allocate d) reward

Table A. Summary of human similarity ratings

Appendix II

Exercise. Identify the word with similar meaning (to the one written in bold) in each sentence. Table B shows the set of questions used in our experiment using context described in section 5.

Question with context	**Offered options**
Question 1 City plans **locate** noisy factories away from houses	a) settle b) turn-up c) site d) situate
Question 2 The massage will **loosen** your muscles.	a) relax b) untie c) undo d) tease
Question 3 **Work** keeps you healthy!	a) employ b) body-of-work c) exercise d) involve
Question 4 **Unfold** the newspaper.	a) spread b) stretch c) extend d) divorce
Question 5 The statistics **relate** crime to poverty.	a) colligate b) associate c) connect d) concern
Question 6 John will **participate** in the raffle.	a) enter b) take-part c) act d) move
Question 7 We **employ** John to do some essay marking.	a) use b) utilize c) apply d) hire
Question 8 The new researcher will **publish** a book on Ecology.	a) bring-up b) issue c) release d) write
Question 9 We want to **associate** results from the akt project to the Dotkom project.	a) link-up b) relate c) affiliate d) connect
Question 10 The European Community **award** was substantial!	a) present b) grant c) allocate d) reward

Table B. Summary of human similarity ratings using context