ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA
CONSIGLIO NAZIONALE DELLE RICERCHE

A. Formica

CONCEPT SIMILARITY BY EVALUATING INFORMATION CONTENTS AND FEATURE VECTORS: A COMBINED APPROACH

R. 642 Giugno 2006

Anna Formica – Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti" del CNR, Viale Manzoni 30 - 00185 Roma, Italy. Email : anna.formica@iasi.cnr.it

This paper appears in Communications of the ACM, 52(3), pp.145-149, 2009.

ISSN: 1128–3378

Collana dei Rapporti dell'Istituto di Analisi dei Sistemi ed Informatica, CNR

viale Manzoni 30, 00185 ROMA, Italy

tel. ++39-06-77161
fax ++39-06-7716461
email: `iasi@iasi.rm.cnr.it`
URL: `http://www.iasi.rm.cnr.it`

Abstract

Evaluating semantic similarity of concepts is a problem that has been extensively investigated in the literature in different areas, such as Artificial Intelligence, Cognitive Science, Databases and Software Engineering. Currently, it is growing in importance in different settings, such as digital libraries, heterogeneous databases and, in particular, the Semantic Web. In such contexts, very often concepts are organized according to a taxonomy (or a hierarchy) and, in addition, are associated with structures (also referred to as *feature vectors*). With this regard, in general, the concept similarity measures proposed in the literature have not been conceived to address both these levels of information (taxonomy and structure), i.e., there exist contributions focusing on the similarity of hierarchically related concepts, and other proposals conceived to compare concept feature vectors. In this article, a method for evaluating similarity of concepts is presented, where both concept taxonomy and concept structures are considered. In particular, such a method has been defined by combining and revisiting (i) the *information content* approach introduced in [9], and further refined in [6], with regard to the comparison of concepts within the taxonomy, and (ii) a method inspired by the *maximum weighted matching* problem in bipartite graph [4], with regard to the comparison of feature vectors. The proposed approach is then compared with two among the most representative similarity measures defined in the literature, and a small data set shows how the proposed measure allows us to reduce the gap existing between them.

1. Introduction

Let us consider a very general knowledge base (KB), essentially defined by a set of concepts that are organized according to a generalization (*ISA*) hierarchy, where each concept may be associated with a structure, or a feature vector, containing the properties describing the concept. Note that the proposed method can be suitably refined to deal with specific KBs, as for instance, Object-Oriented KBs, Geographical KBs, XML-Schemas, GML-schemas, etc... In this article, a very simple and abstract data model has been addressed in order to present the essence of the method, without dealing with aspects that pertain to specific contexts.

A concept has a *name*, an enumerated set of *super-concepts* (taxonomic information), and a tuple of *typed properties*[1] (structural information). For instance, consider the following set of concepts:

```
person = {name:string,SSN:string}
student = ISA(person) {college:string}
worker = ISA(person) {EIN:string,salary:integer}
machine = {name:string,maker:string}
railcar = ISA(machine) {VIN:string,owner:person}
....
```

where *SSN*, *EIN*, and *VIN* stand for *Social Security Number*, *Employer Identification Number*, and *Vehicle Identification Number*, respectively. Therefore, a concept has a left hand side, defined by the name of the concept, and a right hand side containing the hierarchical and/or structural information. For instance, in the case of the concept of name *person*, on the right hand side only a structural information is present, i.e., two typed properties, namely *name* and *SSN*, both of type *string*. In the case of *student*, in addition to the structural information (the property *college* of type *string*), we also have a taxonomic information, expressed by the *ISA* construct. This means that *student* has a super-concept, namely *person*, whose typed properties will be *inherited*. *Inheritance* is a well-known problem that has been extensively investigated in the literature. In the case of the set of concepts above, after inheritance, the following holds:

```
person = {name:string,SSN:string}
student = {name:string,SSN:string,college:string}
worker = {name:string,SSN:string,EIN:string,salary:integer}
machine = {name:string,maker:string}
railcar = {name:string,maker:string,VIN:string,owner:person}
...
```

where all the *ISA* constructs have been removed, and concepts are expressed in terms of their typed properties only. Note that properties can be typed with atomic types (such as *string*, or *integer*) or concept names, as in the case of the property *owner* of *railcar*, that is typed with *person*.

[1] Typeless data models are also noteworthy in order to evaluate this proposal. In this paper, typing has been addressed to provide a method that is more easily adaptable to Semantic Web languages, as for instance XML-Schemas.

In order to keep trace, after inheritance, of the *ISA* relations defined in the original set of concepts, the previous set of concepts is coupled with the *concept hierarchy* of Figure 1. Therefore, such a hierarchy contains, for instance, an arc between *student* and *person*, since *person* was a super-concept of *student*. Note that it also includes properties, such as *college* or *salary*, that are concept as well, i.e., their names can be associated with - possibly empty - sets of super-concepts and typed properties. Furthermore, suppose it also contains arcs among VIN, SSN, EIN and *id_number* (*identification number*). The root of the concept hierarchy is labeled by *Top* which represents the most general concept. Dotted lines stand for paths of arbitrary length.

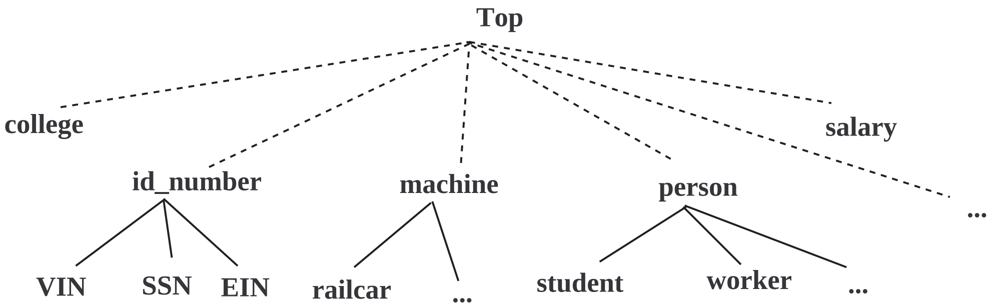

Figure 1: Concept hierarchy

2. Information Content Similarity

Traditionally, in order to evaluate the semantic similarity of hierarchically related concepts, the *edge-counting* approach is adopted. This approach is essentially based on the distance between nodes of the taxonomy, that is, the shorter the path, the more similar the concepts these nodes represent. However, the main drawback of this approach is that links in the taxonomy do not represent uniform distances. In fact, the level of refinement between, for instance, the concept *safety valve* and *valve* is not comparable to the level of refinement between, for instance, *knitting machine* and *machine* [9]. For this reason, a different approach, referred to as the *information content* approach, has been proposed in the mentioned paper, and successively refined in [6], which does not depend on path lengths. It is based on the association of probabilities with the concepts of the hierarchy. In particular, the *probability* of a concept c is defined as:

$p(c) = \frac{freq(c)}{M}$

where $freq(c)$ is the *frequency* of the concept c estimated using noun frequencies from large text corpora, as for instance the *Brown Corpus of American English*, and M is the total number of observed instances of nouns in the corpus. In our example, probabilities have been assigned according to the *SemCor* project [3], which labels subsections of the *Brown Corpus* to senses in the *WordNet* lexicon [10]. According to *SemCor*, the total number of observed instances of nouns is $88,312$. Below the definitions of some of the previous concepts are given, and the related frequencies (the numbers in parenthesis):

- (7229) person – a human being;
- (68) student – a learner who is enrolled in an educational institution;

- (289) worker – a person who works at a specific occupation;
- (600) machine – any mechanical or electrical device that transmits or modifies energy to perform or assist in the performance of human tasks;
- (24) railcar – a wheeled vehicle adapted to the rails of railroad;
- (45) identification number (id_number) – a numeral or string of numerals that is used for identification;
- (1) VIN – Vehicle Identification Number;
- (1) SSN – Social Security Number;
- (1) EIN – Employer Identification Number;
- (10) salary – ....

Once probabilities have been associated with concepts, we obtain the *weighted concept hierarchy* of Figure 2, where *Top*, the most general concept, has probability 1. Note that each concept that occurs in the corpus is counted as an occurrence of each more abstract concept up in the *ISA* hierarchy. For instance, in Figure 2, an occurrence of *worker* is counted toward the frequency of *worker* and *person* (for further details, see [9]).

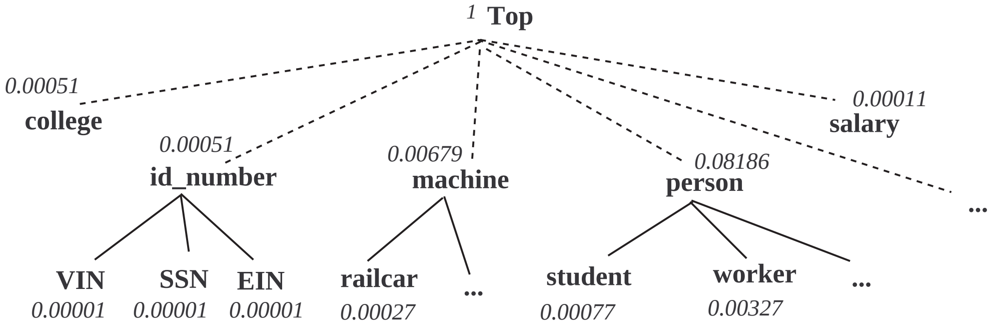

Figure 2: Weighted concept hierarchy

By following the standard argumentation of information theory, the *information content* of a concept c is defined as:

$$-\log p(c)$$

that is, as the probability of a concept increases, the informativeness decreases, therefore the more abstract a concept, the lower its information content. For instance, in our example, the probability of the concept *person* is greater than the probability of *student*, therefore the information content of *person* is less than that of *student* (in fact, with respect to *person*, *student* has the additional property *college*).

In line with [6], we define the notion of *information content similarity* (*ics*) of two concepts c_1,c_2 as follows:

$$ics(c_1, c_2) = \frac{2 \log p(c)}{\log p(c_1) + \log p(c_2)}$$

where c is the concept providing the maximum information content shared by c_1 and c_2 in the taxonomy, i.e., the more information two concepts share, the more similar they are. Note that c is the upper bound of c_1,c_2 in the taxonomy whose information content is maximum, i.e., when defined, the least upper bound.

For instance, consider the concepts *student* and *worker.* The maximum information content shared by these concepts is provided by *person*, that is their least upper bound in the concept hierarchy. Therefore, their *ics* is the following:

$ics(student, worker) = \frac{2 \log p(person)}{\log p(student)+\log p(worker)} = \frac{2*3.61}{10.34+8.26} = 0.39$

Analogously, in the case of *SSN* and *EIN*, the following holds:

$ics(SSN, EIN) = \frac{2 \log p(id\_number)}{\log p(SSN)+\log p(EIN)} = \frac{2*10.94}{16.43+16.43} = 0.67$

and, if we consider the concepts *student* and *railcar*, the *ics* is null since:

$ics(student, railcar) = \frac{2 \log p(Top)}{\log p(student)+\log p(railcar)} = \frac{2*0}{10.34+11.85} = 0.$

Note that, given a concept (or a property) c, $ics(c, c) = 1$, where c can also be a type such as *integer* or *string*, whereas, for instance, $ics(string, integer) = 0$, since we assume that the maximum information content shared by *integer* and *string* in the concept taxonomy is *Top*. Furthermore, given two concepts c_1,c_2 that are synonyms (for instance according to *WordNet*), $ics(c_1, c_2) = 1$ (we assume they label the same node in the hierarchy).

3. Feature Vector Similarity

Concept structures are compared according to a method inspired by the *maximum weighted matching* problem in bipartite graphs, that has been revisited in [4]. In essence, given two concept feature vectors, we have to identify one or more sets of pairs of typed properties that maximize the sum of the products of the *ics* of the properties and the *ics* of the related types. Note that such a sum is evaluated within all the sets of pairs of typed properties such that in the set there are no two pairs sharing an element. For instance, given two concepts, assume that their sets of typed properties represent a set of boys and a set of girls, respectively. A selected set of pairs is a possible set of marriages, when polygamy is not allowed.

In our example, consider the concepts *student* and *worker* defined above. As shown in Figure 3(a), two sets of pairs can be defined such that the above mentioned sum is maximal. These sets are obtained by pairing *name* with *name*, since they are both of type *string*, *SSN* with *SSN*, both typed again with *string*, whereas they differ for the third pair that, in one set, is defined by *college* and *EIN* and, in the other set, by *college* and *salary*, since in both cases their *ics* are null (in fact, according to the taxonomy of Figure 2, the least upper bound between *college* and *EIN* is *Top*, and the same holds in the case of *college* and *salary*).

Note that, as mentioned before, the maximal sum is computed by multiplying the *ics* of the properties with the *ics* of the related types. For this reason, the role of typing is fundamental. Suppose for instance that *worker* has the property *SSN* of type *integer*, rather than of type *string.* In this case, the two sets of pairs with maximal sum are obtained by pairing *name* with *name* as above, whereas *SSN* of *student* is paired with *EIN* of *worker* (since we have seen that their *ics* is equal to 0.67 and their types are both *string*), and *college* is paired with *SSN*, in one set, and with *salary*, in the other set (since in both cases the have null *ics*).

Once identified one set of pairs of typed properties such that the sum of the *ics* is maximal, the sum is normalized. For instance, in our example, it is divided by 4 (that is the highest between the cardinalities of the sets of properties of *student* and *worker*). Therefore we have

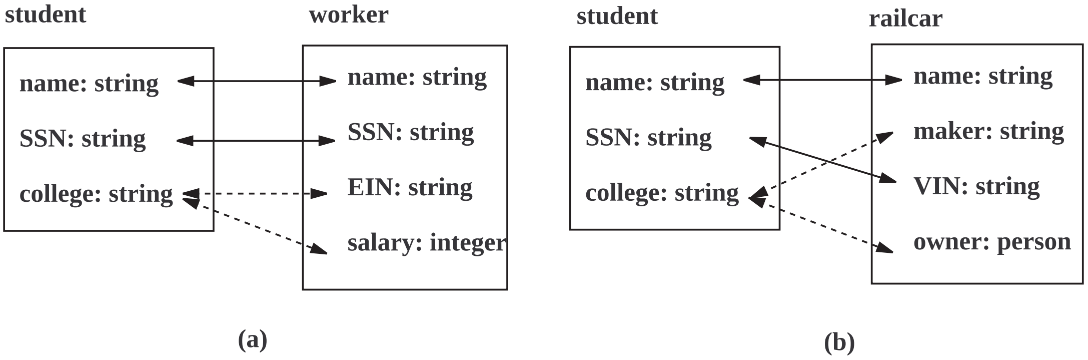

Figure 3: Feature vector matching between: (a) *student* and *worker*, (b) *student* and *railcar*

the notion of *feature vector similarity* (fvs). In particular, if we consider *student* and *worker* the following holds:

$fvs(student, worker) = \frac{1+1+0}{4} = 0.50.$

Let us now compare *student* and *railcar*. As shown in Figure 3(b), *name* is paired with *name*; SSN is paired with VIN since according to the taxonomy of Figure 2, their *ics* is equal to 0.67 (it is computed similarly to the pair SSN, EIN shown before) and, as in the previous case, *college* can be paired with *maker* or *owner* indifferently. Therefore, in this case:

$fvs(student, railcar) = \frac{1+0.67+0}{4} = 0.42$

where 4 is the cardinality of the set of properties of *railcar*, that is greater than that of *student*.

4. Concept Similarity and Related Work

The *ics* and the fvs are then combined to obtain the notion of concept similarity (*Sim*). Given two concepts c_1,c_2, $Sim(c_1, c_2)$ is essentially given by a weighted average between the *ics* and the fvs, as defined below:

$Sim(c_1, c_2) = ics(c_1, c_2) * w + fvs(c_1, c_2) * (1 - w)$

where w is a weight that is established by the domain expert according to the cases, such that $0 \leq w \leq 1$. For instance, in our example, by assuming $w = \frac{1}{2}$, we have:

$Sim(student, worker) = ics(student, worker)\frac{1}{2} + fvs(student, worker)\frac{1}{2} =$
$\frac{1}{2}(0.39 + 0.50) = 0.45$

and:

$Sim(student, railcar) = ics(student, railcar)\frac{1}{2} + fvs(student, railcar)\frac{1}{2} =$
$\frac{1}{2}(0 + 0.42) = 0.21.$

Regarding the other proposals, we have to distinguish the approaches aiming at evaluating semantic similarity of (i) hierarchically organized concepts (hierarchy-based approaches) and (ii) concept structures (feature-based approaches).

In the case of hierarchy-based approaches, we have already mentioned that the drawback of the traditional edge-counting approach has been overcome by the information content approach proposed by *Resnik* [9], further refined by *Lin* [6]. In particular, it is important to note that by following the *Resnik*'s approach, concept similarity is simply given by the maximum information content shared by the comparing concepts, whereas the *Lin*'s approach takes also into account the information contents of the comparing concepts. Other approaches have been proposed in the literature as, for instance, that of *Wu&Palmer* [11]. However, here the *Lin*'s proposal has been selected since it is widely acknowledged in the literature that it shows a higher correlation with human judgements with respect to other proposals, including that of *Resnik* [5].

Regarding feature-based approaches, most of the contributions defined in the literature adopt the *Dice*'s function [7]. Essentially, with respect to the fvs proposed here, *Dice*'s function allows concept similarity to be computed without explicitly considering the similarity degree of properties (that in our case is represented by the *ics*). In particular, given two concepts, say c_1,c_2, each described by a feature vector, say $F(c_1)$,$F(c_2)$, respectively, their similarity ($Dice(c_1, c_2)$) is defined as follows:

$Dice(c_1, c_2) = \frac{2|A(c_1,c_2)|}{|F(c_1)|+|F(c_2)|}$

where:

$A(c_1, c_2) = \{(a, b) \mid a \in F(c_1), b \in F(c_2), (a, b) \in \mathit{Aff}\}$

Aff is the set of pairs of concepts showing affinity such that, similarly to the *maximum weighted matching* problem in bipartite graphs, each feature in $F(c_1)$ and $F(c_2)$ can participate at most in one affinity pair (in the case a feature participates in more than one pair, a pair with maximal affinity value is selected and the remaining pairs are discarded), and $|A(c_1, c_2)|$, $|F(c_1)|$, and $|F(c_2)|$ are the cardinalities of the sets $A(c_1, c_2)$, $F(c_1)$, and $F(c_2)$, respectively.

Consider for instance, *student* and *railcar*. The number of pairs of concepts (properties) showing affinity is 2, and in particular they are (*name*,*name*) and (SSN,VIN). Therefore, we have:

$Dice(student, railcar) = \frac{2*2}{3+4} = 0.57$

It is important to note that, according to the *Dice*'s approach, the affinity of the pair (*name*,*name*) is equivalent to the affinity of the pair (SSN,VIN). This does not hold in the case of the fvs proposed in this article since, as shown before, it is defined on the basis of the *ics* of the pairs of properties, i.e., $ics(name, name) = 1$, and $ics(SSN, VIN) = 0.67$.

It is worth mentioning that, within feature-based approaches, vector based measures have been defined, such as the *cosine* or the *Jaccard* measures [1]. These are mainly used in Information Systems to deal with documents, and are based on vector space models for which semantic similarity of documents is represented by proximity in a vector space. Furthermore, it is worth recalling that in [12], the *SOQA-SimPack Toolkit* has been presented, aiming at evaluating similarities within a given ontology, and between concepts of different ontologies. In particular, it provides a generic and extensible library of ontological similarity measures in order to capture various notions of similarity.

5. Evaluation of the Method

In Table 1, the similarity scores of some of the pairs of the concepts of our running example have been given according to the approaches of *Lin*, *Dice* and *Sim*. It is possible to see how *Sim* is a combination of the *Lin*'s approach, that has been conceived to evaluate semantic similarity of hierarchically organized concepts, without addressing their structures, and the *Dice*'s function that focuses on concept feature vectors, ignoring the taxonomic information. For instance, consider again the similarity between *student* and *railcar*. It is null according to *Lin*, since the maximum information content shared by the concepts in the taxonomy is *Top*, whereas, according to *Dice*, *student* and *railcar* show a discrete affinity (0.57) due to their structures.

Table 1: Comparison among three different similarity measures

	Lin	*Dice*	*Sim*
$(student,\ worker)$	0.39	0.57	0.45
$(student,\ railcar)$	0	0.57	0.21
$(student,\ person)$	0.52	0.80	0.60
$(person,\ machine)$	0	0.50	0.25
$(person,\ railcar)$	0	0.67	0.21

By taking into account both the taxonomy and the structure, on the basis of the *Sim* measure, we obtain an approximately average value, i.e., $Sim(student, railcar) = 0.21$.

In order to evaluate the *Sim* measure, some considerations have to be done, starting with the problem of "ideal values". In general, ideal values are established according to human judgement and, in the literature, often the similarity scores assigned by human subjects in the *Miller&Charles* experiments are addressed (where 28 selected pairs of concepts have been analyzed and one score - on a scale of 0 to 4 - has been given for each pair) [8]. By using the *Miller&Charles* scores, it has already been shown in [6], and also by other authors, see for instance in [5], that *Lin*'s method shows a higher correlation with human judgement than other hierarchy-based approaches, e.g. *Resnik* [9] and *Wu&Palmer* [11]. However, in different contexts, such as conceptual schema analysis, information sharing from multiple heterogeneous sources, intelligent information integration and, more recently, within Web service discovery etc.. [2], the feature-based approach is preferred which takes into account the heterogeneity of the concept features. In particular, the *Dice*'s function recalled above is adopted, whose similarity scores, in most of cases, are significantly different from the ones obtained according to *Lin*, as shown for instance in Table 1[2]. For this reason, the *Sim* measure has been proposed in order to reduce the gap existing between the two approaches (leaving maximum flexibility to the domain expert in assigning specific weights).

References

[1] R.Baeza-Yates, B.d.A.Ribeiro-Neto; *Modern Information Retrieval*; ACM Press, 1999.

[2] D.Bianchini, V.De Antonellis, M.Melchiori; *Capability Matching and Similarity Reasoning in Service Discovery*; M.Missikoff and A.De Nicola (Eds.), Proc. of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability (EMOI-INTEROP), 13-14 June, Porto, Portugal, CEUR-WS.org, 2005.

[3] C.Fellbaum; *A Semantic Network of English: the Mother of all WordNets*; Computers and the Humanities 32, 209-220, 1998.

[4] A.Formica, M.Missikoff; *Concept Similarity in SymOntos: an Enterprise Ontology Management Tool*; The Computer Journal, 45(6), 583-594, 2002.

[5] J. J.Jiang, D. W.Conrath; *Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy*; The Computing Research Repository (CoRR), cmp-lg/9709008, 1997.

[2] Note that the simple data set shown in Table 1 does not want to provide any evaluation of the method. It has been defined to show an example about the gap existing between the hierarchy- and feature-based approaches.

[6] D.Lin; *An Information-Theoretic Definition of Similarity*; Proc. of the Int. Conference on Machine Learning (ICML), Madison, Wisconson, USA, July 24-27, 1998, Morgan Kaufmann, 296-304, 1998.

[7] Y.S.Maarek, D.M.Berry, G.E.Kaiser; *An Information Retrieval Approach For Automatically Constructing Software Libraries*; IEEE Transactions on Software Engineering, 17(8), 800-813, 1991.

[8] G.A.Miller, W.G.Charles; *Contextual correlates of semantic similarity*; Language and Cognitive Processes, 6(1), 1-28, 1991.

[9] P.Resnik; *Using Information Content to Evaluate Semantic Similarity in a Taxonomy*; Proc. of the Int. Joint Conference on Artificial Intelligence (IJCAI), Montreal, Quebec, Canada, August 20-25, 1995, Morgan Kaufmann, 448-453, 1995.

[10] *WordNet 2.1: A lexical database for the English language*; http://www.cogsci.princeton.edu/cgi-bin/webwn, 2005.

[11] Z.Wu, M.Palmer; *Verb semantics and lexical selection*; Proc. of the 32nd Annual Meeting of the Associations for Computational Linguistics; Las Cruces, New Mexico, 133-138, 1994.

[12] P.Ziegler, C.Kiefer, C.Sturm, K.R.Dittrich, A.Bernstein; *Detecting Similarities in Ontologies with the SOQA-SimPack Toolkit*; Proc. of International Conference on Extending Database Technology (EDBT), 26-31 March 2006, Munich, Germany, 59-76, 2006.