CP/CV: Concept Similarity Mining without Frequency Information from Domain Describing Taxonomies *

Jong Wook Kim
Comp. Sci. and Eng. Dept.
Arizona State University, Tempe, AZ 85287
jong@asu.edu

K. Selcuk Candan
Comp. Sci. and Eng. Dept.
Arizona State University, Tempe, AZ 85287
candan@asu.edu

ABSTRACT

Domain specific ontologies are heavily used in many applications. For instance, these form the bases on which similarity/dissimilarity between keywords are extracted for various knowledge discovery and retrieval tasks. Existing similarity computation schemes can be categorized as (a) structure- or (b) information-based approaches. Structure-based approaches compute dissimilarity between keywords using a (weighted) count of edges between two keywords. Information-base approaches, on the other hand, leverage available corpora to extract additional information, such as keyword frequency, to achieve better performance in similarity computation than structure-based approaches. Unfortunately, in many application domains (such as applications that rely on unique-keys in a relational database), frequency information required by information-based approaches *does not exist*. In this paper, we note that there is a third way of computing similarity: if each node in a given hierarchy can be represented as a vector of related concepts, these vectors could be compared to compute similarities. This requires mapping concept-nodes in a given hierarchy onto a concept-space. In this paper, we propose a concept propagation (CP) scheme, which relies on the semantical relationships between concepts implied by the structure of the hierarchy to annotate each concept-node with a concept-vector (CV). We refer to this approach as CP/CV. Comparison of keyword similarity results shows that CP/CV provides significantly better (upto 33%) results than existing structure-based schemes. Also, even if CP/CV does not assume the availability of an appropriate corpus to extract keyword frequency information, our approach matches (and slightly improves on) the performance of information-based approaches.

*This work is supported by an NSF ITR Grant, ITR-0326544; "*ILearn: IT-enabled Ubiquitous Access for Educational Opportunities for Blind Individuals*" and an RSA Grant "*Ubiquitous Environment to Facilitate Access to Textbooks and Related Materials for Adults and School Age Children who are Blind or Visually Impaired*".

CIKM'06, November 5–11, 2006, Arlington, Virginia, USA.

Categories and Subject Descriptors: H.3.1 [INFORMATION STORAGE AND RETRIEVAL][Content Analysis and Indexing]: Indexing methods

General Terms: Algorithms, experimentation, human factors.

Keywords: Mining keyword similarities, concept hierarchies, concept propagation.

1. MOTIVATION

Recently, there has been growing research on integrating keyword search into databases. Kacholia *et al.* [15], BANKS [5], XRank [13], ObjectRank [4], and XSEarch [7] all rely on structural analysis, information retrieval techniques, as well as keyword similarities for ranking database query results. Naturally a particular challenge of these and other techniques, which aim to apply keyword search on domain specific databases, is that keyword (or concept) similarities in specific domains need to be properly measured for these to be effective.

Ontologies and taxonomies are used in diverse areas of science, including biology and medicine, as well as in various standardization and information integration efforts, where it is important to be precise about the relationships of concepts. Given a concept taxonomy for a particular application domain, software systems can represent and organize data pertinent to this domain more effectively than without any prior knowledge of the relationships between concepts in this domain. Furthermore, ontologies enable sharing and integration of data from different domains and data sources. The effectiveness of ontologies in enabling domain-specific treatment of knowledge-application, lead to proliferation of domain specific ontologies, such as UMLS [1] (for medical concepts), TOVE [12] (for enterprise modeling), PLINIUS [37] (for material science) and GENSIM [16] (for molecular biology and biochemistry). Our goal in this paper is to develop an effective measure of similarity between concepts in such a concept taxonomy.

1.1 Mining Concept Similarities

Semantic similarity measures quantify relatedness between two words or concepts. Many traditional knowledge-driven applications (such as text classification [33], word sense disambiguation [2], and data mapping [29, 6]) require mining of such semantic similarity/ dissimilarity values between concepts in a given domain. Therefore, the study of semantic relationships between words in a language has a long history in psychological theory, natural language processing, and knowledge management. There are various general pur-

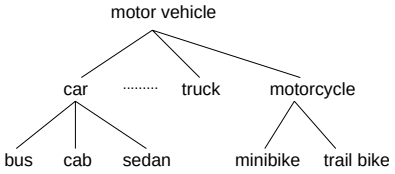


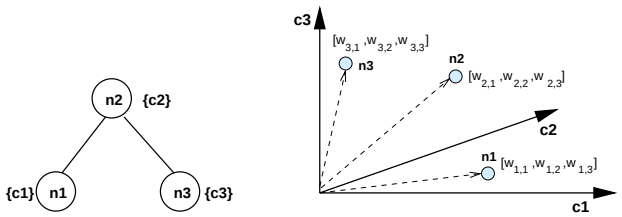
Figure 1: A concept (IS-A) hierarchy: "bus" and "cab" are more closely related than "bus" and "minibike".

pose efforts, such as WordNet [22, 23] and FrameNet [3], to model the lexical knowledge underlying a language in the form of a hierarchical taxonomy, where the structure of the graph represents the knowledge about the relatedness of the words. Intuitively, highly related words are grouped together and the path between two different concept-nodes in the hierarchy reflects how these are related in the real-world. The parent/child edges in these hierarchies may correspond to different relationships, such as IS-A or PARTS-OF. An example IS-A hierarchy is presented in Figure 1.

If we consider the WordNet segment presented in Figure 1, we can intuitively see that the two concepts, "*bus*" and "*cab"*, are more closely related to each other than concepts, "*bus*" and "*minibike*". In the last decade various measures for estimating the semantic similarity of keywords in a given taxonomy are proposed. These measures can be roughly categorized into *structure-based (or, edge-based)* methods and *information-based* methods. In structure-based methods, the semantic similarity between two words is measured by the shortest distance between them [26] or the sum of the edge weights along this shortest path [27]. Information-based methods leverage available corpora to extract additional information, such as keyword frequency, to achieve better similarity evaluation than those approaches that rely only on the structural analysis of a hierarchy. For example, [28] estimates the similarity between two concepts using the information content (i.e., negative logarithm of the probability of encountering an instance of the concept in the corpus) of the concepts subsuming them. When using [28], the similarity between "*bus*" and "*minibike*" in Figure 1 would be determined by the information content of the node "*motor vehicle*", which subsumes both words in the hierarchy.

1.2 Challenge: Measuring Domain Specific Concept Similarities without Frequencies

Generally speaking, given an appropriate corpus, information-based methods show a better performance than structure-only approaches. However, information-based methods need an appropriate (large and representative) corpus. Such a large corpora is usually available in text-retrieval applications: the collection of documents that is going to be indexed can be used to extract keyword frequency information. However, in many other applications, such a large corpora can not be expected to be readily available: for example, in applications that rely on a relational database, the concepts that are used as **unique-keys** in the data can not have frequency information that will support information-based methods for similarity computation. Even for non-unique attributes, the value frequencies (which can depend on how a schema is normalized) do not carry the same information as in text collections. In these cases, the similarities between these concepts have to be extracted from the available concept hierarchy or taxonomy. Therefore, in this paper, we focus on the challenge of learning concept similarities from a given concept hierarchy without having to rely on a large corpus for frequency information extraction.



(a) concept hierarchy (b) concept space

Figure 2: (a) An IS-A hierarchy where each node represents a concept; (b) the corresponding concept-space. The concept-vectors (CVs) representing where the nodes are mapped in the space are computed through concept-propagation, CP. In the proposed approach, similarities are measured using the concept-vectors in this concept space. The mapping from the concept hierarchy to the concept-space (i.e., concept propagation process) relies on the fact that *degrees of generality* computed in both representations should be identical

1.3 Contributions of this Paper

In this paper, we note that there is a third way of mining similarities of keywords/concepts: if each concept-node in a given hierarchy could be represented as a vector, then these vectors could be compared to compute concept similarity values (Figure 2). Based on this observation, we propose a concept-propagation (CP) scheme which leverages the semantical relationships between concept-nodes (implied by the structure of the concept hierarchy) to annotate each node with a concept-vector (CV). The CVs are then used for similarity computations. We refer to this novel approach as CP/CV. In particular,

- we propose a concept-vector (in a concept-space) representation of concept-nodes in a hierarchy (Section 3),
- we develop a method for quantifying the *degree of generality* of a concept-node relative to an other node in a hierarchy (Section 4),
- we present a method for quantifying the degree of generality of a concept-vector relative to an other vector in a concept-space (Section 5), and
- we introduce a concept propagation algorithm, based on the observation that *measuring degrees of generality between concept-nodes in a given hierarchy and in the corresponding concept-space should provide the same results* (Section 6).

The proposed CP/CV similarities between the concept-nodes are then computed in the resulting concept-space using cosine similarities [39] of the resulting concept-vectors. In Section 7, we experimentally evaluate the proposed CP/CV similarity mining technique. Comparison of similarity results shows that CP/CV provides significantly better (upto 33%) results than existing structure-based schemes. Furthermore, CP/CV matches (in fact slightly improves) the performance of information-based approaches, even though it does not have to assume the existence of a representative corpus with appropriate frequency information.

2. RELATED WORK

Here, we present the related work in the domains of mining semantic similarities of concepts in a taxonomy and spreading activation and propagation-based approaches used in Web indexing and mining.

Mining of Concept Similarities: There have been a number of proposals for measuring semantic similarities between keywords in a taxonomy. As mentioned in the introduction, these approaches can be classified into two categories: structure-based or information-based methods. [26] proposes that the conceptual distance between two concept-nodes should be defined as the shortest path between two nodes in the taxonomy and that this should satisfy metric distance properties. This approach proved to be very useful in small and specific domains, such as medical semantic nets. However, it ignores that (a) the semantic distance between neighboring nodes are not always equal and that (b) the varying local densities in the taxonomy can have strong impacts on the semantic distance between concept-nodes. To overcome these shortcomings, [27] associate weights to the edges in the hierarchy: the edge weight is affected both by its depth in the hierarchy and the local density in the taxonomy. To capture the effect of the depth, [38] estimates the conceptual distance between two concepts, c_1 and c_2, by counting the number of edges between them, and normalizing this value using the number of edges from the root of the hierarchy to the closest common ancestor of c_1 and c_2.

Information-based methods, on the other hand, measure the semantic similarity between two concepts based on the amount of information content of the common ancestors of two given concept-nodes [28, 19, 20, 14]. The information content of a concept is defined as the negative logarithm of the probability of encountering an instance of the concept in the given corpus. For example, in [28], similarity between two concepts, c_1 and c_2, is defined as

$$sim(c_1, c_2) = \max_{c \in Subsume(c_1,c_2)} [-\log P(c)],$$

where $Subsume(c_1, c_2)$ is the set of concepts that subsume c_1 and c_2, and $P(c)$ is the probability of encountering an instance of concept, c, in a corpus. Recently, [21] applied a similar approach for mining similarities of Web pages. Their approach is based on the information-based analysis of the hierarchy of Web pages, though they generalize their results to non-hierarchical Web data as well.

Propagation and Spreading Activation: Spreading activation is a general scheme used for propagation of knowledge on data represented in the form of graphs. In spreading activation, activation of one concept in a given node in the graph will spread to several or many related nodes. Spreading activation is used heavily in information retrieval [8] and Web mining [11]. For example, [11] presents a method to improve Web pages annotations using spreading activation (of available annotations) over the Web graph. In a work related to similarity mining, [18] proposes measuring the semantic similarity between words in a semantic network using spreading activation approach. Unlike our approach, however, [18] assumes that initial weights (frequency information obtained from a corpus) are available to be spread.

Propagation is also used in Web mining. One successful approach for organizing web query results based on available web structure is *topic distillation* proposed in [17]. The basic idea in topic-distillation is to consider the structure of the Web and propagate scores between pages in a way to organize topic spaces in terms of smaller sets of hub and authoritative pages. Other methods propagate the term frequency values or (given a query) the relevance score itself. For instance, given a query [34] propagates the relevance score between web pages connected with hyperlinks. [36, 32] on the other hand, propagate the term frequency values between neighboring pages. Recently, [25] proposes a generic relevance propagation framework, which brings together techniques from [34] and [36], as well as different propagation methods: hyperlink-level/sitemap-level and score-level/term-frequency-level propagation for indexing.

3. CONCEPT SPACE

In CP/CV, we map concept-nodes in a hierarchy into a concept-space for similarity computation. This section presents the vector interpretation of the concept-nodes on which the CP/CV approach is based.

3.1 Vector-Space and Similarity

In various domains, including text mining and information retrieval, concepts are usually represented as vectors in a feature (or keyword) space. For instance, *latent semantic indexing* (LSI) [9] and *principal component analysis* (PCA) [35], analyze the keywords of the documents in a corpus to identify (*mutually independent*) concepts that are dominant in the corpus. The resulting dominant concepts are represented as vectors in the keyword space.

In this work, we also aim to represent concept-nodes in a given hierarchy as vectors in a concept-space. Yet, there are fundamental differences between the above approaches and CP/CV: first of all, LSI and PCA extract concept vectors from term-document matrix (i.e., the corpus); secondly, the concepts identified through LSI and PCA are mutually independent (i.e., their mutual similarities are zero).

Note that one major advantage of the vector representation of *concepts* is that we can leverage various operations that are proven to be well supported within the vector-space model. One such operation, used successfully in the information retrieval and text mining literatures, is the use of cosine similarity to compute the similarity of two vectors [39].

3.2 Concept-Space and Concept-Vectors

Let $\mathcal{H}(N, E)$ denote a concept hierarchy, where N is the set of concept-nodes (corresponding to the concepts in the hierarchy) and E is the set of edges between the parent/child pairs in $\mathcal{H}$.

DEFINITION 3.1 (CONCEPT-SPACE (CS)). *A concept hierarchy, $\mathcal{H}(N, E)$, with m concept-nodes has a corresponding concept-space (CS) with m concept-dimensions.*

DEFINITION 3.2 (CONCEPT-VECTOR (CV)). *Given a hierarchy, $\mathcal{H}(N, E)$, and the corresponding m-dimensional concept-space, CS, each concept-node in $n_i \in N$ maps to a concept-vector, $V_{n_i} = [w_{n_i,1}, w_{n_i,2}, ..., w_{n_i,m}]$. Here, $w_{n_i,k}$ denotes the weight of the k-th* concept-dimension *of V_{n_i}.*

EXAMPLE 3.1. *Figure 3(a) presents a concept hierarchy. Since the number of nodes in the hierarchy is 4, the corresponding* concept-space *has 4 dimensions (Figure 3(b)).*

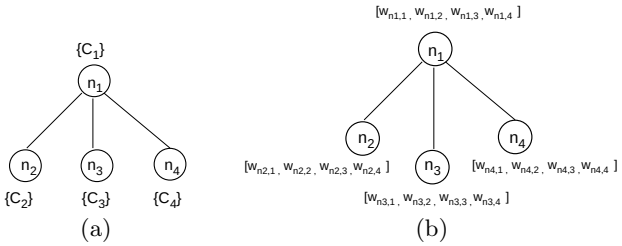


Figure 3: (a) A concept hierarchy where each node represents a concept; (b) nodes of the graph are annotated with *concept vector*s ($w_{n_j,i}$ denotes the contribution of the concept associated with node n_i to the concept associated with node n_j)

3.3 Challenge: Identifying the Concept-Vectors for the Nodes in the Concept Hierarchy

Concept-vectors provide a mechanism through which similarity between concepts can be measured. Given two concept-nodes, n_i and n_j, and their concept-vectors, V_{n_i} and V_{n_j}, we can compute the cosine similarity [39] between these two concept-nodes as

$$sim_{cos}(n_i, n_j) = cosine(V_{n_i}, V_{n_j}) = \frac{\sum_k w_{n_i,k} \times w_{n_j,k}}{\sqrt{\sum_k {w_{n_i,k}}^2 \times \sum_k {w_{n_j,k}}^2}}.$$

However, Definition 3.2 does not state *how to* compute the concept-vectors. It is clear that simply setting the *concept-vector* for node n_i as $V_{n_i} = [0, 0, \ldots, 1, \ldots, 0]$, which includes a non-zero value only for concept c_i, would not be helpful. If this scheme was used, similarity between any two concept-nodes, n_i and n_j where $i \neq j$, would be computed as zero.

In this paper, we propose a concept propagation (CP) method for propagating the concepts (i.e., weights of concept-dimensions), between neighboring nodes. As discussed in Section 1.3, CP process is governed by the constraint that the concept-vectors obtained through propagation should have the same (generality) semantics inherent in the concept hierarchy. Therefore, we need mechanisms to quantify the generality value between two concept-nodes in the hierarchy (Section 4) as well as two concept-vectors in the concept-space (Section 5). This observation is used in Section 6 for propagation.

4. COMPUTING THE RELATIVE GENERALITY OF NODES IN A CONCEPT HIERARCHY

One way to understand the semantic relationship between nodes in a concept hierarchy is to study the *generality* directions and degrees between two adjacent nodes. Since in IS-A hierarchies ancestors are more general than their descendants, the *generality* direction (from child to parent) is clear. However, the degree of generality between a given pair of nodes is not self evident. Thus, in this section, we develop a method for quantifying the generality degree between concept-nodes in a given hierarchy.

4.1 Depth, Density, and Generality

As observed in the literature, the degree of generality between neighboring concepts in a concept hierarchy is related to the local density of the hierarchy as well as the depth of the nodes [27]. Prior work (such as [27]) used this observation for associating weights to edges for structure-based computation of concept similarities.

We note that although *generality* degree between two concepts is related to their *similarities*, these two are not equivalent[1]. Information-based methods overcome this by supplementing the available structural information with *information content* extracted from an available corpus.

4.2 Splitting and Sharing the Concept Range

We can informally state the two basic properties of concept hierarchies, that we leverage for measuring the degree of generality between concept-nodes, as follows:

- A more general concept-node in the hierarchy *subsumes* its children concepts.
- The *concepts subsumed* by sibling concept-nodes are usually non-overlapping. In other words, the relationship between two siblings is captured only through their ancestor nodes.

These properties are used in the literature (for instance in the information-based approaches, such as [28]) to directly compute concept similarities. Unlike previous works (such as [27, 28]), where density and depth are used for measuring similarities between concepts directly, we aim to use these only for measuring the degree of *generality* of one node relative to the other. Thus, CP/CV does not fall into the same pitfall of directly substituting *generality* for *dissimilarity* as the earlier structure-based schemes do.

We represent the semantic coverage of a concept hierarchy as a range, $[0, 1]$. Since the root of the hierarchy subsumes all the other nodes, this range corresponds to the root's share. Naturally, (a) since a parent node subsumes its children entirely, the concept range of the parent covers the concept ranges of its children, and (b) as one moves down in the concept hierarchy, the concept-nodes get more specialized and they subsume less concepts. Thus, the sizes of the concept ranges corresponding to the nodes decrease monotonically as one moves deeper in the hierarchy. Using these two observations, the size of the concept range, $share_{n_i}$, of a node n_i can be recursively defined as follows:

$$share_{n_i} = \begin{cases} 1.0 & if\ n_i\ is\ the\ root \\ share_{n_p} \times \dfrac{1}{num\_children(n_p)}, & otherwise \end{cases}$$

Here n_p is the parent of n_i in the hierarchy and the number children of n_p ($num\_children(n_p)$) gives the local density of the hierarchy relative to node n_i.

Figure 4 provides a sample concept hierarchy and shows how the concepts in the hierarchy share the corresponding concept range. The root node of the hierarchy is the most general concept and occupies the entire range, while the deeper nodes are more specific and have smaller shares. Note that, in the absence of any prior or external/corpus-based knowledge, the children of a given concept-node are assumed to split the concept range of the parent uniformly [2].

[1]In the experiments section, we show that edge weights that represent the degree of generality does not significantly improve similarity computation.

[2]In our future work, we will investigate whether incorporating corpus-based knowledge for more informed splits of the concept-range improves CP/CV results or not.

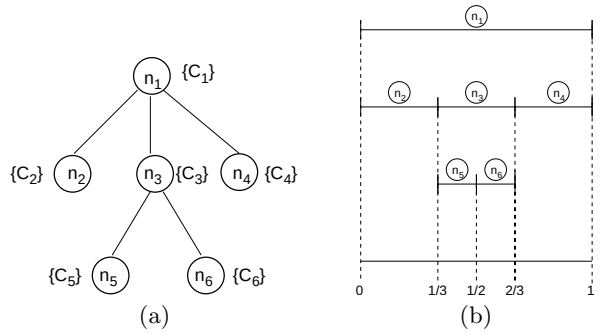


Figure 4: (a) A sample concept hierarchy and (b) the corresponding split of the unit concept range (we assume uniform splits in the absence of any prior or external/corpus-based knowledge)

4.3 Degree of Generality by Structure

Given the above definition of *shares* of concept-nodes, we are ready to define the degree of *generality* between two nodes in a concept hierarchy.

DEFINITION 4.1 (DEGREE OF GEN. (BY STRUCTURE)). *Given two concept-nodes, n_i and n_j (where n_i is an ancestor of n_j) and the corresponding shares, $share_{n_i}$ and $share_{n_j}$, of the concept range, we can define the degree of* generality *of n_i relative to n_j based on the structure information inherent in the hierarchy as*

$$G^{str}_{n_i,n_j} = \frac{share_{n_i}}{share_{n_j}}.$$

Since n_i is an ancestor of n_j, the degree of *generality* of n_i relative to n_j, $G^{str}_{n_i,n_j}$, is greater than 1.0.

EXAMPLE 4.1. *Let us consider the example hierarchy and the corresponding concept range split given in Figure 4. In this example, by Definition 4.1, we can compute $G^{str}_{n_1,n_3}$ as $\frac{1.0}{1/3} = 3.0$ and $G^{str}_{n_1,n_6}$ as $\frac{1.0}{(1/3)\times(1/2)} = 6.0$. In other words, we can state that the degree of* generality *of n_1 relative to n_6 is twice as large as that of n_1 relative to n_3.*

As stated before, unlike prior work, we do not use the *relative degree of generality* as a substitute for measuring *similarity*. Instead, CP/CV uses the correspondence between degrees of generality computed by structure (Definition 4.1) and the degrees of generality computed in the corresponding concept-space (next section) to map concept-nodes into concept-vectors, which then enable similarity computations.

5. COMPUTING THE RELATIVE GENERALITY BETWEEN CONCEPT-VECTORS

Let us consider two concept-nodes, n_i and n_j, in a given hierarchy. Let V_{n_i} and V_{n_j} be the corresponding concept-vectors. Naturally, ensuring that these concept-vectors preserve the degrees of generality (computed as in the previous section) between n_i and n_j requires a similar way of quantifying the degree of generality between V_{n_i} and V_{n_j} in the concept-space. In this section, we propose such a mechanism for computing the degree of generality between two concept-vectors.

One way to think of the degree of generality of one node relative to the other is in terms of constraints imposed on them by their concept-vectors: intuitively, the statement that "*node n_i is more general than node n_j*" can be interpreted as n_i being less constrained than n_j by its concept-vector.

EXAMPLE 5.1. *Let us consider two nodes, n_a and n_b, where n_a is an ancestor of n_b. Let us assume that $\mathcal{D}_{n_a}$ has three non-zero concept-dimensions (corresponding to concepts c_1, c_2, and c_3), while $\mathcal{D}_{n_b}$ has two non-zero concept-dimensions (corresponding to c_2 and c_3):*

- *Since the ancestor, n_a, is more general than the descendant, n_b, the extra concept, c_1, must render n_a less constrained. In a sense, if n_b is interpreted as $c_2 \vee c_3$, then n_a should be interpreted as $c_1 \vee c_2 \vee c_3$ (less constraining than $c_2 \vee c_3$) .*

In order to be able to benefit from this observation in measuring degrees of generality, we need to be able to quantify how well a given vector can be interpreted as the disjunction of the corresponding concepts.

Extended boolean model [31] of vector spaces associate well defined disjunctive and conjunctive semantics to document in order to be able to answer boolean queries on vector data. We will use a similar treatment of the vector space to quantify how well a given vector represents the disjunction of the corresponding concepts. Let $\mathcal{O} = [0, \ldots, 0]$ denote the origin of the concept-space. $\mathcal{O}$, corresponds to a (hypothetical) concept-vector where all concept-dimensions are zero valued. In other words, $\mathcal{O}$ can be interpreted as $(\neg c_1 \wedge \neg c_2 \wedge \neg \ldots \wedge \neg c_m)$ or equivalently as $\neg(c_1 \vee c_2 \vee \ldots \vee c_m)$. Since $\mathcal{O}$ corresponds to $\neg(c_1 \vee c_2 \vee \ldots \vee c_m)$, how much a vector V represents a disjunct can be measured by $|V - \mathcal{O}|$; i.e., the length, $|V|$, of the vector V.

Thus, we can generalize the observation in Example 5.1 as follows: Let us be given two concept-vectors, V_{n_i} and V_{n_j}. Let the concept-dimensions of V_{n_i} and V_{n_j} be denoted as $\mathcal{D}_{n_i}$ and $\mathcal{D}_{n_j}$, respectively. These two vectors will have some dimensions that are *non-zero* in both V_{n_i} and V_{n_j}. These dimensions are denoted as the common dimensions, $\mathcal{D}_{common(n_i,n_j)}$, of V_{n_i} and V_{n_j}. If the non-zero dimensions in $\mathcal{D}_{n_i}$ subsume the non-zero dimensions in $\mathcal{D}_{n_j}$ and if n_i is more general than n_j, then the vector V_{n_i} should be longer than the vector V_{n_j}. In other words, we can define the degree of generality of node n_i relative to n_j based on the relative lengths of the corresponding vectors.

DEFINITION 5.1 (DEGREE OF GEN. IN CONCEPT-SPACE). *Given two nodes, n_i and n_j, and their corresponding concept-vectors, V_{n_i} and V_{n_j}, if $\mathcal{D}_{n_j} = \mathcal{D}_{common(n_i,n_j)}$, then we can define the degree of* generality *(in the concept-space) of n_i relative to n_j as*

$$G^{cs}_{n_i,n_j} = \frac{|V_{n_i}|}{|V_{n_j}|}.$$

6. CONCEPT PROPAGATION

The purpose of propagation is to identify the concept-vectors that represent the concept nodes. The process repeatedly enriches the *concept-vector*s of the nodes by enabling neighboring nodes to exchange concept weights. Before the propagation process starts, concept-vectors of the nodes are simply initialized with the concepts corresponding to each node; i.e., if the node n_i in the concept hierarchy

corresponds to the concept c_i, then the initial concept-vector of this node is $V_{n_i} = [0, 0, \ldots, 1, \ldots, 0]$, where the only non-zero weight is associated with the concept-dimension, c_i.

6.1 Preserving the Semantic Structure inherent in the Class Hierarchy

Naturally, propagation of weights of concept-dimensions moves the concept-vectors in the concept-space. Yet, after the propagation, the semantic properties, such as the degrees of *generality* of the nodes, should be preserved. Therefore, the propagation process should be done in such a way that for any pairs of parent/child nodes, the resulting degree of the *generality* (i.e., G^{cs}) based on the vector-space model should be same with the degree of generality (i.e., G^{str}) computed based on structure. In other words, for all neighboring (parent/child) n_i and n_j in the node hierarchy,

$$G^{cs}_{n_i,n_j} = G^{str}_{n_i,n_j},$$

must hold after propagation. Since, initially, each concept-vector has only one single non-zero concept-dimension, while when propagation process ends, the concept-vectors have non-zero weights for more than one dimension, we refer to this process as *concept propagation* (CP).

6.2 Concept Propagation between a Parent Node and its Children

Using the *preservation of generality* principle stated above, we first develop an algorithm to propagate concepts (or concepts weights) between parent and children nodes in the concept hierarchy. We will extend this to the propagation of concepts in a whole hierarchy in Subsection 6.3.

6.2.1 Computing Propagation Degrees

The propagation algorithm is governed by a per-neighbor *propagation degree* which governs how much concept weights two adjacent nodes in a hierarchy should exchange.

DEFINITION 6.1 (PROPAGATION DEGREE, α). *Let us consider a parent node, par, its children, $chld_i \in Chld(par)$, and the corresponding concept-vectors, $V_{par} = [w_{par,1}, w_{par,2}, \ldots, w_{par,m}]$ and $V_{chld_i} = [w_{chld_i,1}, w_{chld_i,2}, ..., w_{chld_i,m}]$.*

The propagation degrees *($\alpha_{par \to chld_i}$ and $\alpha_{chld_i \to par}$) between the parent node and its children are such that after propagation, we obtain* concept-vectors*, $V'_{par} = [w'_{par,1}, w'_{par,2}, \ldots, w'_{par,m}]$ and $V'_{chld_i} = [w'_{chld_i,1}, w'_{chld_i,2}, ..., w'_{chld_i,m}]$, where*

$$w'_{par,k} = w_{par,k} + \left(\sum_{chld_i \in Chld(par)} \alpha_{child_i \to par} \times w_{chld_i,k} \right),$$

and for all $chld_i \in Chld(par)$ we have

$$w'_{chld_i,k} = w_{chld_i,k} + \alpha_{par \to chld_i} \times w_{par,k}.$$

Since, per Section 4, the parent concept subsumes all child concepts, we can set the *propagation degrees* from children to parent (i.e., $\alpha_{child_i \to par}$) to 1.0. In order to compute the *propagation degrees* from parent node to its children (i.e., $\alpha_{par \to chld_i}$), on the other hand, we rely on the *preservation of generality* principle: after propagation, the degree of *generality* computed in the concept space (Section 5) should be the same with the degree of generality computed based on structure (Section 4). That is, for each $chld_i \in Chld(par)$, we need to ensure that

$$G^{cs\prime}_{par,chld_i} = G^{str}_{par,chld_i},$$

where $G^{cs\prime}_{par,chld_i}$ denote the degree of generality between the concept-vectors of the parent and child vectors after the propagation. Since during the propagation the parent inherits all concepts from its children, after the propagation, we also have

$$\mathcal{D}_{chld_i} = \mathcal{D}_{common(par,chld_i)}.$$

Therefore, using Definition 5.1, we can restate the *preservation of generality* principle as follows:

$$G^{cs\prime}_{par,chld_i} = G^{str}_{par,chld_i} = \frac{|V'_{par}|}{|V'_{chld_i}|}.$$

By using this equation and the value of $G^{str}_{par,chld_i}$, computed in Section 4, we can solve for the propagation degree ($\alpha_{par \to chld_i}$) for each $chld_i \in Chld(par)$.

6.2.2 Iterative Concept Propagation Process

As shown in Figure 5, concept propagation is an iterative process, repeated until all concepts have chance to get propagated across all nodes:

1. The three nodes in Figure 5(a) each correspond to a different concept.
2. Each node is initially annotated with a concept-vector with single non-zero weight (Figure 5 (b)).
3. After the first concept propagation step, the concept-vector corresponding to the parent node had the opportunity to be enriched with all possible concepts from its children. On the other hand, the two siblings did not have an opportunity to receive each others' concepts yet (Figure 5 (c)). Thus, the propagation process should be repeated.
4. Since concept-vectors have been shifted in space in the previous propagation step, new *propagation degree*s

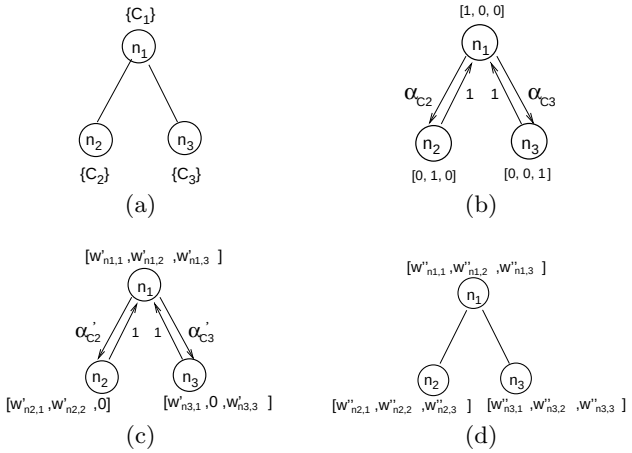


Figure 5: (a) A sample hierarchy, (b) initial concept-vectors and propagation degrees, (c) concept-vectors and propagation degrees after the first iteration and (d) status after the second iteration

must be computed during the second iteration. During this second iteration, all nodes had the opportunity to receive all relevant concepts in the structure (Figure 5 (d)). Therefore, we can stop *concept propagation.*

6.3 Propagation in the Concept Hierarchy

By iteratively propagating concepts between a parent node and its children as described in the previous section, we enable the nodes' concept-vectors to get enriched based on their semantic relationships relative to each other. However, in a large concept hierarchy, considering only parent and its children nodes is not sufficient. The propagation process should not be limited to nodes between a parent and children, but should be performed iteratively on the entire hierarchy.

For representational and computational convenience, in this section we represent the various parameters involved in concept propagation as matrices.

6.3.1 Propagation Adjacency Matrix

Let us be given a concept hierarchy $\mathcal{H}(N, E)$, where nodes in N denote individual concepts and E represents the set of edges between pairs of nodes in N.

DEFINITION 6.2 (PROPAGATION ADJACENCY MATRIX). *Given a concept hierarchy graph $\mathcal{H}(N, E)$, its corresponding propagation adjacency metric,* $\mathtt{M}$*, is defined as follows:*

- *if there is an edge $e_{ij} \in E$ (i.e., n_i is the parent of n_j), then*
  - *entry* $\mathtt{M[i, j]}$ *is equal to $\alpha_{n_i \to n_j}$ (i.e., the parent to child propagation degree defined in Definition 6.1)*
  - *entry* $\mathtt{M[j, i]}$ *is equal to $\alpha_{n_j \to n_i} = 1.0$ (i.e., the child to parent propagation degree defined in Definition 6.1),*
- *otherwise,* $\mathtt{M[i, j]}$ *is equal to 0.*

Note that the diagonal values of $\mathtt{M}$ are all equal to 0.

6.3.2 Concept-Vector Matrix

For convenience, we also represent all concept-vectors corresponding to the nodes in the hierarchy in the form of a single matrix:

DEFINITION 6.3 (CONCEPT-VECTOR MATRIX). *Given a concept hierarchy graph, $\mathcal{H}(N, E)$, the corresponding* concept-vector matrix, $\mathtt{CV}$*, is a matrix, where k-th column of* $\mathtt{CV}$ *corresponds to the* concept-vector *of node $n_k \in N$.*

Since there are m concept-nodes and since each concept-vector has m dimensions, the size of $\mathtt{CV}$ is $m \times m$.

6.3.3 Concept Propagation Process

Given a propagation adjacency matrix, $\mathtt{M}$, and a concept-vector matrix, $\mathtt{CV}$, we execute propagation using a concept propagation operator:

DEFINITION 6.4 (CONCEPT PROPAGATION OPERATOR). *The concept propagation operator, $\oplus$, is such that, given a concept-vector matrix* $\mathtt{CV}$ *and a propagation adjacency matrix* $\mathtt{M}$*,*

$$\mathtt{P} = \mathtt{CV} \oplus \mathtt{M}$$

is a matrix, where $\mathtt{P}[k, i]$ *is the cumulative propagation weight for concept-dimension, c_k, in the concept-vector, V_{n_i}.*

After the an iteration with $\oplus$ operator, the computed cumulative propagation weights in $\mathtt{P} = \mathtt{CV} \oplus \mathtt{M}$ should be added to the original values in $\mathtt{CV}$. Based on Definition 6.1, the entry $\mathtt{P}[k, i]$ can be computed as follows:

$$\mathtt{P}[k, i] = \sum_{1 \le h \le m} \mathtt{CV}_{k,h} \times \mathtt{M}_{i,h}.$$

In other words, the concept propagation operator is

$$\mathtt{P} = \mathtt{CV} \oplus \mathtt{M} = \mathtt{CV.M},$$

where $\mathtt{CV.M}$ is the matrix product of $\mathtt{CV}$ and $\mathtt{M}$. Thus, the new *enriched* concept-node matrix after the first iteration of concept propagation is equal to

$$\mathtt{CV}_1 = \mathtt{CV} + \mathtt{P} = \mathtt{CV} + (\mathtt{CV} \oplus \mathtt{M}) = \mathtt{CV} + \mathtt{CV.M} = \mathtt{CV.(I + M)}$$

where $\mathtt{I}$ is the identity matrix. Note that since all diagonal values in $\mathtt{M}$ are zero, we will use $\mathtt{M}_\mathtt{I}$, where all diagonal values are 1 and all non-diagonal entries are those in $\mathtt{M}$, to denote $\mathtt{I + M}$.

We can generalize this process as follows: Let d be the diameter (the greatest number of edges between any nodes) in the graph, $\mathcal{S}(N, E)$. Then, the final concept-vector matrix can be computed by repeated application of the $\oplus$ (i.e., matrix multiplication) as follows:

$$\mathtt{CV}_{final} = \mathtt{CV.M}_{\mathtt{I}1}\mathtt{.M}_{\mathtt{I}2}\mathtt{.M}_{\mathtt{I}2} \ldots \mathtt{M}_{\mathtt{I}d},$$

where $\mathtt{M}_{\mathtt{I}m}$ is the propagation adjacency matrix computed for the m^{th} iteration.

6.3.4 Stopping Condition

Since, after the d^{th} iteration, all nodes are enriched with all concepts (with appropriate weights), the construction of the concept-vectors is complete and the process stops.

7. EXPERIMENTS

In this section, we describe the experiments we carried out to evaluate the effectiveness of the concept propagation method we introduced in this paper. First we describe the experimental setup and then we will discuss the results.

7.1 Experimental Setup

Our aim in this paper is to develop a technique that can be used to mine concept (keyword) similarities, without the help of external information (such as a representative corpus). Naturally, the most reasonable way to evaluate the efficiency of CP/CV technique proposed in this paper is to discover the correlation of the resulting concept similarity judgments with human common sense. Thus, to evaluate the performance of our approach effectively, we need (a) a concept hierarchy, (b) ground truth (i.e., a user study of similarities) on this hierarchy, and (c) representative implementations of alternative (i.e., information-based and structure-based) approaches. Fortunately, such a concept hierarchy (WordNet), appropriate user studies [24, 28] on this concept hierarchy, and already reported similarity results [26, 28] for WordNet do exist:

Concept Hierarchy. As discussed in Section 2, WordNet [22, 23] is a lexical reference system for English nouns, verbs, adjective, and adverbs. Here, we use the WordNet version 2.0, composed of a total 115424 synsets that represent concepts. Among these synsets, we focus on the 79689 noun synsets and the IS-A relationship between them. Note

Table 1: Sample similarity values from various user studies and alternative methods of computation (the similarity values returned by different approaches are not normalized)

Word pair	User Studies		Prior Results			CP/CV	
	MC [24] (user study)	RR [28] (user study)	$Stru_{cnt}$ [26]	$Stru_{wght}$ [27] (our impl.)	Info [28]	CP/CV partial hierarchy	CP/CV complete hierarchy
rooster-voyage	0.08	0	0	0.1965	0	0	0
noon-string	0.08	0	0	0.1462	0	0	0.00002
glass-magician	0.11	0.1	22	0.6242	1.0105	0.00002	0.00122
cord-smile	0.13	0.1	20	0.3972	2.3544	0	0.00001
coast-forest	0.42	0.6	0	0.9549	0	0.03886	0.00185
lad-wizard	0.42	0.7	26	1.9996	2.9683	0.00086	0.00281
monk-slave	0.55	0.7	27	1.9996	2.9683	0.02219	0.02896
forest-graveyard	0.84	0.6	0	0.9611	0	0.00220	0.00001
coast-hill	0.87	0.7	26	1.9998	6.2344	0.17288	0.05697
food-rooster	0.89	1.1	18	0.6483	1.0105	0	0
monk-oracle	1.10	0.8	24	0.8933	2.9683	0	0.05904
car-journey	1.16	0.7	0	0.8972	0	0	0
brother-lad	1.66	1.2	26	1.9995	2.9355	0.01210	0.20490
crane-implement	1.68	0.3	24	1.9999	2.9683	0.04410	0.11036
brother-monk	2.82	2.4	24	1.9996	2.9683	0.00021	0.00255
implement-tool	2.95	3.4	29	1.9999	6.0787	0.47448	0.63054
bird-crane	2.97	2.1	27	1.9999	9.3139	0.17571	0.24814
bird-cock	3.05	2.2	29	2	9.3139	0.93913	0.79107
food-fruit	3.08	2.1	27	0.8983	5.0076	0.00720	0.00001
furnace-stove	3.11	2.6	23	1.9995	1.7135	0.00006	0.00018
midday-noon	3.42	3.6	30	2	12.393	1	1
magician-wizard	3.50	3.5	30	2	13.666	1	1
asylum-madhouse	3.61	3.6	29	1.9999	15.666	0.99612	0.99590
coast-shore	3.70	3.5	29	1.9998	10.808	0.97584	0.98547
boy-lad	3.76	3.5	29	1.9999	8.424	0.99002	0.99608
journey-voyage	3.84	3.5	29	1.9999	6.7537	0.95546	0.71694
gem-jewel	3.84	3.5	30	2	14.929	1	1
automobile-car	3.92	3.9	30	2	8.0411	1	1

that **WordNet is not a domain specific ontology.** However, using WordNet, we are able to concretely and fairly compare the performance of the proposed algorithm against existing structure- and information-based approaches as described next.

Ground Truth. Commonly used ground truth data to evaluate methods for computing the semantic similarity between words comes from an experiment carried by Miller and Charles [24]. The authors did a user study where assessors were given 30 pairs of words and asked to rate these words for similarity in meaning on a scale from 0 (dissimilar) to 4 (highly similar). In 1999, Resnik [28] replicated the experiment by Miler and Charles. The results obtained from this second study were highly correlated (0.9015) with the Miller and Charles study. The high correlation provides support for the validity of both user tests, yet the imperfect agreement between the studies indicates that 0.9 (i.e., the agreement between two user studies on the same data) is essentially an upper bound for meaningful correlation degrees.

The first three columns in Table 1 contain the word-pairs used in these two studies and the corresponding similarity values (MC for Miler and Charles' study and RR for the Resnik's replication of the experiment)[3]. In this section, we use both Miler and Charles (MC) and Resnik (RR) data sets as ground truth.

Information- and Structure-based Similarity Results for our Comparative Study. A major advantage of using MC and RR ground truth data is that there are already various published works that evaluate their algorithms using this data set. Therefore, we can directly compare our results with these published results.

Table 1 also presents similarity results for the same word pairs using representatives of different approaches. The column titled *Info* contains the semantic similarity values measured by the information-based method reported in [28]. The column titled $Stru_{cnt}$ lists the semantic similarity values computed using a structure-based method (edge counting [26]).

We have also considered an alternative structure-based method, where edges are weighted based on the depth and local density [27]. Since [27] does not contain enough experiment data and enough details regarding how exactly the edge weights are set, we implemented this method ourselves as follows: Each word in WordNet has possibly several senses. For example the word, "car" has five different senses and each sense of this word has a corresponding concept-node. Let us consider two words, wd_i and wd_j and the corresponding sets of concept-nodes, $con(wd_i)$ and $con(wd_j)$. We computed the similarity between wd_i and wd_j using the weighted edge sum method, $Stru_{wght}$, as also done in [28]:

$$2maxWeight - \min_{n_i \in con(wd_i),\ n_j \in con(wd_j)} [sum_{wght}(n_i, n_j)]$$

where,

- given edge, e_{ij}, between n_i and n_j (where n_i is a parent of n_j) the edge weight is computed using the concept range splitting approach that captures both the depth of the concept-nodes and their local densities,
- $maxWeight$ is the maximum possible sum of the edge weight between root and leaf nodes in WordNet, and

[3]Among the 30 pairs of words used by Miler and Charles, two pairs are missing in WordNet; thus, we use only 28 pairs.

- $sum_{wght}(n_i, n_j)$ is the sum of the edge weights along the shortest path between n_i and n_j. Note that, while information-based approaches take the maximum value when there are multiple subsumers, structure-based approaches rely on the shortest path to compute the semantic similarity between two words.

The column titled $Stru_{wgth}$ lists the semantic similarity values computed using (our implementation of) the structure-based method with weighted edges [27].

Note that there are other approaches which leverage other available information for improved similarity evaluation. For example, [18] proposes a spreading activation based approach which assumes the availability of initial edge weights that are extracted using the frequency information obtained from a corpus. [19] leverages not only IS-A relationships, but also PARTS-OF relationships between concepts. [19] also assumes the existence of training data (pairs of words from [30]) to learn and tune various parameters. Since CP/CV does not assume any such extra information, we do not compare it against [18, 19]. Incorporation of such extra knowledge in CP/CV is part of our future work.

Implementation of CP/CV on WordNet. Given two words, wd_i and wd_j, and the corresponding sets of concept nodes, $con(wd_i)$ and $con(wd_j)$, we computed the CP/CV similarity between wd_i and wd_j as

$$\max_{n_i \in con(wd_i),\ n_j \in con(wd_j)} [cosine(V_{n_i}, V_{n_j})],$$

where V_{n_i} and V_{n_j} are the concept-vectors (CVs) obtained through concept propagation (CP).

We carried similarity value computations on two graphs: *Complete* WordNet (79,689 noun concept nodes) and *partial* WordNet, consisting of the minimum subtrees that contain all the words in the user study (10,121 concept nodes). Table 1 also contains the semantic similarity values returned by CP/CV for both partial and complete hierarchies.

Evaluation Metric (Correlation Coefficient). As in [28], to evaluate concept propagation methods, we use the correlation with the ground truth as the performance metric. Given m-pairs of observations $(x_1, y_1), (x_2, y_2).....(x_m, y_m)$, the correlation coefficient, r, is defined as

$$r = \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{m}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{m}(y_i - \bar{y})^2}},$$

and measures how closely (linearly) related x observations are with the y observations [10]. The correlation value of 1.0 means that there is a perfect linear relationship between the two data sets, while the value of 0.0 indicates that two data sets are not linearly related.

7.2 Results and Discussion

With the above experiment setup, we compared correlation coefficients obtained though CP/CV with correlation coefficients returned by various alternative approaches. Table 2 lists the correlation coefficients of various approaches against; i.e., Miler and Charles's, MC, and Resnik replication, RR, user studies. To make sure that the results we report are statistically significant, we verified these correlation values using Fisher's Z-test [10]. According to this test, our statistical confidence for all the coefficient values reported in Table 2 is higher than 95%.

Table 2: Correlation coefficients from various approaches against the ground truth from the user studies, MC and RR

Similarity Method	Correlation against MC	Correlation against RR
Structure-based ($Stru_{cnt}$)	0.6644	0.6533
Structure-based ($Stru_{wght}$)	0.6979	0.6754
CP/CV on partial hierarchy	**0.8006**	**0.8544**
CP/CV on complete hierarchy	**0.8138**	**0.8693**
Information-based (Info)	0.7978	0.8144

The main observation from this table is that CP/CV provides better results than all alternatives for both partial and complete hierarchies and against ground truth from both user studies.

CP/CV provides a correlation coefficient value of 0.8138 against Miler and Charles's Rating (MC); this is a significant improvement against both structure-based methods, edge counting and sum of edge weights (21.4% and 16.6% respectively). The correlation coefficient value obtained against the Resnik user study is even higher: 0.8639 on the complete hierarchy. Note that considering that the correlation agreement between two user studies is about 0.9, this correlation between CP/CV and RR is quite high. This also translates to larger improvements against structured-based methods (33.1% and 28.7% improvement on edge counting and sum of the edge weight methods, respectively).

Note that, for both user studies, performance of CP/CV seems to be slightly better than even the performance of the information-based methods. As expected (since information-based methods can leverage an external corpus, while CP/CV only uses the hierarchy itself), the difference in improvement is less pronounced (only upto 6%). Since our goal, in this paper, is to address the needs of the cases (such as relational databases with unique keys) where sufficient frequency information to support information-based approaches is not readily available, we do not further investigate the impact of this 6% gain. However, the fact that CP/CV matches the performance of information-based approaches without having to rely on frequency information provides a strong validation for the proposed approach.

Comparison of the CP/CV values in Table 2 computed on the partial and complete WordNet hierarchies highlight that there is some value in propagating concepts even though they are not in the immediate neighborhood (partial trees) of the words of interest. Figure 6 shows this more explicitly. Here, the curve represents the correlation coefficient values between RR user study and CP/CV, obtained at different iterations of the CP algorithm (Section 6). The first value on the curve corresponds to 0^{th} iteration (no propagation), the next value to 1^{st} iteration (i.e., propagation between immediate neighbors), and so on. Since it takes at least k iterations of CP for concept-nodes at distance k from each other to exchange weights, this plot also shows the effect of the distance (of concept-nodes) in contributing to the concept-vectors of each-other. As shown in Figure 6, CP/CV observes significant improvements during the first 3 iterations. Beyond these, the correlation coefficient values becomes more or less stable and the maximum correlation value is observed after the 10^{th} iteration. After this point, the obtained correlation coefficient values start to decrease slightly with additional iterations (though a new stable point is reached around $18th$ iteration which is also the maximum

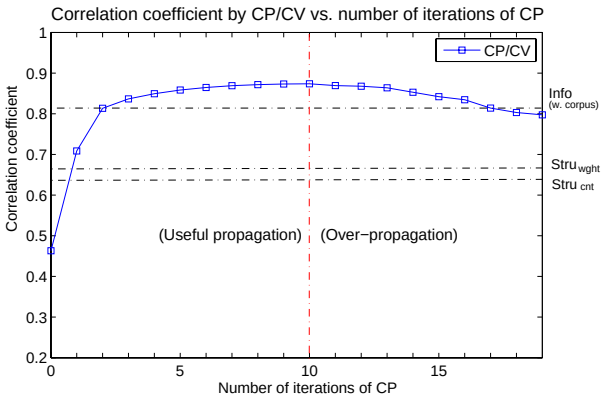


Figure 6: Correlation coefficients (between RR user study and CP/CV on the complete hierarchy): there is a significant improvement in the first few iterations and the best result is obtained around the 10th iteration; after 10th iteration context-vectors get over-propagated

depth of the WordNet). We refer to the slightly drop after the 10^{th} iteration as over-propagation and conjecture that this is because of the average depth WordNet has been passed. Note that, in general, it should be possible to identify (without having to use ground truth data) the point at which the performance starts dropping simply by checking if the inter-iteration correlations (i.e., correlations between the similarity values returned by consecutive iterations of CP) starts dropping. Our future work will also involve investigation of the over-propagation issue.

8. CONCLUSIONS AND FUTURE WORK

In this paper, we first proposed a vector representation for the concept-nodes in an IS-A hierarchy. We then presented two methods for quantifying the generality relationships between concept-nodes: one measures generality directly in the hierarchy, while the other uses concept-vectors. We conjectured that the degree of generality between two nodes should be the same whether it is measured in the hierarchy or in the concept-space. Using this observation, we developed a concept propagation algorithm (CP) to map concept-nodes in an hierarchy into a concept-space so that the similarities between nodes can be measured using the resulting concept-vectors (CV). Experiments showed that the CP/CV algorithm provides a significant improvement in results when compared with structure-based methods and as good or better results than information-based methods (without having to require an available corpus).

9. REFERENCES

[1] Unified Medical Language System (UMLS). http://umlsinfo.nlm.nih.gov/.

[2] E. Agirre and G. Rigau. Word Sense Disambiguation using Conceptual Density. In *COLING*, 1996.

[3] C.F. Baker et al. The Berkeley FrameNet Project. In *COLING*, 1998.

[4] A. Balmin, V. Hristidis, and Y. Papakonstantinou. ObjectRank: Authority-based keyword search in databases. In *SIGMOD*, 2004.

[5] G. Bhalotia, *et al.* Keyword Searching and Browsing in Databases using BANKS. *ICDE*, 2002.

[6] K.S. Candan et al. Discovering mappings in hierarchical data from multiple sources using the inherent structure. KAIS, 2006.

[7] S. Cohen, *et al.* XSEarch: A semantic search engine for XML. In *VLDB*, 2003.

[8] F. Crestani. Application of Spreading Activation Techniques in Information Retrieval. Artificial Intelligence Review, 11(6), 97.

[9] S. Deerwester et al. Indexing by Latent Semantic Analysis. JASIS, 41(6), 1990.

[10] J.L. Devore. Probability and Statistics for Engineering and the Sciences. International Thomson Publishing Company.

[11] F. Gelgi et al. Improving Web Data Annotations with Spreading Activation. In *WISE*, 2005.

[12] M. Gruninger, and M.S. Fox Methodology for the Design and Evaluation of Ontologies. In *Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995

[13] L. Guo, *et al.* XRANK: Ranked keyword search over XML documents. In *SIGMOD*, 2003.

[14] J.J. Jiang, and D.W. Conrath. Semantic Similarity Based on Corpus Statistics and Lexical. In *ROCLING X*, 1997.

[15] V. Kacholia, *et al.* Bidirectional Expansion For Keyword Search on Graph Databases. *VLDB*, 2005.

[16] P.D. Karp. A Qualitative Biochemistry and its Application to the Regulation of the Trytophan Operon. In *Artificial Intelligence and Molecular Biology*, 289-235.

[17] J. Kleinberg. Authoritative sources in a hyperlinked environment. JACM, 46(5), 1999.

[18] H. Kozima and T. Furugori. Similarity between Words Computed by Spreading Activation on an English Dictionary. In *EACL'93*.

[19] Y. Li et al. An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources. TKDE, 15(4), 2003.

[20] D. Lin. An Information-Theoretic Definition of Similarity. In *ICML*, 1998.

[21] A.G. Maguitman et al. Algorithmic detection of semantic similarity. In *WWW*, 2005.

[22] G.A. Miller et al. Introduction to WordNet: An on-line lexical database. International Journal of Lexicography, 3(4), 1990.

[23] G.A. Miller. Nouns in WordNet: a lexical inheritance system. International Journal of Lexicography, 3(4), 1990.

[24] G.A. Miller and W.G. Charles. Contextual correlates of semantic similarity. In Language and Cognitive Processes, 6(1), 1991.

[25] T. Qin et al. A Study of Relevance Propagation for Web Search. In *SIGIR*, 2005.

[26] R. Rada, et al. Development and application of a metric on semantic nets. IEEE Transactions on Systems, Man and Cybernetics, 19(1), 1989.

[27] R. Richardson and A.F. Smeaton. Using WordNet in an Knowledge-Based Approach to Information Retrieval. Working paper CA-1294, Dublin City Univ., Dublin, 1994.

[28] P. Resnik. Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. JAIR, Vol.11, 1999.

[29] M.A. Rodryguez and M.J. Egenhofer, Determining Semantic Similarity among Entity Classes from Different Ontologies. TKDE, 15(2), 2003.

[30] H. Rubenstein and J. Goodenough. Contextual correlates of synonymy. CACM, 8 (10), 1965.

[31] G. Salton et al. Extended Boolean information retrieval. CACM, 26(11). 1983.

[32] J. Savoy et al. Ranking Schemes in Hybrid Boolean Systems: A New Approach. JASIS, 49(3), 1997.

[33] S. Scott and S. Matwin. Text Classification Using WordNet Hypernyms. Workshop on Usage of WordNet in Natural Language Processing Systems, 1998.

[34] A. Shakery and C. Zhai. Relevance Propagation for Topic Distillation UIUC TREC-2003 Web Track Experiments. In *TREC*, 2003.

[35] L.I. Smith. A Tutorial on Principal Components Analysis. Cornell University, USA, 2002.

[36] R. Song et al. Microsoft Research Asia at Web Track and Terabyte Track of TREC 2004. In *TREC*, 2004.

[37] P.E. van der Vet, P.H. Speel, and N.J.I. Mars. The Plinius ontology of ceramic materials. In *Workshop on Comparison of Implemented Ontologies*, 1994

[38] Z. Wu, and M. Palmer. Verb Semantics and Lexical Selection. In *ACL*, 1994.

[39] R.B. Yates and B.R. Neto. Modern Information Retrieval. Addison Wesley, 1999.