Computing and Informatics, Vol. 30, 2011, 881–900

CONCEPT VECTOR FOR SIMILARITY MEASUREMENT BASED ON HIERARCHICAL DOMAIN STRUCTURE

Hong-Zhe Liu, Hong Bao

Beijing Jiaotong University
Beijing Union University
Beijing, China
e-mail: {xxtliuhongzhe, baohong}@buu.edu.cn

De Xu

Beijing Jiaotong University
Beijing, China
e-mail: dxu@bjtu.edu.cn

Communicated by Miloš Cerňak

Abstract. The concept vector model generalizes standard representations of similarity concept in terms of tree-like structure. In the model, each concept node in the hierarchical tree has ancestor and descendent concept nodes composing its relevancy nodes, thus a concept node is represented as a concept vector according to its relevancy nodes' density and the similarity of the two concepts is obtained by computing cosine similarity between their vectors. In addition, the model is adjusted in terms of local density and multiple descendents problem. The model contains structure information inherent and hidden in the tree. We show that this measure compares favorably to other measures, and it is flexible in that it can make comparisons between any two concepts in a hierarchical tree without relying on additional dictionary or corpus information.

Keywords: Concept similarity, concept vector model, cosine similarity measure, hierarchical taxonomy

1 INTRODUCTION

Semantic similarity between concepts is becoming a generic problem for many applications of computational linguistics and artificial intelligence. The notion of similarity is to identify concepts having common "characteristics". Humans can judge relatedness between concepts even if they do not know how to define that relatedness formally. For example, even a small child can tell that "apple" and "orange" have more to do with each other than "apple" and "toothpaste". Formally, the way in which these pairs of concepts are related to each is called an "is-a" hierarchy. However, even dissimilar entities may be semantically related in some way. For example, "apple" and "orange" have some similarity, while "glass" and "water," "tree" and "shade," or "gym" and "weights" have no formal similarity but are still related in some way. Semantic similarity is a type of semantic relatedness. In this paper, we focus on semantic similarity. The similarity measures that make use of hierarchical structure can be grouped into three categories, including edge based: Rada's [7], Wu and Palmer's [11]; information content based: Resnik's [6]; edge and information content based: Leacock and Chodorow's [1], Lin's [2], Jiang and Conrath's [5] distance measure; feature based measure including Banerjee and Pedersen's extended gloss overlap (lesk) [8], Patwardhan context vectors [9]. In this paper, we propose a novel vector based method to compute concept similarity in a hierarchical taxonomy by cosine similarity.

2 NEED FOR A NEW MEASURE

Edge counting based methods that consider the position of the concepts in the hierarchy is better than plain path length methods. The simple edge counting based method is sensitive to the problem of varying link distances [6], which cause nodes in richly structured parts of ontology to be consistently judged less similar to one another than in sketchily structured parts of hierarchy. Overall, all these edge based methods only make use of a few structure information of the hierarchy, so it can not make fine grained distinctions for any two concepts in similarity computing.

Information content based methods need an additional large text corpus to compute word frequency. In addition, they ignore the structure of the taxonomy, so they normally generate a coarse result for comparison of concepts. In particular, they do not differentiate the similarity values of any pairs of concepts in a sub-hierarchy as long as their lowest common subsumer is the same [5].

The feature based methods rely on the WordNet gloss of the compared words, but the glosses can be very short and do not provide sufficient vocabulary [8], so the listed measures expand the glosses of words through hyper/hyponym or other relations in the hierarchy to include glosses of concepts that are known to be related to the concept being compared.

Given an appropriate corpus, the edge and information content based hybrid methods usually have better performance than those of edge based or information content based approaches, but such a large corpus is usually available only 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 domain specific applications, a large text corpus can not be expected to be readily available, and in many applications that rely on a relational database also does not carry the same information as in text collections. In these cases, the similarities between these concepts have to be extracted from the existing concept hierarchy only. Thus, in this paper, we focus on the challenge of learning concept similarities which make use of full structure information inherent and hidden in a given hierarchy.

3 CONTRIBUTIONS OF THIS PAPER

In this paper, we propose a way of mining similarities of concepts without relying on any corpus information: if each concept node in a given hierarchy could be represented as a concept vector, then these vectors could be compared to compute concept similarity. Essentially, such concept vectors would capture the semantic information (necessary for similarity computation), otherwise inherent but hidden within the structure of the hierarchy. Based on this observation, we propose a model which leverages the semantic relationships between concept nodes (implied by the structure of the concept hierarchy) to annotate each concept node with a concept vector. The concept vectors are then used for similarity computations. The main contributions of our work are as follows:

1. A method for identifying a node's relevancy nodes in a concept hierarchical tree is defined.
2. A method for quantifying the density of a concept node relative to another concept node in the hierarchy is defined.
3. A novel concept vector representation of concept nodes in a concept hierarchical tree is proposed. Thus a method for computing the similarity of two concepts through their concept vectors is presented.

4 RELEVANCY NODES BASED CONCEPT VECTOR MODEL

We propose a Relevancy Nodes based Concept Vector Model (RNCVM) in which we map concept nodes in a hierarchy into a concept vector, and then we propose a method for similarity computation based on this model. First we define our concept hierarchical model, which is a presupposition of our method; secondly we elaborate on the origin and challenge of this model; and then we propose the RNCVM based similarity computation method; finally, we test our method with experiments and compare it with related methods.

4.1 Concept Hierarchical Model Definition

Definition 1 (Concept hierarchical model)**.** Denote $H(N, E)$ to be a rooted tree where N is the set of concept nodes (corresponding to the concepts) in the tree and E is the set of edges between the parent/child pairs in H. The semantic coverage of the child concept nodes is the partition of the semantic coverage of their parent concept node.

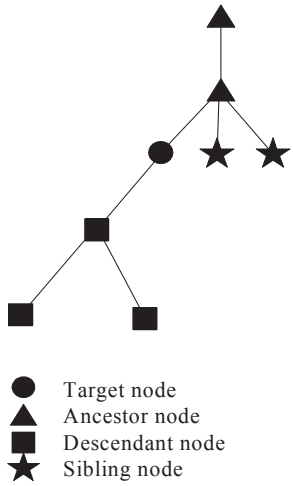

Fig. 1. The concept node types illustration

A concept node is a *parent* of another concept node if it is one step higher in the hierarchy and closer to the root concept node. Each concept node in a tree has zero or more *child concept nodes*, which are one step below their parent concept node in the hierarchy. *Sibling concept nodes* share the same parent concept node. A concept node has at most one parent concept node. Concept nodes that do not have any children are called *leaf concept nodes*. The topmost concept node in the hierarchy is called the *root concept node*. Being the topmost concept node, the root concept node will not have parents, and it is the symbol of the universe. All concept nodes (except root concept node) can be reached from the root concept node by following edges and concept nodes on the path, and all these concept nodes on the path composed of the *ancestor concept nodes* of that concept node. All concept nodes below a particular concept node are called *descendents* of that concept node. Figure 1 above illustrated the concept node types.

The concept hierarchical model is the premise of our method, and our similarity computation is from cosine similarity which is based on the orthogonality of its components, so the semantic coverage of the concept nodes should be independent.

So we limit the semantic coverage of the child concept nodes are the partition (instead of covering) of the semantic coverage of their parent concept node. That is, the concepts subsumed by sibling concept nodes are usually non-overlapping; the relationship between two siblings is captured only through their ancestor concept nodes.

4.2 Ground Truths for Concept Hierarchical Model

From human intuition and literature work, the following judgments can be inferred in the concept hierarchical model.

Density and similarity
With regard to the tree density, it can be observed that the densities in different part of the hierarchy are different. *The greater the density, the closer the distance between the nodes* [5, 10]. For example, the 'plant' section of the knowledge base is very dense, individual node having up to three and four hundreds children, collections of generally unpronounceable plant species; it can argue that the distance between nodes in such a section of structure should be very small relative to other less dense regions. That is in Figure 2, the similarity value of the left part should be less than the similarity value of the right part of the hierarchy.

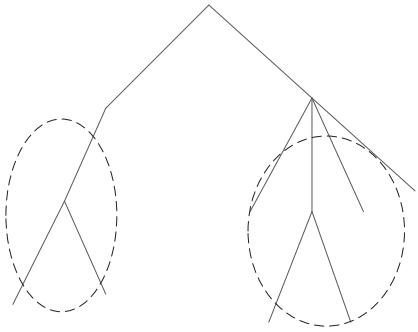

Fig. 2. Local density effect

Depth and similarity
The deeper the depth of the nodes located, the higher the similarity of them. The foundation is that the distance shrinks as one descends the hierarchy, since differentiation is based on finer and finer details [5]. That is in figure 3, the value of sim (C_1, C_2) should be less than the value of sim (C_3, C_4).

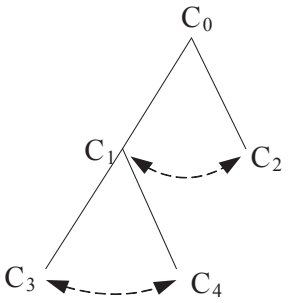

Fig. 3. Depth effect

Path length and similarity

Semantic network includes concepts (usually nouns or noun phrases) that are linked to one another by named relations, for example, hyper/hyponym relation ('is-a' relation) and hol/meronym relation ('part-of' relation). If the semantic network is linked only by taxonomic 'is-a' relation, it is generally called 'is-a' semantic network or 'is-a' taxonomy. In this kind of semantic network, parent concept is more generalized than child concept, while child 'is a kind of' its parent concept.

Rada et al. pointed out that the assessment of similarity in a semantic network can be in fact thought of as involving just taxonomic 'is-a' relation, and the simplest form of determining the distance between two elemental concept nodes, A and B, is the shortest path that links A and B, i.e. the minimum number of edges that separate A and B [7]. However, Jiang and Conrath then pointed out in a more realistic scenario, the distances between any two adjacent nodes are not necessarily equal. It is therefore necessary to consider that the edge connecting the two nodes should be weighted. To determine the edge weight automatically, certain aspects should be considered in the implementation. Most of these are typically related to the structural characteristics of a hierarchical network. Some conceivable features are: local network density (the number of child links that span out from a parent node), depth of a node in the hierarchy, type of link, and finally, perhaps the most important of all, the strength of an edge link [5]. From Rada et al. and Jiang and Conrath, at least we can state that if *the shorter path is contained within the longer path in an 'is-a' taxonomy, the concept nodes pair with shorter path between them has greater concept similarity than that of with longer path between them.* That is in Figure 4, the value of sim (C_0, C_3) should be less than the value of sim (C_0, C_1).

4.3 Concept Vector and Semantic Similarity

Concept vectors provide a mechanism through which similarity between concepts can be measured.

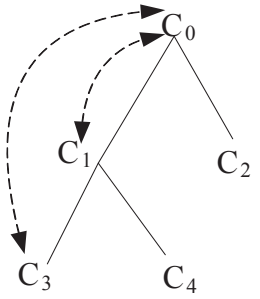

Fig. 4. Path length effect

Definition 2 (Concept vector)**.** Given a concept hierarchy model, $H(N, E)$, with n concept nodes, the concept vector of a concept node C_i in this hierarchy has n dimensions. The concept node C_i's concept vector denoted as $\overrightarrow{C_i} = (v_{i,1}, v_{i,2}, \ldots, v_{i,n})$, $v_{i,1}$, $v_{i,2}$, ..., $v_{i,n}(i = 1, 2, \ldots, n)$ are the dimension values corresponding concepts C_1, C_2, ..., C_n relative to concept C_i.

Given two concept nodes, and their concept vectors, $\overrightarrow{C_i}, \overrightarrow{C_j}$, then their similarity is computed by help of Equation (1):

$$sim(C_i, C_j) = \frac{\overrightarrow{Ci} \bullet \overrightarrow{Cj}}{\left\|\overrightarrow{Ci}\right\| \left\|\overrightarrow{Cj}\right\|}. \tag{1}$$

4.4 Identifying the Concept Vectors for the Concept Nodes in the Hierarchy

As mentioned above, in the traditional corpus based method, the weight of concepts (the frequency of the concept) is derived from a large text corpus. We discuss a given hierarchy without a large corpus for frequency information extraction. Therefore, we need mechanisms to leverage the weights of concept nodes in the hierarchy. Essentially, our concept vectors would capture the semantic information inherent but hidden within the structure of the hierarchy which is the most challenging part of our work.

Local density as a weighting function

Consider that the document-document similarity computation, documents are represented as vectors; in the vector each dimension corresponds to a separate term. If a term occurs in a document, its value in the vector is non-zero. Usually a document is represented as a vector and the frequencies of a cluster of terms appeared in the document are used as dimension values. Vector operations can be used to compare document-document similarity. Here in a concept hierarchy model, the dimension

values of each concept can be obtained only from the hierarchical structure. From observation, the density information of each concept node is inherent and hidden in the hierarchy.

Definition 3 (Local density)**.** The density of a root concept node in a given concept hierarchy model is equal to 1, the density of other concept nodes equals the number of sibling concept nodes of that concept node plus 1 (itself).

Definition 3 defines the situation of the uniform concept node local density. If sibling concept nodes have different density, it can be obtained from a large text corpus using traditional method as in references [5, 6]. For example, in reference [6], frequencies of concepts in the WordNet taxonomy were estimated using noun frequencies from the Brown Corpus of American English which is a large (more than 1 000 000 words) collection of text from news articles to science fictions. Each noun that occurred in the corpus was counted as an occurrence of each taxonomic class containing it. But, as mentioned above, such text corpuses are usually hard to obtain in many domain specific applications (for example, biology and medicine) and in applications that rely on relational databases. Even if the large text corpus is available, these methods are slow due to the huge text statistics work, so we choose to use uniform density value in Definition 3 to substitute their real distribution values. Experiment in Section 5.2 shows that our solution has ideal human correlation values.

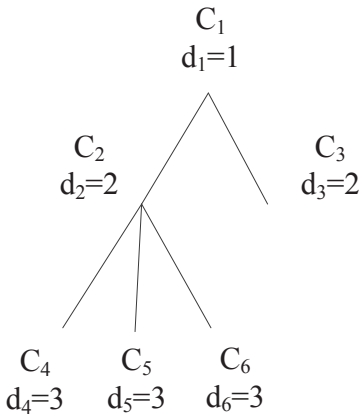

Fig. 5. Tree example to show concept density

Figure 5 provides a sample concept hierarchy. It shows how the concepts in the hierarchy share their local density. The density of root concept node C_1 is 1, the densities of C_2 and C_3 are 2, and densities of C_4, C_5, and C_6 they are 3.

Consider the vector space model's approach origins in document-document similarity. The presumption is that, given a certain number of terms, the frequency

of these terms in a document can be used as vectors to compute query-document similarity. In the situations we have described, the density information of a given node's relevancy nodes were used as vectors to compute internode similarity.

Relevancy nodes based concept vector

In the document-document similarity computation, a cluster of terms appeared in the document are used as dimension values. Given a concept node in the hierarchy, its ancestor concept nodes subsume its attributes, and its descendent concept nodes inherit it; so except the concept node itself, its ancestor and descendent concept nodes are relevancy to that concept node, which we used as "terms" in our structure.

Definition 4 (Relevancy Nodes)**.** Given a concept node in the hierarchy, the concept node itself, its ancestor and descendent concept nodes compose its relevancy nodes.

Consider the vector space model's approach origins in document-document similarity. The presumption is that, given a certain number of terms, the frequency of these terms in a document was used as vectors to compute the query-document similarity. In our situations, the density information of a node's all relevancy nodes was used as vectors to compute internodes similarity.

Definition 5 (Relevancy nodes based concept vectors for HCT)**.** Given an HCT with n concept nodes, the concept vector of C_i is denoted as $\overrightarrow{C_i} = (v_{i,1}, v_{i,2}, \ldots, v_{i,n})$ and $v_{i,j} (i = 1, 2, \ldots, n; j = 1, 2, \ldots, n)$ is the dimension value corresponding to all concept nodes relative to the particular concept node C_i, defined as follows using Equation (2):

$$v_{i,j} = \begin{cases} d_j & \text{if } C_j \text{ is the relevancy node of concept } C_i \\ d_j & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

d_j is the local density of concept node C_j.

For example, for concept node C_2 in Figure 5, the concept node C_2 itself, its ancestor concept node C_1, and its descendent concept nodes C_4, C_5, and C_6 compose C_2's relevancy nodes. Their local densities d_2, d_1, d_4, d_5, and d_6 are used as C_2's dimension values. C_3 is not a relevancy node of C_2, so its dimension value for concept vector $\overrightarrow{C_2}$ is 0. If we were to list all concept nodes in sequential order of concept vectors according to the tree's breadth-first traversal sequence, we would have C_2's concept vector $\overrightarrow{C_2} = (1, 2, 0, 3, 3, 3)$. Similarly, C_1's concept vector is $\overrightarrow{C_1} = (1, 2, 2, 3, 3, 3)$. C_3's concept vector is $\overrightarrow{C_3} = (1, 0, 3, 0, 0, 0)$, and C_4's concept vector is $\overrightarrow{C_4} = (1, 2, 0, 3, 0, 0)$.

4.5 RNCVM Example

For example, we have hierarchy structure in Figure 6. The concept vectors for each concept node in the hierarchy are listed as follows.

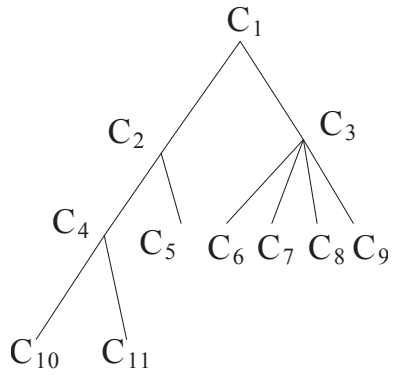

Fig. 6. Hierarchy taxonomy structure

We list all concept nodes' dimension values sequential order according to the structure's breadth-first traversal sequence as follows:

$$\overrightarrow{C_i} = (v_{i,1}, v_{i,2}, v_{i,3}, v_{i,4}, v_{i,5}, v_{i,6}, v_{i,7}, v_{i,8}, v_{i,9}, v_{i,10}, v_{i,11}), \quad i = 1, 2, \ldots, 11.$$

Particularly,

$$\begin{aligned}
\overrightarrow{C_1} &= (d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}, d_{11}) \\
\overrightarrow{C_2} &= (d_1, d_2, 0, d_4, d_5, 0, 0, 0, 0, d_{10}, d_{11}) \\
\overrightarrow{C_3} &= (d_1, 0, d_3, 0, 0, d_6, d_7, d_8, d_9, 0, 0) \\
\overrightarrow{C_4} &= (d_1, d_2, 0, d_4, 0, 0, 0, 0, 0, d_{10}, d_{11}) \\
\overrightarrow{C_5} &= (d_1, d_2, 0, 0, d_5, 0, 0, 0, 0, 0, 0) \\
\overrightarrow{C_6} &= (d_1, 0, d_3, 0, 0, d_6, 0, 0, 0, 0, 0) \\
\overrightarrow{C_7} &= (d_1, 0, d_3, 0, 0, 0, d_7, 0, 0, 0, 0) \\
\overrightarrow{C_8} &= (d_1, 0, d_3, 0, 0, 0, 0, d_8, 0, 0, 0) \\
\overrightarrow{C_9} &= (d_1, 0, d_3, 0, 0, 0, 0, 0, d_9, 0, 0) \\
\overrightarrow{C_{10}} &= (d_1, d_2, 0, d_4, 0, 0, 0, 0, 0, d_{10}, 0) \\
\overrightarrow{C_{11}} &= (d_1, d_2, 0, d_4, 0, 0, 0, 0, 0, 0, d_{11})
\end{aligned}$$

d_i is local density of concept node $C_i (i = 1, 2, \ldots 11)$.

Substitute the symbol with concrete values:

$$\overrightarrow{C_1} = (1, 2, 2, 2, 2, 4, 4, 4, 4, 2, 2)$$

$$\begin{aligned}
\overrightarrow{C_2} &= (1,2,0,2,2,0,0,0,0,2,2)\\
\overrightarrow{C_3} &= (1,0,2,0,0,4,4,4,4,0,0)\\
\overrightarrow{C_4} &= (1,2,0,2,0,0,0,0,0,2,2)\\
\overrightarrow{C_5} &= (1,2,0,0,2,0,0,0,0,0,0)\\
\overrightarrow{C_6} &= (1,0,2,0,0,4,0,0,0,0,0)\\
\overrightarrow{C_7} &= (1,0,2,0,0,0,4,0,0,0,0)\\
\overrightarrow{C_8} &= (1,0,2,0,0,0,0,4,0,0,0)\\
\overrightarrow{C_9} &= (1,0,2,0,0,0,0,0,4,0,0)\\
\overrightarrow{C_{10}} &= (1,2,0,2,0,0,0,0,0,2,0)\\
\overrightarrow{C_{11}} &= (1,2,0,2,0,0,0,0,0,0,2)
\end{aligned}$$

The similarity between any pair of words can be computed; for example, if we compute similarity values between $\overrightarrow{C_{10}}, \overrightarrow{C_{11}}$, their similarity values can be computed as follows:

$$sim(C_{10}, C_{11}) = \frac{\overrightarrow{C_{10}} \bullet \overrightarrow{C_{11}}}{\left\|\overrightarrow{C_{10}}\right\| \left\|\overrightarrow{C_{11}}\right\|} \approx 0.69.$$

4.6 Adjustments on RNCVM with Big Structure

Local density problem

According to the statement in Section 4.2, concerning density and similarity, the denser the structure of the concept nodes located, and the higher their similarity, but our measurement does not give perfect results. We explain it in the following.

In Figure 7, we derive our local density information from how many sibling concept nodes a concept node has; if a concept node has 1 sibling concept node, it has density 2, and if a concept node has 3 sibling concept nodes, it has density 4, while in the sparse part (left oval part) and dense part (right oval part) of Figure 7, the values of this measure for C_1, C_2, C_3, C_4 do not reflect our expectations that the tree with relatively denser representation of the concepts scored higher than other ones with sparser representation in situation, while the values of C_7, C_8, C_9, C_{10} of this measure do reflect the expectations.

In Figure 7, with the increase of the local density, if we do not expect the similarity value of C_3, C_4, C_5, C_6 decrease very fast or if we do not expect the similarity value of C_9, C_{10} increase very fast, we have a measure to balance that.

Our solution to balance this problem is shown in Figure 8. In our method, each additional sibling concept node of concept node C_1 contributes. Our balance measure is to let each additional sibling concept node contribute less; let $\beta(0 < \beta < 1)$ be the adjustment factor, we multiply $\beta, \beta^2, \beta^3, \ldots, \beta^{n-1}$ to each additional sibling concept nodes. So local density of concept nodes $C_1, C_2, \ldots, C_n$ is $(1 + \beta + \beta^2 + \ldots + \beta^{n-1})$ instead of n, $(1+\beta+\beta^2+\ldots+\beta^{n-1})$ decreased compared to n, and the decreased local density will generate increased similarity values between any two of these concept

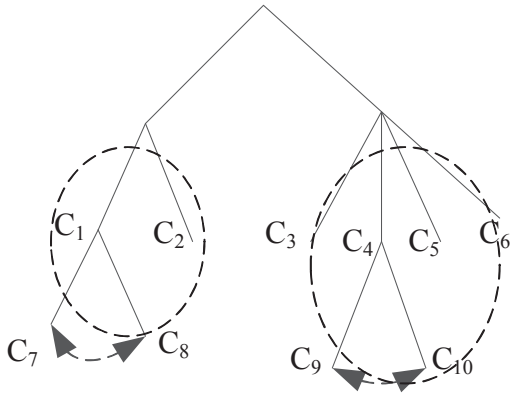

Fig. 7. Illustrating the local density problem

nodes. With the adjustment of $C_1, C_2, \ldots, C_n$, the similarity value of any word pair of $C_1, C_2, \ldots, C_n$ will increase and the similarity value of word pair from the descendent nodes of $C_1, C_2, \ldots, C_n$ will decrease.

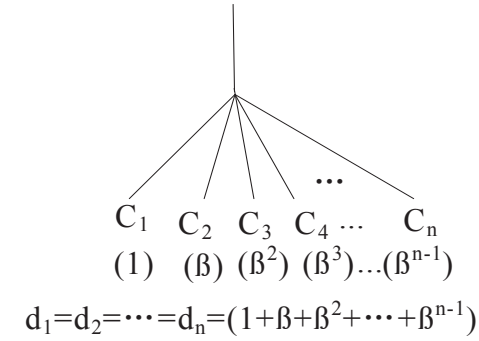

Fig. 8. The balanced measure for local density problem

Multiple descendent concept nodes problem

Our concept vector approach is a consequence of its origins in document-document similarity. Different feature terms may have different weights according to its importance for ranking the target text. There is similar problem for our internodes similarity computing. Different relevancy nodes may have different effect on similarity computing. Next we will elaborate on it.

Consider that we form our concept vector from a concept node's relevancy nodes. Observe the following structure in Figure 9; concept nodes C_3, C_4 have the same ancestor concept nodes in left and right trees, but left C_4 has none or few descendent concept nodes, while right C_4 has multiple descendent concept nodes, sim (C_3, C_4) of right tree is dramatically decreased compared to sim (C_3, C_4) of left tree when C_4's descendents increase. We know that C_3, C_4 inherit from the same ancestors C_1, C_2, they should have similar concept similarity in left and right. The only difference between the left and right parts of Figure 9 is that C_4 of the right part has multiple descendents.

The descendent concept nodes do have effect on similarity computation, but they should not have effect as much as their ancestor concept nodes. The main problem brought here is that, if the subtree of one concept node is richly represented where the subtree of the other concept node is sketchily represented, similarity comparison within the two parts will be incommensurable. According to our intuition, we allowed each additional layer of descendent concept nodes to contribute less, let it be $d_{L_i} * \alpha, d_{L_i}$ is the local density of the i^{th} descendent layer and $(DL_i)(i = 1, 2 \ldots n)$. α is the adjustment factor. This has the effect of trying to raise the cosine similarity between C_3 and C_4, but experiments have shown that with the increase of layer(i) – the linear decrease of α is not strong enough to overcome the increase of layer (i). Intuitively, the reason for this behavior is that the upper layer overlaps the lower layer, as i increases the magnitude of the "overlap" prorogated; so, as shown in Figure 10, we established $\alpha(0 < \alpha < 1)$ as our multiple-descendent-problem adjustment factor and multiplied $\alpha, \alpha^2, \alpha^3$ by each additional descendent layer. If C_4 has n layers of descendent concept nodes, the local densities of each layer of C_4's descendent nodes were adjusted as values $d_{L_1} * \alpha, d_{L_2} * \alpha^2, d_{L_3} * \alpha^3 \ldots d_{L_n} * \alpha^n$ illustrated in Figure 10. The decreased local densities of $d_{L_1} * \alpha, d_{L_2} * \alpha^2, d_{L_3} * \alpha^3 \ldots d_{L_n} * \alpha^n$ brought increased value of sim (C_3, C_4). We also made an experiment to let each layer of C_4's descendent nodes decline faster than $d_{L_1}, d_{L_2} * \alpha, d_{L_3} * \alpha^2 \ldots d_{L_n} * \alpha^{n-1}$, but empirical data show that they decline too fast, so that C_4's descendent nodes have little effect on the computation of similarity.

5 COMPARISON WITH RELATED WORK

Sufficient experiments are implemented to test RNCVM. First all the related methods and RNCVM are tested and analyzed with computer sciences department taxonomy, and then RNCVM is applied on WordNet and compared with all WordNet implemented methods.

5.1 Experiment 1: Computing Concept Similarity for Computer Science Department Taxonomy

Figure 11 is computer science department taxonomy, the edges are 'is-a' relations; in the real scenario, an assistant professor may take some Ph. D. courses, then there

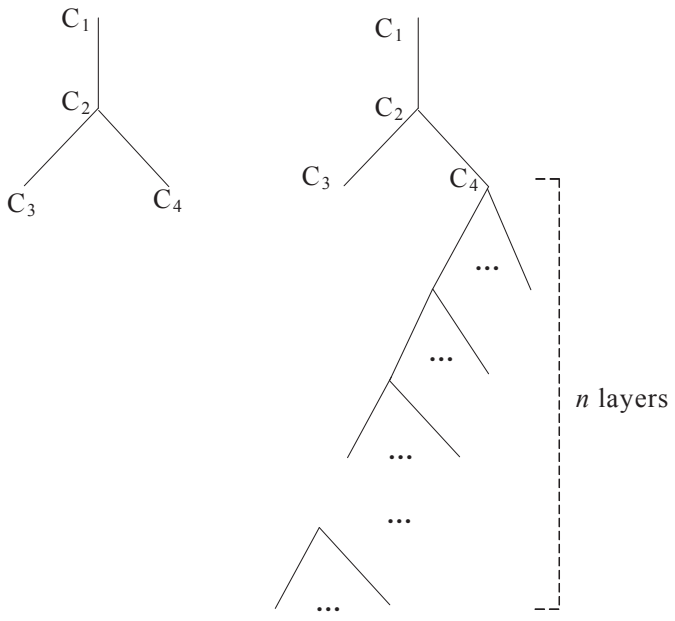

Fig. 9. The problem with multiple descendent concept nodes

should have a non 'is-a' edge between 'Assistant Professor' and 'Ph. D. Courses', but in this experiment we compute word similarity which involves only 'is-a' kind of relations according to Rada et al. [7]. So we omitted edges beside 'is-a' relations.

The only kind of methods which can be applied to the structure in the related work are the edge based (Rada et al.'s and Wu and Palmer's) methods.

The following Table 1 lists the comparison result.

Comparison	**A**	**B**	**C**
Rada et al.	no	no	yes
Wu and Palmer	no	yes	yes
RNCVM	**yes (part)**	**yes**	**yes**
Resnik *Jiang and Conrath* *Lin* *Leacock and Chodorow*	CANNOT BE APPLIED WITHOUT A TEXT CORPUS		
Banerjee and Pedersen *Patwardhan*	CANNOT BE APPLIED WITHOUT DICTIONARY GLOSSES		

Table 1. Comparison result

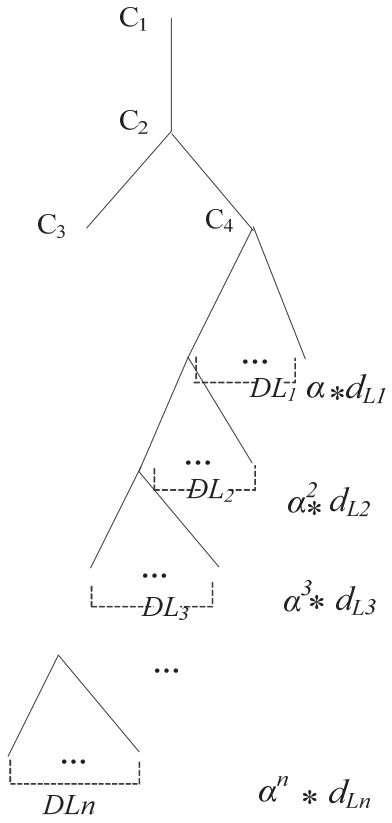

Fig. 10. The balanced measure for multiple descendent concept nodes problem

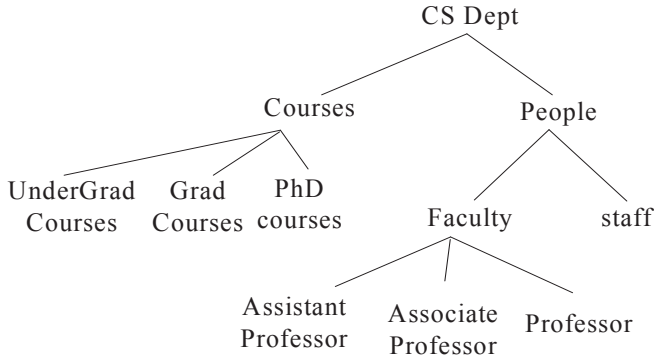

Fig. 11. Computer sciences department taxonomy

Note:

A: The denser the structure, the greater the similarity.

B: The deeper the depth, the greater the similarity.

C: The shorter the path, the greater the similarity.

Details are described in Section 4.2.

"yes" in the table symbolizes the measures computed by that approach coinciding with the ground truth,

"no" in the table symbolizes the measures computed by that approach contradicting with the ground truth.

This set of experiments shows that most of the existing method can not apply to the computer science department taxonomy in Figure 11, compared to Rada et al. and Wu and Palmer's methods, our method can give best expression to tree structure information.

5.2 Experiment 2: Testing RNCVM on WordNet

In order to fully evaluate our method, all WordNet implemented methods including Rada et al.'s [7], Leacock and Chodorow [1], Jiang and Conrath [5], Resnik [6], Lin [2], Hirst and St-Onge [3], Wu and Palmer [11], Banerjee and Pedersen's extended gloss overlap (lesk) [8], Patwardhan context vectors [9] are conducted to compare with our work. WordNet version 2.0 [13] was used. WordNet is organized by semantic relations, since a semantic relation is a relation between meanings, and meanings can be represented by synsets, it is natural to think of semantic relations as pointers between synsets. Given one word pair in WordNet, each word has multiple meanings; we choose the max similarity value of all meaning pairs as the similarity value of the word pair.

There are kinds of relations between WordNet nouns, hyper/hyponym ('is-a' link) and hol/meronym ('part-of' relation), and syn/antonym relations, etc. Hyper/hyponym relation is a main relation in WordNet, and it accounts for nearly 80 percent of all link types. We focus on the noun synsets and the hyper/hyponym relations.

A WordNet 2.0 noun has 25 unique beginners, which is abstracted into 10 hierarchies further. If the two words are in one synset, they have the same concept vectors, so their similarity (relatedness) value is 1.

Commonly used ground truth data to evaluate methods for computing the semantic similarity between words comes from an experiment carried by Miller and Charles [4]. 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).

Table 2 is the similarity result with different β, α and their correlation to human's judgments. Experiments show that the RNCVM gets the best correlation value 0.906

with human judgments when $\beta = 1$, $\alpha = 0.5$. From $\beta = 1$, we learn that the local density problem does not need to adjust for WordNet similarity computation. This experiment result of β, α value can be generalized to solve this kind of problem with our method.

Correlation	$\beta = 1$	$\beta = 0.75$	$\beta = 0.5$	$\beta = 0.25$	$\beta = 0$
$\alpha = 1$	0.876	0.722	0.721	0.717	0.71
$\alpha = 0.75$	0.887	0.735	0.733	0.728	0.721
$\alpha = 0.5$	0.906	0.745	0.743	0.739	0.733
$\alpha = 0.25$	0.879	0.738	0.756	0.714	0.759
$\alpha = 0$	0.863	0.765	0.767	0.769	0.771

Table 2. Corellation with different β, α

Table 3 lists the Miller's human judgements of 30 pairs of concepts similarity values and our similarity values with $\beta = 1$, $\alpha = 0.5$.

In order to make fair comparisons, an independent software package that would compute similarity values using previously established strategies while allowing the use of WordNet 2.0 was used. One freely available package is that of Siddharth Patwardhan and Ted Pederson [12]. Table 4 is Pearson's correlation coefficient from different approaches against the user studies.

This set of experiments shows that RNCVM performs best.

6 CONCLUSION

In this paper, a vector representation for concept nodes in a hierarchical taxonomy is proposed. Each concept node in the hierarchy has its density information, and each concept node has its ancestor and descendent nodes composing its relevancy nodes. A concept is then represented as a concept vector according to its relevancy nodes' density, and the similarity is computed with cosine similarity measure. The concept vector model contains full taxonomy structure information inherent and hidden in the tree. The method is adjustable in term of multiple sibling concept nodes and descendent concept nodes. Two sets of experiments were conducted, one is to apply the proposed method and related methods to a given domain taxonomy structure without additional dictionary and corpus information, and another experiment is to apply the proposed method to compute similarity for WordNet noun pairs. Experiment shows that our method performs best in the given structure, and provides improvement in the WordNet experiment result when compared with related methods. All of these experiments provide a strong validation for the proposed approach.

Acknowledgement

This project was supported by the project supported by Beijing Municipal Commission of Education (Grant No. KM201111417002), Funding Project for Academic

Word pair	**Human judgments**	**RNCVM** ($\beta = 1$, $\alpha = 0.5$)
car-automobile	3.92	1
gem-jewel	3.84	1
journal-voyage	3.84	0.89638
boy-lad	3.76	0.99873
coast-shore	3.7	0.99288
asylum-madhouse	3.61	0.99998
magician-wizard	3.5	1
midday-noon	3.42	1
furnace-stove	3.11	0.38562
food-fruit	3.08	0.56182
bird-cock	3.05	0.67782
bird-crane	2.97	0.66437
tool-implement	2.95	0.69087
brother-monk	2.82	0.99969
crane-implement	1.68	0.01411
lad-brother	1.66	0.19865
journey-car	1.16	0.0002
monk-oracle	1.1	0.01405
cemetery-woodland	0.95	0.01072
food-rooster	0.89	0.00972
coast-hill	0.87	0.68446
forest-graveyard	0.84	0.01071
shore-woodland	0.63	0.08009
monk-slave	0.55	0.01408
coast-forest	0.42	0.07916
lad-wizard	0.42	0.01412
chord-smile	0.13	0.0588
glass-magician	0.11	0.00949
noon-string	0.08	0.00023
rooster-voyage	0.08	0.00099

Table 3. Human judgements' result and ours

Human Resources Development in Institutions of Higher Learning under the Jurisdiction of Beijing Municipality (Grant No. PHR201108419 and Grant No. PHR 200907120), and the National Natural Science Foundation of China (Grant No. 6097-2045).

Similarity measure	**Correlation**
Jiang and Conrath	0.695
Hirst St. Onge	0.689
Leacoke and Chodorow	0.821
Resnik	0.775
Wu and Palmer	0.803
Lin	0.823
Banerjee and Pedersen	0.685
Patwardhan and Pedersen	0.778
RNCVM	0.906

Table 4. Pearson's correlation coefficient from different approaches against the user studies

REFERENCES

[1] Leacock, C.—Chodorow, M.: Combining Local Context and Wordnet Sense Similarity for Word Sense Identification. In WordNet, an Electronic Lexical Database, May 1998, pp. 265–283.

[2] Lin, D.: An Information-Theoretic Definition of Similarity. In Proceedings of ICML 1998, pp. 296–304.

[3] Hirst, G.—St-Onge, D.: Lexical Chains As Representations of Context for the Detection and Correction of Malapropisms. In WordNet, an Electronic Database, 1998, pp. 305–332.

[4] Miller, G. A.—Charles, W. G.: Contextual Correlates of Semantic Similarity. Language and Cognitive Processes, Vol. 6, 1991, No. 1, pp. 1–28.

[5] Jiang, J.—Conrath, D.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In Proceedings of COLING, Taiwan 1997.

[6] Resnik, P.: WordNet and Distributional Analysis: A Class-based Approach to Lexical Discovery. Proceedings of the AAAI Symposium on Probabilistic Approaches to Natural Language, San Joe, CA 1992.

[7] Rada, R.—Mili, H.—Bicknell, E.—Bletner, M.: Development and Application of a Metric on Semantic Nets. IEEE Transactions on Systems, Man, and Cybernetics, Vol. 19, 1989, No. 1, pp. 17–30.

[8] Banerjee, S.—Pedersen, T.: Extended Gloss Overlaps As a Measure of Semantic Relatedness. In Proceedings of IJCAI, Mexico 2003, pp. 805–810.

[9] Patwardhan, S.: Incorporating Dictionary and Corpus Information Into a Vector Measure of Semantic Relatedness. MSc Thesis, University of Minnesota 2003.

[10] Richardson, R.—Smeaton, A.F.: Using WordNet in a Knowledge-Based Approach to Information Retrieval. Workng Paper, CA-0395, School of Computer Applications, Dublin City University, Ireland 1995

[11] Wu, Z.—Palmer, M.: Verb semantics and lexical selection. In Proceedings of ACL 1994, pp. 133–138.

[12] Patwardhan, S.—Pederson, T.: Freely Similarity Package. Aailable on: `http://wn-similarity.sourceforge.net/`.

[13] Wordnet website, available on: `http://wordnet.princeton.edu/`.

Hong-Zhe Liu now is a Ph. D. candidate at School of computer and information technology of Beijing Jiao tong University, Beijing, China. She receive her M. Sc. degree in Computer Science from California State University, USA, in 1999. She has been an Assistant Professor of Computer Science Department, Beijing Union University, Beijing, China since 2009. Her research interests include semantic web, artificial intelligence and distributed systems.

Hong Bao is a Ph. D. candidate at School of computer and information technology of Beijing Jiao tong University, Beijing, China. He received his M. Sc. degree in Computer Science from Liverpool University, U.K., in 2005. He has been Professor of Beijing Union University since 2001, and now he is Vice President of Beijing Union University. His research interest include image processing and distributed systems.

De Xu is now a Professor at Institute of Computer Science and Engineering, Beijing Jiaotong University, Beijing. His research interests include database system and multimedia processing.