Ontology-based Semantic Relevance Measure

Sang Keun Rhee, Jihye Lee, and Myon-Woong Park

Intelligence and Interaction Research Center,
Korea Institute of Science and Technology,
39-1 Hawolgok-dong, Seongbuk-gu, Seoul, 136-791, Korea
{greyrhee,bluesea,myon}@kist.re.kr

Abstract. Semantic relevance among information resources can play an effective role in information retrieval, and there are several different approaches to measure semantic similarities. However, they are usually limited measuring similarities among structured concepts, and some also considers the similarities between individual resources but they are limited to to textual documents. This paper aims to extend the similarity to relevance so that even the relations between completely different resources can also be considered, and explore the possibility of measuring such semantic relevance not limited to a specific structure. The method presented in this paper is a novel measure for semantic relevance among resources that are organised within an ontological structure.

Key words: Semantic Relevance, Ontology, Information Search, Knowledge Sharing.

1 Introduction

There have long been various researches to enhance information retrieval and knowledge sharing, and ontology-based semantic technology opened a new possibility of improvement. Upon retrieving information, usually a list of results is provided based on the given query. However, the search result is generally concentrating on the relevance between the query and each resource, and to find other resources that are similar or relevant to a certain resource, users need to go back to the previous results list or initiate another query. Since each resource representing different knowledge is semantically connected to each other, measuring and utilising the semantic relevance between those resources can provide more flexible range of results which are not strictly limited to the given query. Such measure can also be used to provide a guideline or recommendation on what other resources are related and needed to be considered when the user is examining a certain resource.

In this paper, we propose a methodology to measure semantic relevance between resources, based on ontological representation. First, generic aspects of semantic similarity measure are briefly discussed with some existing methodologies. Then, an ontology-based methodology for measuring semantic relevance between resources is proposed and described in detail, and it is concluded with a discussion.

2 Generic Semantic Similarity Measure

Semantic similarity is the likeness of the semantic content among documents or the meaning among a set of structured terms, and there are several methods to discover the semantic similarities among concepts or resources. One of the oldest but still effective approaches is text-based technique, including vector-based model such as latent semantic analysis[1]. Also, there are several researches using direct relations between objects, based on either tree structure[3] or more generic and complex graph structure[2]. Their approach is typically (1) edge-based[4] that considers the number of edges between nodes but does not rate different characteristics for each edges, (2) node-based[4][3] that is more accurate but needs more computation for the content of each concept, or (3) hybrid[5][6] which merges both edge-based and node-based methods.

3 Semantic Relevance Measure

The method for semantic relevance measure proposed in this paper aims to calculate the relevance between resources as numeric values based on the semantic relations between concepts.

3.1 Ontological Representation

To measure semantic relevance between resources, all resources should be semantically structured and represented. An ontology would be the ideal representation for this purpose. Such ontology can be constructed in two different ways, depending on the purpose of the system and the type of resources being managed. Firstly, the ontology can be designed such that some classes contain resources as their instances whereas other classes represent other semantic concepts. For example, an ontology can have a class called *Documents* containing all the resources as documents form, and another class called *Person* containing personal profiles which are not regarded as resources hence not the target objects for searching but used as additional semantic information to produce, for instance, personalised search results. Secondly, all instances in all classes can represent resources. This is appropriate when everything described within the ontology is regarded as a resource. In this case, for above example, personal profiles are also regarded as resources hence they are also the target objects for searching.

Once the fundamental structure of the ontology - the concepts, attributes and properties - is defined, a set of rules should be defined to specify additional semantic information. The rules can be defined in the usual way, but for our methodology, some additional rules may be required to concentrate on representing the meaningful semantic relations *between resources*. As a simple example, suppose we have an ontology with a class *Resource* containing all the resources as its instances and another class *Topic* containing the topic hierarchy of topics as its instances(Fig. 1). Here, one may define a rule between a resource R_1 which

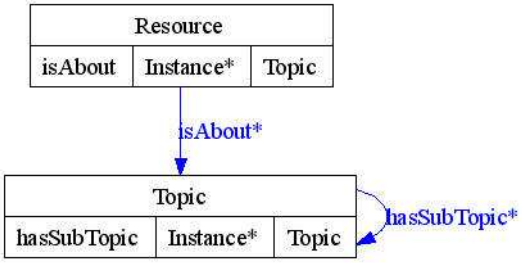

Fig. 1. A simple example ontology.

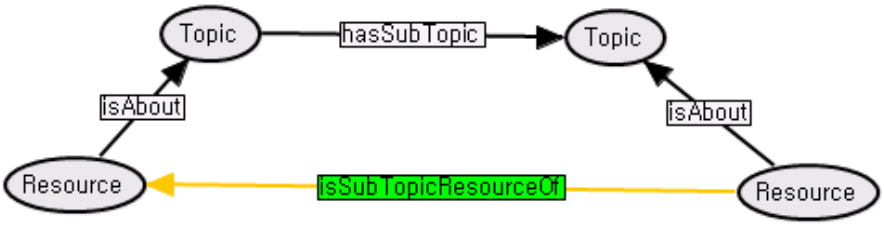

Fig. 2. An example rule.

is about Topic T_1 and another resource R_2 which is about Topic T_2, a subtopic of T_1(Fig. 2). This example rule can be defined in F-Logic[7] as following:

$$R_2[isSubTopicResourceOf \twoheadrightarrow R_1] \leftarrow$$
$$R_1 : Resource \wedge R_2 : Resource \wedge T_1 : Topic \wedge T_2 : Topic$$
$$\wedge\, R_1[isAbout \twoheadrightarrow T_1] \wedge R_2[isAbout \twoheadrightarrow T_2] \wedge T_1[hasSubTopic \twoheadrightarrow T_2]$$

The fundamental classes(concepts), properties(attributes and relations) and rules should be defined by domain experts, and the resources(instances) can be created by either a limited number of experts or the users themselves depending on the purpose and functionalities of the system.

3.2 Semantic Distance between Concepts

Once the ontology is defined, the next step is assigning a numeric value for each relation or rule, representing the semantic distance between concepts(classes). The semantic distance is the opposite term of semantic relevance - in other words, the higher the value is, the less relevant the given two concepts are. Depending on the structure of the ontology, the resources can be stored in a single class, multiple classes or all classes may contain resources. In any case, the ultimate concern at this stage is defining the semantic distance among classes containing

resources. Distances between other class can also be defined, but it would only be an intermediate step to assign the right value to rules.

Deciding the right value for each relation or rule can be tricky. The values can instantiated manually by domain experts, and some historical evaluation methods can be applied to update those values. Also, there are several researches on detecting the semantic similarity between concepts in both tree-based and graph-based structure, some of which are briefly mentioned in Section 2. However, the detailed methods for defining semantic distance between concepts are not the scope of this paper hence not listed here.

3.3 Semantic Relevance Measure between Resources

To infer the semantic relevance between resources(instances), we propose a graph-based relevance measure to exploit various relations and rules between resources. The final semantic relevance value can be determined via the following five steps:

Graph Creation The ontology is interpreted into a directed graph $G = (V, E)$ where:

- V is a set of nodes, representing all resources;
- E is a set of edges, representing relations and rules between resources.

Since additional rules as well as relations are interpreted as edges, it is possible that multiple edges $e_1(x, y), e_2(x, y), \ldots, e_n(x, y)$ are created between two nodes $x \in V$ and $y \in V$. If there is an edge whose final node is its initial node, such edge is ignored hence not included in the graph.

$$\forall x \in V : \qquad e(x, x) \notin E \tag{1}$$

For each edge $e_i(x, y) \in E$ where $x, y \in V$, a numeric value $d_{e_i(x,y)}$ is assigned as its label to represent the *distance* from node x to node y. This value is derived from the distance between concepts(Section 3.2). The relevance value between two nodes is the inverse of the distance value.

$$relevance = \frac{1}{distance} \tag{2}$$

Merging Multiple Edges Once the graph is constructed, the next step is to merge multiple edges into a single edge, where there are multiple edges with same direction between two adjacent nodes. For two adjacent nodes $x, y \in V$, edges $e_1(x, y), e_2(x, y), \ldots, e_n(x, y) \in E$ from node x to node y, and the distance value $d_{e_i(x,y)}$ for edge $e_i(x, y)$ where $1 \leq i \leq n$, the semantic relevance value R_{xy} is defined as follows:

$$R_{xy} = \sum_{i=1}^{n} \frac{1}{d_{e_i(x,y)}} \tag{3}$$

The distance value $d_{e'(x,y)}$ of the merged edge $e'(x, y)$ is:

$$d_{e'(x,y)} = \frac{1}{R_{xy}} \tag{4}$$

By applying this edge merging algorithm above, a new graph with no multiple edges with same direction between two adjacent nodes can be obtained. Fig. 3 shows an example illustration of merging multiple edges.

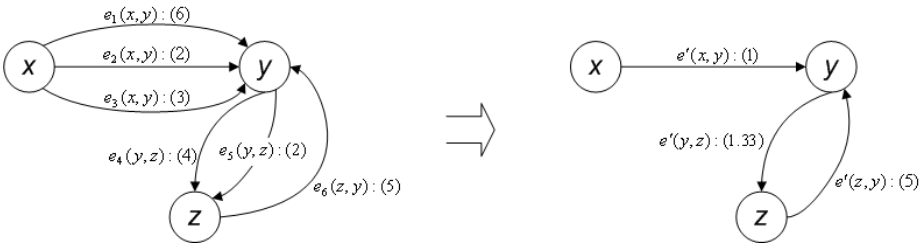

Fig. 3. An example illustration of merging multiple edges. The numbers in parentheses represents the distance values.

Paths Finding For two non-adjacent nodes, there can be multiple paths. Here, a path is a valid one if and only if no node is visited more than once(i.e. simple path). A simple graph traversal algorithm can be applied to obtain the list of paths for each pair of nodes.

Single Path Distance In a weighted graph, the weight of a path is the sum of the weights of each visited edges, and this can be applied to calculate the distance of a single path in our graph. Therefore, for a path $P(a_1a_n)$ which visits n nodes $a_1, a_2, \ldots, a_n \in V$, the distance of the path $D_{P(a_1a_n)}$ can be defined as follows:

$$D_{P(a_1a_n)} = \sum_{k=1}^{n-1} d_{e'(a_k a_{k+1})} \tag{5}$$

However, it is found that it can produce undesirable results, and the desired semantic distance value of a path should be greater than the simple sum of the edge distances. More precisely, the value added should become greater every time it passes another node. Therefore, the above equation is replaced by the following one with a value k.

$$D_{P(a_1a_n)} = \sum_{k=1}^{n-1} (k \times d_{e'(a_k a_{k+1})}) \tag{6}$$

Merging Multiple Paths The last step is to merge multiple paths between two nodes so that the final relation value can be obtained. This method is fundamentally the same as merging multiple edges.

For n paths $P_1, P_2, \ldots, P_n$ between two non-adjacent nodes $x, y \in V$, the semantic relation value R_{xy} is as follows:

$$R_{xy} = \sum_{i=1}^{n} \frac{1}{D_{P_k(xy)}} \tag{7}$$

4 Discussion

In this paper, we presented a novel measure of semantic relevance between resources based on ontological representation. Unlike previous approaches, this measures the semantic relevance between individual resources rather than concepts, and it also utilises all the different meaningful relations and rules for its calculation. It can be readily applied to complex ontologies where each resource have multiple different properties and their values so that many services can become possible or improved especially in information retrieval or recommender systems. This approach can be particularly useful in engineering design area where knowledge reuse and flexible knowledge searching is vital, but it can also be applied in many other areas such as knowledge portal, e-learning, etc.

Acknowledgments. This work was supported by the Korea Institute of Science and Technology(KIST) with the project IRS(Intelligent Responsive Space), which is a part of TSI(Tangible Space Initiative) project.

References

1. Landauer, T., Foltz, P. W., Laham, D.: Introduction to Latent Semantic Analysis Discourse Processes, 25. (1998) 259–284
2. Maguitman, A. G., Menczer, F., Roinestad, H., Vespignani, A.: Algorithmic Detection of Semantic Similarity. Proc. WWW2005, Chiba, Japan (2005)
3. Lin, D.: An Information-Theoretic Definition of Similarity Proc. 15th International Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA (1998) 296–304
4. Resnik, P.: Using Information Content to Evaluate Semantic Similarity in a Texonomy. IJCAI. (1995) 448–453
5. Jiang, J. J., Conrath, D. W.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. Proc. International Conf. Research on Computational Linguistics, Taiwan (1997)
6. Leacock, C., Chodorow, M.: Combining Local Context and WordNet Similarity for Word Sense Identification. Christiane Fellbaum(Eds.), WordNet: An Electronic Lexical Database. Cambridge, MA:MIT Press (1998) 265–283
7. Kifer, M., Lausen, G., Wu, J.: Logical Foundations of Object-Oriented and Frame-Based Languages. Journal of the ACM. (1995)