



BioTechnology

An Indian Journal

FULL PAPER

BTALJ, 8(11), 2013 [1572-1577]

Ontology similarity measure algorithm with operational cost and application in biology science

Wei Gao^{1*}, Li Shi²

¹School of Information Science and Technology, Yunnan Normal University, Kunming 650500, (CHINA)

²Institute of Medical Biology, Chinese Academy of Medical Sciences & Peking Union Medical College, Kunming, 650092, (CHINA)

ABSTRACT

Ontology similarity calculation is widely used in various fields such as biology science. In this paper, we propose new algorithms for ontology similarity measurement such that the new computational models consider operational cost in the real implement. Then, we apply it into biology science and it is highlighted that new calculating version is designed for multi-dividing setting. The experiment data on "Go" ontology demonstrate the new algorithm have higher efficiency in biology science application.

© 2013 Trade Science Inc. - INDIA

KEYWORDS

Ontology;
Similarity measure;
Ontology mapping;
Biology science;
Go ontology.

INTRODUCTION

Ontology abstracts certain application field of the real world into a set of concepts and relationships among concepts. Integrating the ontology into the technology of text information retrieval not only inherits the advantages of information retrieval but also overcomes the limitations that concepts information retrieval cannot deal with the relationships of the concepts. Now, ontology similarity computation is widely used in medical science, biology science (see^[1-4] for instance) and social science. As ontology used in information retrieval and biology science, every vertex can be regarded as a concept of ontology, measure the similarity of vertices using the information of ontology graph.

Let G be an ontology graph corresponding to ontology O , the aim of ontology similarity measure is to find a similarity function $Sim: V \times V \rightarrow [0, 1] \cup \{0\}$ which

maps each pair of vertices to a real number. A hot trick to get optimal similarity between vertices on ontology is by a function which maps ontology graph into a line and maps every vertex in graph into a real-value. Hence, the similarity between vertices is measured by the difference of their corresponding scores. Some efficient ontology algorithms can refer^[5-10]. Several theoretical analyses for ontology algorithm can refer^[11-18].

In this paper, we present a new ontology algorithm for ontology similarity measuring which considers operational cost in the computational model. Specifically, we propose several simultaneous processes for biology applications from optimistic bias and pessimistic bias view. The organization of rest paper is as follows: we describe the simultaneous process in next section; then, we present the new versions of simultaneous process for ontology algorithms; finally, experiment data is given to show that our new algorithms have high accuracy in biology science.

SETTING AND NOTATIONS

In this section, we present the standard simultaneous process technology which was proposed by Tulabandhula, and Rudinin^[19].

As we know, computer learning algorithms are employed to obtain predictions, and these predictions are usually help to make a policy or plot action, where there is a cost to implement such policy or action. Simultaneous process is a trick to align statistical modeling with decision making. It provided a way to propagate the uncertainty in predictive modeling to the uncertainty in operational cost (i.e., cost by the practitioner in solving the problem, and is regarded as regularization term in an objective function of computer algorithm). The technology admits to explore the range of operational costs associated with the collection of appropriate algorithm models and allow possible optimistic or pessimistic costs, which depend on the equilibrium coefficient. Any prior knowledge for the operational cost can help to restrict the hypothesis space of objective function and thus contribution to the algorithm.

The simultaneous process is a special class of decision theory. The goal of standard decision theory is to yield a policy for minimizing the expected cost. For propagating the uncertainty in modeling to the uncertainty in costs, simultaneous process determines the range of predictive models and corresponding policy decisions or actions. It admits a regularization term in algorithm model which contains encoding the policy (or action) with its associated cost and an adjustable equilibrium coefficient.

Let $S = \{(v_i, y_i)\}_{i=1}^n$ be a labeled training set, where $v_i \in V, y_i \in Y$. The goal of ontology algorithm is to learn an optimal function: $V \rightarrow Y$ (or $V \times V \rightarrow \{0\}$) from sample set S . Generally, f^* is obtained by minimization model:

$$f^* = \arg \min_{f \in F} \left(\sum_{i=1}^n l(f(v_i) - y_i) + \lambda N(f) \right),$$

where $l: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}^+$, $N: f \rightarrow \mathbb{R}^+$ is regularizer, λ is

called equilibrium coefficient. The term $\sum_{i=1}^n l(f(v_i) - y_i)$

is depended on error of function f on sample set S , and the term $\lambda N(f)$ rely on the smoothness of function f .

The typical loss functions used in such computational model are the square loss $l(f(v_i) - y_i) = (f(v_i) - y_i)^2$, exponential loss $l(f(v_i) - y_i) = e^{|f(v_i) - y_i|}$, logistic loss

$$l(f(v_i) - y_i) = \log(1 + (f(v_i) - y_i)), \text{ hinge loss}$$

$$l(f(v_i) - y_i) = \max\{1 - |f(v_i) - y_i|, 0\}.$$

Function class F is usually the class of all linear functionals.

Let $\{\tilde{v}_i\}_{i=1}^m$ be set of unlabeled vertices in ontology graph. The aim of organization is to produce a policy p which minimizes a certain operational cost $\text{Cost}(p, f^*, \{\tilde{v}_i\}_{i=1}^m)$. If the organization knew the $\{\tilde{v}_i\}_{i=1}^m$'s true labels in advance, it would select a policy to optimize the operational cost reckon on these labels without f^* .

However, these labels are unknown. We have no choose but to calculate the operational costs using the model's predictions. The main difference between the standard sequential process and simultaneous process is whether f^* is selected using or ignoring the knowledge of the operational cost.

The detail for sequential process computing can split following two steps (see^[19]).

A 1: Deduce function f^* from sample set S using standard learning algorithm:

$$f^* = \arg \min_{f \in F} \left(\sum_{i=1}^n l(f(v_i) - y_i) + \lambda N(f) \right).$$

A 2: Select policy p^* to minimize the operational cost:

$$p^* = \arg \min_{p \in P} (\text{Cost}(p, f^*, \{\tilde{v}_i\}_i)).$$

The operational cost $\text{Cost}(p, f^*, \{\tilde{v}_i\}_i)$ is the amount the organization will spend if policy p is selected in response to the values of $\{f(\tilde{v}_i)\}_i$.

The simultaneous process is obtained by combining A 1 and A 2 together. The optimistic bias is chosen if we would like to prove lower costs, and pessimistic bias is selected if we prefer higher costs. The equilibrium coefficient λ is used to control the degree of optimism or pessimism. That is to say, the optimistic bias lowers costs if there is uncertainty, but the pessi-

FULL PAPER

mistic bias raises them. The processes for simultaneous process are stated as follows (see^[19] for more detail).

B 1: Get a model f^* obeying one of the following:

Optimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\sum_{i=1}^n l(f(v_i) - y_i) + \lambda N(f) + C \text{Cost}(p, f^*, \{\tilde{v}_i\}_i) \right). \quad (1)$$

Pessimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\sum_{i=1}^n l(f(v_i) - y_i) + \lambda N(f) - C \text{Cost}(p, f^*, \{\tilde{v}_i\}_i) \right). \quad (2)$$

B 2: Yield the policy.

$$p^* = \arg \min_{p \in P} (\text{Cost}(p, f^*, \{\tilde{v}_i\}_i)). \quad (3)$$

Here C is a positive constant. In what follows, we always assume that $C > 0$ is a constant.

ASSOCIATED WITH ONTOLOGY SETTING

Gao and Gao^[5] presented a ontology algorithm based on pair computation:

$$f^* = \arg \min_{f \in F} \left(\sum_{i=1}^n l(f, v_i, v'_i, y_i) + \lambda N(f) \right). \quad (4)$$

Here, $f: V \times V \rightarrow \square \cup \{0\}$ which calculates the similarity of vertices directly and $S = \{(v_i, v'_i, y_i)\}_{i=1}^n$. By integrating (4) into standard simultaneous process (1-3), we get the first ontology algorithm stated below:

Algorithm 1: Ontology algorithm with operational cost based on pair calculating.

Step 1: Get a model f^* obeying one of the following:

Optimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\sum_{i=1}^n l(f, v_i, v'_i, y_i) + \lambda N(f) + C \text{Cost}(p, f^*, \{\tilde{v}_i\}_i) \right). \quad (5)$$

Pessimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\sum_{i=1}^n l(f, v_i, v'_i, y_i) + \lambda N(f) - C \text{Cost}(p, f^*, \{\tilde{v}_i\}_i) \right). \quad (6)$$

Step 2: Yield the policy.

$$p^* = \arg \min_{p \in P} (\text{Cost}(p, f^*, \{\tilde{v}_i\}_i)). \quad (7)$$

Agarwal and Niyogi [20] presented an ontology algorithm based on ranking learning method:

$$f^* = \arg \min_{f \in F} \left(\frac{1}{\binom{n}{2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^n l(f, (v_i, y_i), (v_j, y_j)) + \lambda N(f) \right). \quad (8)$$

By integrating (8) into standard simultaneous process (1-3), we get the second ontology algorithm stated below:

Algorithm 2: Ontology algorithm with operational cost based on pair calculating.

Step 1: Get a model f^* obeying one of the following:

Optimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\frac{1}{\binom{n}{2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^n l(f, (v_i, y_i), (v_j, y_j)) + \lambda N(f) + C \text{Cost}(p, f^*, \{\tilde{v}_i\}_i) \right). \quad (9)$$

Pessimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\frac{1}{\binom{n}{2}} \sum_{i=1}^{n-1} \sum_{j=i+1}^n l(f, (v_i, y_i), (v_j, y_j)) + \lambda N(f) - C \text{Cost}(p, f^*, \{\tilde{v}_i\}_i) \right). \quad (10)$$

Step 2: Yield the policy.

$$p^* = \arg \min_{p \in P} (\text{Cost}(p, f^*, \{\tilde{v}_i\}_i)). \quad (11)$$

Via this simultaneous process for ontology setting, we get the function f on $V \times V$ or V using algorithm (5-7) or (9-11). For (9-11), the ontology graph is mapped into a line consisting of real numbers. This similarity between two concepts can be measured by comparing the difference between their corresponding real numbers. For each $v \in V(G)$, $f(v)$ is a target value for vertex v using regular graph.

EXPERIMENT

We use "Go" ontology O_1 which was constructed

in^[21](Figure 1 shows the basic structure of O_1) for our experiment. $P@N$ (Precision Ratio^[22]) is used to measure the equality of the experiment. We first give the closest N concepts for every vertex on the ontology graph by expert, and then we obtain the first N concepts for every vertex on ontology graph by the algorithm and compute the precision ratio.

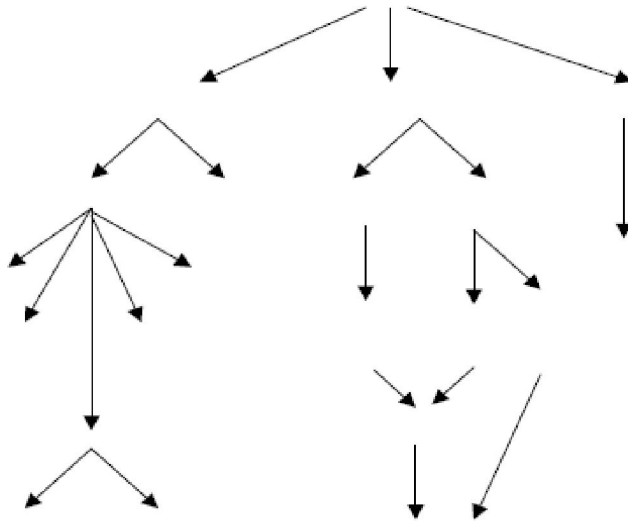


Figure 1 : “Go” ontology

Notice that there are “Molecular function”, “Biological process” and “Cellular component” three branches in GO ontology. Hence, the new version of simultaneous process is presented for this application. The first author of this paper raises the multi-dividing ontology algorithm as below (see^[11] for more detail). There is an instance space V from which vertices are drawn, and the learner is given a training sample $(S_1, S_2, \dots, S_k) \in V^{n_1} \times V^{n_2} \dots$ consisting of a sequence of training sample $S_a = (v_1^a, \dots, v_{n_a}^a)$ ($1 \leq a \leq k$). The goal is to learn from these samples a real-valued ontology score function $f: V \rightarrow \mathbb{R}$ that orders the future S_a vertices rank higher than S_b where $a < b$. We assume that instances in each S_a are drawn randomly and independently ac-

ording to some (unknown) distribution D_a on the instance space V respectively. Formally, the empirical model of multi-dividing ontology algorithm can be expressed as follows:

$$f^* = \arg \min_{f \in F} \left(\sum_{a=1}^{k-1} \sum_{b=a+1}^k \sum_{i: v_i \in S_a} \sum_{j: v_j \in S_b} l(f, v_i, v_j) + \lambda N(f) \right). \tag{12}$$

In terms of (12), we get the new simultaneous process algorithm as follows. Due to the structure of “GO” ontology graph, this version is suitable for “GO” ontology application.

Algorithm 3: Ontology algorithm with operational cost for multi-dividing setting.

Step 1: Get a model f^* obeying one of the following:

Optimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\sum_{a=1}^{k-1} \sum_{b=a+1}^k \sum_{i: v_i \in S_a} \sum_{j: v_j \in S_b} l(f, v_i, v_j) + \lambda N(f) + CCost(p, f^*, \{\tilde{v}_i\}_i) \right). \tag{13}$$

Pessimistic Bias:

$$f^* = \arg \min_{f \in F} \left(\sum_{a=1}^{k-1} \sum_{b=a+1}^k \sum_{i: v_i \in S_a} \sum_{j: v_j \in S_b} l(f, v_i, v_j) + \lambda N(f) - CCost(p, f^*, \{\tilde{v}_i\}_i) \right). \tag{14}$$

Step 2: Yield the policy.

$$p^* = \arg \min_{p \in P} (Cost(p, f^*, \{\tilde{v}_i\}_i)). \tag{15}$$

In the experiment, let F be a reproducing kernel Hilbert space (RKHS) of real-valued functions on V associated with a Mercer kernel $K: V \times V \rightarrow \mathbb{R}$, and $N: F \rightarrow \mathbb{R}$ be the regularizer defined by $N(f) = \|f\|_F^2$, where $\|\cdot\|_F$ denotes the RKHS norm in F .

At the same time, we employ ontology technologies in^[5-7] to the “GO” ontology. Calculate the

TABLE 1 : The experiment results of ontology similarity measure

	$P@3$ average precision ratio	$P@5$ average precision ratio	$P@10$ average precision ratio	$P@20$ average precision ratio
Algorithm3	56.44%	65.73%	78.39%	89.72%
Algorithm in [5]	43.56%	49.38%	56.47%	71.94%
Algorithm in [6]	42.13%	51.83%	60.19%	72.39%
Algorithm in [7]	46.38%	53.48%	62.34%	74.59%

FULL PAPER

accuracy by these three algorithms and compare the result to algorithm 3 using optimistic bias (13), part of the data refer to Table 1. From the experiment result display in TABLE 1, we arrived at the conclusion that our algorithm is more efficiently than algorithms raised in^[5-7], especially when N is larger enough. Therefore, the new ontology algorithm 3 for multi-dividing setting with operational cost has high efficiency.

CONCLUSION

As a data representation model, ontology has been widely used in biology science, and proved to have a high efficiency. In this paper, we apply the trick of simultaneous process to design the new ontology similarity computing model and use it in Go ontology. The new algorithm has high quality according to the experiment data above. More importantly, the algorithm reduces the operational cost in implement.

ACKNOWLEDGEMENTS

First, we thank the reviewers for their constructive comments in improving the quality of this paper. This work was supported in part by Key Laboratory of Educational Informatization for Nationalities, Ministry of Education, the National Natural Science Foundation of China (60903131), Key Science and Technology Research Project of Education Ministry (210210) and Jiangsu University Natural Science Research Project (10KJD52002). We also would like to thank the anonymous referees for providing us with constructive comments and suggestions.

REFERENCES

- [1] P.Mork, P.Bernstein; Adapting a generic match algorithm to align ontologies of human anatomy, In: 20th International Conf. on Data Engineering, Los Alamitos, CA, USA, Publisher: IEEE Comput.Soc., 787-790 (2004).
- [2] X.Su, J.Gulla; Semantic enrichment for ontology mapping, The 9th International Conference on Information Systems (NLDB), 217-228 (2004).
- [3] P.Lambrix, A.Edberg; Evaluation of ontology tools in bioinformatics, Paci Symposium on Biocomputing, New York: IEEE Computer Society Press, 529-600 (2003).
- [4] F.Gu, C.Cao, Y.Sui, W.Tian; Domain-Specific Ontology of Botany, JyComput. Sci. & Technol., **19(2)**, 238-248 (2004).
- [5] W.Gao, L.Liang; Ontology similarity measure by optimizing NDCG measure and application in physics education, Future Communication, Computing, Control and Management, **142**, 415-421 (2011).
- [6] Y.Gao, W.Gao; Ontology similarity measure and ontology mapping via learning optimization similarity function, International Journal of Machine Learning and Computing, **2(2)**, 107-112 (2012).
- [7] X.Huang, T.Xu, W.Gao, Z.Jia; Ontology similarity measure and ontology mapping via fast ontology method, International Journal of Applied Physics and Mathematics, **1(1)**, 54-59 (2011).
- [8] W.Gao, M.Lan; Ontology mapping algorithm based on ontology learning method, Microelectronics & computer, **28(9)**, 59-61 (2011).
- [9] Y. Wang, W.Gao, Y.Zhang, Y.Gao; Ontology similarity computation use ontology learning method, In Proceeding of the 3rd International Conference on Computational Intelligence and Industrial Application, 20-22 (2010).
- [10] X.Huang, T.Xu, W.Gao, S.Gong; Ontology similarity measure and ontology mapping using half transductive ranking, In Proceedings of 2011 4th IEEE International conference on computer science and information technology. Chengdu, China, 571-574 (2011).
- [11] W.Gao, T.Xu; Stability Analysis of Learning Algorithms for Ontology Similarity Computation, Abstract and Applied Analysis, Article ID 174802, **2013**, 9 (2013).
- [12] W.Gao, Y.Gao, Y.Zhang; Strong and weak stability of k -partite ranking algorithm. Journal of Information, **11(A)**, 4585-4590 (2012).
- [13] W.Gao, T.Xu; Characteristics of optimal function for ontology similarity measure via multi-dividing, Journal of networks, **8**, 1251-1259 (2012).
- [14] W.Gao, T.Xu, J.Gan, J.Zhou; Linear statistical analysis of multi-dividing ontology algorithm, Journal of Information and Computational Science, In press, (2014).
- [15] Y.Gao, W.Gao, L.Liang; Statistical characteristics for multi-dividing ontology algorithm in AUC criterion setting, Manuscript.
- [16] W.Gao, Y.Gao, Y. Zhang, L.Liang; Minimax learning rate for multi-dividing ontology algorithm, Manuscript.

- [17] W.Gao, L.Yan, L.Liang; Piecewise function approximation and vertex partitioning schemes for multi-dividing ontology algorithm in AUC criterion setting (I), International Journal of Computer Applications in Technology, In press, (2013).
- [18] L.Yan, W.Gao, J.Li; Piecewise function approximation and vertex partitioning schemes for multi-dividing ontology algorithm in AUC criterion setting (II). Journal of Applied Science, **16**, 3257-3262 (2013).
- [19] T.Tulabandhula, C.Rudin; Machine Learning with Operational Costs, Journal of Machine Learning Research, **14**, 1989-2028 (2013).
- [20] S.Agarwal, P.Niyogi; Generalization bounds for ranking algorithms via algorithmic stability, Journal of Machine Learning Research, **10**, 441-474 (2009).
- [21] <http://www.geneontology.org>.
- [22] N.Craswell, D.Hawking; Overview of the TREC 2003 web track, In Proceeding of the Twelfth Text Retrieval Conference. Gaithersburg, Maryland, NIST Special Publication, 78-92 (2003).