Fuse and Divide Technologies for Sparse Vector Learning in Ontology Algorithms

Xiao Yu[1], Jianzhang Wu[2*], Wei Gao[3]

***Abstract*—As a semantic representation model, ontology has penetrated into all areas of natural science and social science. The core issue of ontology applications is similarity computing between ontology concepts. In this article, we report a sparse vector learning algorithm for ontology similarity measure and ontology mapping in terms of sub-gradient calculating and iterative computation. The main procedure of our iterative computation is based on the tricks of fuse and divide. The simulation experimental results show that the new proposed algorithm has high efficiency and accuracy in ontology similarity measure and ontology mapping in biology and physical education science.**

***Index Terms*—Ontology, similarity measure, ontology mapping, sparse vector, fuse and divide**

I. INTRODUCTION

THE concept "ontology" was originally used in philosophy which represents the essence of the matter and the necessary connection between things. Later, as a knowledge representation and conceptual shared model, ontology was brought into computer science where it turned out to be useful in image retrieval, knowledge management and information retrieval search extension. What's more, as an effective concept semantic model, ontology also finds its place in the other disciplines like social science, medical science, biology science, pharmacology science and geography science (for instance, see Raad and Evermann [1], Ali et al., [2], Gao and Shi [3], Gao et al., [4], and Gopal and Gowri Ganesh [5]).

In fact, the ontology model is a graph G=(V,E), each vertex v in an ontology graph G stands for a concept and each edge $e=v_iv_j$ on an ontology graph G stands for a relationship between concepts v_i and v_j. The ontology similarity measure aims to find a similarity function Sim: $V\times V \rightarrow \mathbb{R}^+ \cup \{0\}$ through which each pair of vertices is mapped to a real number. The ontology mapping aims to bridge the link between two or more ontologies. Let G_1 and G_2 be two ontology graphs corresponding to ontology O_1 and O_2 respectively. For each $v\in G_1$, determine that a set $S_v\subseteq V(G_2)$ where the concepts corresponding to vertices in S_v is semantically close to the concept corresponding to v. There is one method to get such mapping, which is, for each $v\in G_1$, computing the similarity $S(v,v_j)$ where $v_j\in V(G_2)$ and choose a parameter 0<M<1. Then S_v is a collection such that the element in S_v meets $S(v,v_j)\geq M$. In this way, the key of ontology mapping is to get a similarity function S and select a suitable parameter M.

In the past few years, ontology similarity-based technologies were used in various applications. Amini et al., [6] reported a technology for the integration of multiple domain taxonomies to build reference ontology on which the scholars' background knowledge could be profiled. Sun et al., [7] studied uncertain features in the generation and application of metadata by virtue of ontology similarity measuring and the technologies of probability statistic. Zhong et al., [8] raised a hybrid assessment algorithm based on ACT-R cognitive learning theory and ontology knowledge map. Mahfoudh et al., [9] proposed a formal approach for evolving ontologies using Typed Graph Grammars. Gao et al., [10] presented an ontology sparse vector algorithm for ontology applications by virtue of the stopping condition judgment and dual problem solution.

Concerning ontology similarity measure and ontology mapping, several effective learning tricks were introduced and proved to be in high efficiency. Gao et al., [11] raised new ontology mapping algorithm by means of harmonic analysis and diffusion regularization on hypergraph. Gao and Shi [12] proposed a new ontology similarity computation technology, which helps to make the operational cost considered in the real implement. Gao and Xu [13] presented the ontology similarity measuring and ontology mapping algorithms on basis of minimum error entropy criterion. Wu et al., [14] reported a bilinear model for ontology mapping.

Furthermore, several papers contributed to the theoretical analysis of ontology learning algorithm. Gao et al., [15] studied the strong and weak stability for k-partite ranking based ontology algorithm. Gao and Xu [16] presented the uniform stability analysis of ontology learning computation. Considering the gradient learning algorithm for ontology computing, Gao and Zhu [17] found the way to get some generation bound. Gao et al., [18] proposed the piecewise function approximation and vertex partitioning schemes for multi-dividing ontology algorithm in AUC criterion setting.

In this paper, we present a new ontology sparse vector learning algorithm for ontology similarity computation and ontology mapping by virtue of fuse and divide technologies. By means of the sparse vector, the ontology graph is mapped into a real line and vertices are mapped into real numbers. Then the similarity between vertices is measured by the difference between their corresponding real numbers. The

Manuscript received Dec. 05, 2015; revised Feb. 23, 2016. This work was supported in part by supported in part by the Key Laboratory of Educational Informatization for Nationalities, Ministry of Education, and NSFC (No.11401519).

X. Yu is with the School of Continuing Education, Southeast University, Nanjing 210096, China (e-mail: pp_xyu@seu.edu.cn).

J. Z. Wu is with School of Computer Science and Engineer, Southeast University, Nanjing 210096, China (corresponding author to provide phone: 025-84335843; fax: 025-84335843; e-mail: jzwu@njnet.edu.cn).

W. Gao is with the School of Information and Technology, Yunnan Normal University, Kunming 650500, China (e-mail: gaowei@ynnu.edu.cn).

rest of the paper is arranged as follows: we present the notations and setting in Section 2; the ontology optimization algorithm and iterative strategies are raised in Section 3, and the technologies to tackle the details in algorithm are included in this section; at last, the experiments on biology, physical education, plant science and chemical index data are designed to show the efficiency of the algorithm.

II. Setting

Let *V* be an instance space. For each vertex in ontology graph, a *p* dimension vector expresses information including its name, instance, attribute and structure, and semantic information of the concept which corresponds to the vertex and that is contained in name and attributes components of its vector. Let $v=\{v_1,\cdots,v_p\}$ be a vector that corresponds to a vertex *v*. To promote the representation, we try confusing the notations and using *v* to denote both the ontology vertex and its corresponding vector. The ontology learning algorithms are set to get an optimal ontology (score) function $f: V\to\mathbb{R}$, and the similarity between two vertices is judged by the difference between two corresponding real numbers. The core of this algorithm is dimensionality reduction, i.e., choosing one dimension vector to express *p* dimension vector. In specific, an ontology function *f* is a dimensionality reduction function $f: \mathbb{R}^p\to\mathbb{R}$.

Since all the information in the ontology graph (including the vertex concept, attribute and the neighborhood structure) is contained in the corresponding vector, it's always with high dimension. For instance, the information of all genes may be contained in only a vector in biological ontology. In addition, ontology structure becomes very complicated because of the ontology graph with large number of vertices, and the GIS (Geographic Information System) ontology may be taken as a typical example. A result may be gained from these factors, that is, the similarity calculation of ontology application is very large. However, in practice, it is small part of the vector components that determine the similarity between the vertices. For example, in biological ontology, a genetic disease often results from a small number of diseased genes, having nothing with most other genes. Moreover, proofs could be found in geographic information system ontology as well. If an accident happens and causes casualties, what we need to do is find the nearest hospital, having nothing with the neighboring schools and shops. In other words, we just need to find neighborhood information that meet our specific requirements on the ontology graph. Therefore, sparse ontology algorithm researches have attracted great academic and industrial interests.

In the practice implement, one sparse ontology function is expressed by

$$f_\beta(v)=\sum_{i=1}^{p}v_i\beta_i+\delta. \tag{1}$$

Here $\boldsymbol{\beta}=(\beta_1,\cdots,\beta_p)$ is a sparse vector and δ is a noise term. The sparse vector $\boldsymbol{\beta}$ is to shrink irrelevant component to zero. To determine the ontology function *f*, we should learn the sparse vector $\boldsymbol{\beta}$ first.

In our paper, we consider the general versions for learning $\boldsymbol{\beta}$. Let $\{v_i,y_i\}_{i=1}^n$ be a sample set with *n* vertex, $\boldsymbol{V}\in\mathbb{R}^{n\times p}$ be the matrix of *n* samples such that each sample vertex lies in a *p* dimension space, and $\boldsymbol{y}=(y_1,\cdots,y_n)\in\mathbb{R}^n$ be the vector of outputs of the these *n* sample vertex. Hence, the regression function (1) can be expressed as the linear model:

$$\boldsymbol{y}=\boldsymbol{V\beta}+\delta, \tag{2}$$

where δ is the *n* dimension vector for noise distributed as $N(0,\sigma^2I_{n\times n})$.

The ontology regression obtains an estimate of the sparse vector by solving the following optimization problem:

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} l(\boldsymbol{\beta})+\lambda\|\boldsymbol{\beta}\|_1, \tag{3}$$

where $l(\boldsymbol{\beta})=\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{V\beta}\|_2^2=\frac{1}{2}(\boldsymbol{y}-\boldsymbol{V\beta})^{\mathrm{T}}(\boldsymbol{y}-\boldsymbol{V\beta})$ is the loss term, $\|\boldsymbol{\beta}\|_1=\sum_{i=1}^p|\beta_i|$ is the l_1-norm balance term that measures the sparseness of vector $\boldsymbol{\beta}$, and $\lambda>0$ is the balance parameter which controls the sparsity level. On the selection of the balance parameter λ, readers can refer to Mancinelli et al., [19], Zhu et al., [20], Mukhopadhyay and Bhattacharya [21], Ishibuchi and Nojima [22], Zhang et al., [23] and Varmuza et al., [24] for more details about the method of cross-validation. In our article, cross-validation is not considered for dealing with parameters, and the idea of fuse and divide will be employed in next section.

It restricts the ontology applicability to complex high dimensional situation in many applied fields if we don't limit any ontology structure among the input vertex (variables). More structured ontology restrictions on the input vertices such as pairwise similarities should be utilized by using a more complicated ontology sparsity penalty that brings ontology sparsity patterns among related vertices. In this paper, we use $\Psi(\boldsymbol{\beta})$ to denote the general structured sparsity-inducing ontology penalty term, and the ontology optimization problem (3) can be extended as:

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p} R(\boldsymbol{\beta})=l(\boldsymbol{\beta})+\Psi(\boldsymbol{\beta})+\lambda\|\boldsymbol{\beta}\|_1. \tag{4}$$

In this paper, we only consider $\Psi(\boldsymbol{\beta})=\gamma\sum_{i=1}^{p-1}|\boldsymbol{\beta}_{i+1}-\boldsymbol{\beta}_i|$ and $R(\boldsymbol{\beta})$ in (4) also denoted by $R_{\lambda,\gamma}(\boldsymbol{\beta})$.

III. Algorithm Description

A. Notations and Basic Line of Algorithm

In this section, we present the main algorithm in our paper. The implement of our ontology algorithm is based on iterative computation. Note that there are two main parameters λ and γ in computation model (4). Hence, we should consider the treatment of combination (λ,γ). Our idea is progressive: at the beginning, we set $\gamma=0$, and then increase γ. We say coefficients are being fused if neighboring coefficients are forced to be equal to each other in this increasing process. Let F_i (i=1,…, $n_F(\gamma)$) be the fused collections of coefficients at γ where $n_F(\gamma)$ is the number of such collections. Assume that $F_i=\{k \mid l_i \le k \le u_i\}$ for ensuring these collections to be valid. Use $\beta_{F_i}(\gamma)$ for any $\beta_k(\gamma)$ with $k\in F_i$ and restrict the dependency of F_i onto γ. Furthermore, we associate these collections with coefficients: $\beta_{F_i} \neq 0$ if F_i are active at present and otherwise β_{F_i} = 0. For this purpose, the assumption of initial fused collections should be slightly modified. Set $s_k = \text{sign}(\boldsymbol{\beta}_k)$ if $\boldsymbol{\beta}_k \neq 0$ and $s_k \in [-1, 1]$ otherwise. And, let $t_{v_k v_l} = \text{sign}(\boldsymbol{\beta}_k - \boldsymbol{\beta}_l)$ if $\boldsymbol{\beta}_k \neq \boldsymbol{\beta}_l$ and $t_{v_k v_l} \in [-1, 1]$ if $\boldsymbol{\beta}_k = \boldsymbol{\beta}_l$. Notice that the k-th sub-gradient for $R_{\lambda,\gamma}(\boldsymbol{\beta})$ can be stated as

$$\frac{\partial R_{\lambda,\gamma}(\boldsymbol{\beta})}{\partial \beta_k} = -(\boldsymbol{V}^{\mathrm{T}}\boldsymbol{y})_k + (\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}\boldsymbol{\beta})_k + \lambda s_k + \gamma \sum_{v_l : v_k v_l \in E} t_{v_k v_l} .$$

If $\boldsymbol{\beta}_k = \boldsymbol{\beta}_l$ then the variables are considered fused, and unfused if $\boldsymbol{\beta}_k \neq \boldsymbol{\beta}_l$. Under this setting, all fused variables in collection are either active or not: active if $\boldsymbol{\beta}_k \neq 0$, and unactive if $\boldsymbol{\beta}_k$ =0. Hence, we should activate a whole collection if we want to activate coefficients. In this way, s_k is used to instead of $\boldsymbol{\beta}_k$ for keeping track of collections of inactive variables.

Specifically, let $p_F(\gamma)$ be the number of fused variable collections for parameter γ (assume λ is fixed). For the collections F_i (i=1,…, $n_F(\gamma)$) to be valid, we suppose that $\bigcup_{i=1}^{n_F(\gamma)} F_i = \{1,\cdots,n\}$; $F_i \cap F_j = \varnothing$ established for $i \neq j$; $\boldsymbol{\beta}_k(\gamma)$ = $\boldsymbol{\beta}_l(\gamma)$ and $s_k(\gamma)$ = $s_l(\gamma)$ if k, $l \in F_i$; $\boldsymbol{\beta}_k(\gamma) \neq \boldsymbol{\beta}_l(\gamma)$ or $s_k(\gamma) \neq s_l(\gamma)$ for all parameters in an interval $(\gamma, \gamma+\varepsilon)$ with certain positive number ε if $k\in F_i$, $l\in F_j$, $i \neq j$ and F_i and F_j have a connecting edge; v_k and v_l are connected in ontology graph G through a path with vertices in F_i if $k, l\in F_i$.

For parameter γ^0 (assume that λ is fixed), the active fused collections $A(\gamma^0)$ and inactive fused collections $N(\gamma^0)$ are defined by

$$A(\gamma^0) = \{i \mid \beta_{F_i}(\gamma) \neq 0 \text{ for } \gamma \in (\gamma^0, \gamma^0+\varepsilon)\}$$

and

$$N(\gamma^0) = \{1,\cdots, p_F(\gamma^0)\} - A(\gamma^0).$$

Here, a collection is defined as active at γ^0 if $\beta_{F_i}(\gamma) \neq 0$ for $\gamma > \gamma^0$ instead of if $\beta_{F_i}(\gamma^0) \neq 0$. We will show how $\boldsymbol{\beta}_k$, s_k and $t_{v_k v_l}$ change with γ in the following contents.

The line of steps for the derivation of the algorithm list as follows. First, we will combine the conditions of fused and active collections into the $R_{\lambda,\gamma}(\boldsymbol{\beta})$ and then determine the derivative of $\beta_{F_i}(\gamma)$ regarding to γ. Second, we present how s_k and $t_{v_k v_l}$ change with γ and apply it to explain if collections of variables are active or inactive and have to be fused or divided. Then, we manifest that the solution is piecewise linear with respect to γ.

B. Sub-gradient Computation for the Algorithm

We state a predictor matrix that combines the information in view of active collections and collections of fused variables. For i = 1,⋯, $p_F(\gamma)$, using F_i, the elements in matrix $\boldsymbol{V}^F \in \mathbb{R}^{n\times p_F(\gamma)}$ is denoted by $v_i^F = \sum_{k\in F_i} v_k$. Thus, we combine the active collections into $\boldsymbol{V}^{F,A} \in \mathbb{R}^{n\times|A(\gamma)|}$ by virtue of dropping all columns in $\boldsymbol{V}^F$ that don't correspond to active collections. Similarly, $k \in F_i$ with $i\in A(\gamma)$, we refer to the columns of $\boldsymbol{V}$ corresponding to coefficients in active collections in terms of $\boldsymbol{V}^A$. Let β^F be a vector satisfies $\beta_i^F = \beta_{F_i}$. Then, the constrained ontology model combining active and fused collections is denoted by

$$\begin{aligned} &R_{F,A,\lambda,\gamma}(\boldsymbol{\beta}) \\ &= \frac{1}{2}(\boldsymbol{y}-\boldsymbol{V}^F\boldsymbol{\beta}^F)^{\mathrm{T}}(\boldsymbol{y}-\boldsymbol{V}^F\boldsymbol{\beta}^F) + \lambda \sum_{i\in A(\gamma)} |F_i||\beta_i^F| \\ &+\gamma \sum_{i<j} \left|\{(k,l): k\in F_i, l\in F_j\}\right| \left|\beta_i^F - \beta_j^F\right|. \end{aligned} \quad (5)$$

If the active and fused collections are correct, then minimize of $R_{F,A,\lambda,\gamma}(\boldsymbol{\beta})$ is equal to the minimize $R_{\lambda,\gamma}(\boldsymbol{\beta})$. Assume s_{F_i} is a simple version for s_k with $k\in F_i$ like β_{F_i}. The sub-gradient for this restricted ontology model with respect to γ can be expressed as

$$\begin{aligned} \frac{\partial R_{F,A,\lambda,\gamma}(\boldsymbol{\beta})}{\partial \beta_i^F} &= -\boldsymbol{V}_i^{F,T}\boldsymbol{y} + \boldsymbol{V}_i^{F,T}\boldsymbol{V}^F\beta^F + \lambda|F_i|s_{F_i} \\ &+\gamma \sum_{j\in A(\gamma)} \left|\{(k,l): k\in F_i, l\in F_j\}\right| \text{sign}(\beta_{F_i} - \beta_{F_j}) = 0. \end{aligned}$$

Let $\boldsymbol{a}$ and $\boldsymbol{b}$ be vectors defined by

$$a_i = |F_i| s_{F_i},$$

and

$$b_i = \sum_{i\neq j} \left|\{(k,l): k\in F_i, l\in F_j\}\right| \text{sign}(\beta_{F_i} - \beta_{F_j}).$$

Then the above sub-gradient can be re-written as

$$\frac{\partial R_{F,A,\gamma}(\boldsymbol{\beta})}{\partial \beta^F} = -\boldsymbol{V}_i^{F,T}\boldsymbol{y} + \boldsymbol{V}_i^{F,T}\boldsymbol{V}^F\beta^F + \lambda\boldsymbol{a} + \gamma\boldsymbol{b} = 0.$$

For $i \in A(\gamma)$, we deduce that $s_i^F(\gamma) = s_{F_i}(\gamma) = 1$ with respect to γ and for $i \in N(\gamma)$ we infer $\beta_i^F(\gamma) = \beta_{F_i}(\gamma) = 0$. Therefore, we obtain $(\boldsymbol{V}^{F,A})^{\mathrm{T}}\boldsymbol{V}^{F,A}\frac{\partial \beta^{F,A}}{\partial \gamma} + \boldsymbol{b}^A = 0$ and $(\boldsymbol{V}^{F,N})^{\mathrm{T}}\boldsymbol{V}^{F,A}\frac{\partial \beta^{F,A}}{\partial \gamma} + \lambda\frac{\partial \boldsymbol{a}^N}{\partial \gamma} + \boldsymbol{b}^N = 0$.

Furthermore, these can be solved by

$$\frac{\partial \beta^{F,A}}{\partial \gamma} = ((\boldsymbol{V}^{F,A})^{\mathrm{T}}\boldsymbol{V}^{F,A})^{-1} + \boldsymbol{b}^A \tag{6}$$

and

$$\frac{\partial \boldsymbol{a}^N}{\partial \gamma} = -\frac{1}{\lambda}((\boldsymbol{V}^{F,N})^{\mathrm{T}}\boldsymbol{V}^{F,A}\frac{\partial \beta^{F,A}}{\partial \gamma} + \boldsymbol{b}^N). \tag{7}$$

from which we yield $\frac{\partial s_{F_i}}{\partial \gamma} = \frac{1}{|F_i|}\frac{\partial a_i^N}{\partial \gamma}$. Therefore, we have calculated the derivative of β^F and s^F by virtue of (6).

We next determine the properties of $t_{v_k v_l}$ regarding with γ or equivalently of $\tau_{v_k v_l} = \gamma t_{v_k v_l}$ for dividing collections. By assuming the solution at γ that meets the sub-gradient equations, we verify, for any coefficient $k \in F_i$,

$$-\boldsymbol{v}_k^{\mathrm{T}}\boldsymbol{y} + \boldsymbol{v}_k^{\mathrm{T}}\boldsymbol{V}^A\boldsymbol{\beta}^A + \lambda s_k + \gamma \sum_{l:(v_k,v_l)\in E, k,l\in F_i} t_{v_k v_l} = 0.$$

It obtained by grouping the $t_{v_k v_l}$ according to whether k and l in the same or different collections. By taking the derivative regarding with γ, we infer

$$\boldsymbol{v}_k^{\mathrm{T}}\boldsymbol{V}^A\frac{\partial \beta^A}{\partial \gamma} + \lambda\frac{\partial s_k}{\partial \gamma} + \sum_{l:(v_k,v_l)\in E, k,l\in F_i}\frac{\partial \tau_{kl}}{\partial \gamma} + \sum_{l:(v_k,v_l)\in E, k,l\in F_i} t_{v_k v_l} = 0.$$

For determining $\frac{\partial \tau_{kl}}{\partial \gamma}$, we use the maximum flow setting in ontology graph and let $t_{v_k v_l} = \mathrm{sign}(\beta_k - \beta_l)$ for $k \in F_i$ and $l \in F_i$. The push p_k on vertex v_k is stated by

$$p_k = -\boldsymbol{v}_k^{\mathrm{T}}\boldsymbol{V}^A\frac{\partial \beta^A}{\partial \gamma} - \lambda\frac{\partial s_k}{\partial \gamma} - \sum_{l:(v_k,v_l)\in E, l\notin F_i} t_{v_k v_l}$$

$$= \sum_{l:(v_k,v_l)\in E, k,l\in F_i}\frac{\partial \tau_{kl}}{\partial \gamma} \quad \text{for } k=1,\ldots,p.$$

Let f_{kl} be the maximal flow from vertex v_k to vertex v_l in F_i. By setting $\frac{\partial \tau_{kl}}{\partial \gamma} = f_{kl}$, we use the maximum flow technology to determine $\frac{\partial \tau_{kl}}{\partial \gamma}$.

C. Maximum flow ontology graph and Calculating of Δ

For certain γ^0, let $F_1,\ldots, F_{pF(\gamma^0)}$ be the valid grouping of the variables. Let G_i be the sub ontology graph restricted to the vertices in the collection F_i. Let $\tilde{G}_i = (\tilde{V}_i, \tilde{E}_i, \tilde{C}_i)$ be the F_i associated maximum-flow ontology graph defined as follows: for each of the sub ontology graph G_i, we artificial add a source vertex v_r and a sink vertex v_s, then $\tilde{V}_i = V_i \cup (v_r, v_s)$; $\tilde{E}_i = E_i \cup \{(v_r, v_l): p_l > 0\} \cup \{(v_k, v_s): p_k < 0\}$; and for $k,l \in F_i$,

$$(c_{kl}, c_{lk}) = \begin{cases} (-\infty, +\infty) & \text{if } \tau_{kl} \in (-\gamma, \gamma) \\ (1, +\infty) & \text{if } \tau_{kl} = \gamma \\ (-\infty, 1) & \text{if } \tau_{kl} = -\gamma \end{cases},$$

$(c_{rl}, c_{lr}) = (p_l, 0)$ for the edge from the source vertex v_r and $(c_{ks}, c_{sk}) = (-p_k, 0)$ for the edge from the sink vertex v_k. At last, $\tilde{C}_i$ is determined by $\tilde{C}_i = \{c_{kl} : k, l \in \tilde{V}_i\}$.

Let $\beta_k(\gamma^0)$, $s_k(\gamma^0)$ and $\tau_{kl}(\gamma^0)$ be the solution to the fused ontology problem with parameter $\gamma = \gamma^0$. Suppose $\tilde{G}_i$ has a maximum flow which has maximum capacity (i.e., $f_{rl} = c_{rl}$ for all $(v_r, v_l) \in \tilde{E}_i$), and $\frac{\partial \beta_{F_i}}{\partial \gamma}$ and $\frac{\partial s_k}{\partial \gamma}$ determined by (6) and (7) respectively. Then there exists certain positive number Δ such that for any $\gamma \in [\gamma^0, \gamma^0 + \Delta]$, the solution to the fused ontology problem is piecewise linear in γ which given by

$$\beta_k(\gamma) = \beta_k(\gamma^0) + \frac{\partial \beta_{F_i}}{\partial \gamma}(\gamma^0)\cdot(\gamma - \gamma^0) \text{ for } k \in F_i,$$

$$s_k(\gamma) = s_k(\gamma^0) + \frac{\partial s_{F_i}}{\partial \gamma}(\gamma^0)\cdot(\gamma - \gamma^0) \text{ for } k \in F_i,$$

$$\tau_{kl}(\gamma^0) = \begin{cases} \tau_{kl}(\gamma^0) + f_{kl}(\gamma - \gamma^0) & \text{for } k, l \in F_i \text{ for some } i \\ \mathrm{sign}(\tau_{kl}(\gamma^0))\gamma & \text{otherwise} \end{cases}.$$

Now, we introduce the positive number Δ. Let the hitting time of groups i and j at γ be

$$h_{ij}(\gamma)=\begin{cases}\dfrac{\beta_{F_i}-\beta_{F_j}}{\dfrac{\partial\beta_{F_i}}{\partial\gamma}-\dfrac{\partial\beta_{F_j}}{\partial\gamma}}+\gamma & \text{if } \exists k\in F_i, l\in F_j \text{ with } (v_k,v_l)\in E\\ \infty & \text{otherwise}\end{cases}.$$

The hitting time $h(\gamma)$ is then denoted by $h(\gamma)=\min_{h_{ij}>\gamma^0} h_{ij}(\gamma^0)$. For fixed flows f_{kl}, the violation time of the constraint on $\tau_{kl}(\gamma)$ is defined as

$$v_{kl}(\gamma^0)=\begin{cases}\dfrac{\left|\text{sign}(f_{kl})\gamma^0-\tau_{kl}(\gamma^0)\right|}{|f_{kl}|-1}+\gamma & \text{if } |f_{kl}|>1\\ \infty & \text{otherwise}\end{cases}.$$

Then, the violation time $v(\gamma)$ is defined by $v(\gamma^0)=\min v_{kl}(\gamma^0)$. Let

$$act_i(\gamma)=\begin{cases}\dfrac{1-s_{F_i}}{\dfrac{\partial s_{F_i}}{\partial\gamma}}+\gamma & \text{if } \dfrac{\partial s_{F_i}}{\partial\gamma}>0\\ \dfrac{1+s_{F_i}}{-\dfrac{\partial s_{F_i}}{\partial\gamma}}+\gamma & \text{if } \dfrac{\partial s_{F_i}}{\partial\gamma}<0\\ \infty & \text{otherwise}\end{cases}$$

be the activation time of collection F_i with $i\in N(\gamma)$. Then the activation time $\text{act}(\gamma)$ is denoted by $\text{act}(\gamma)=\min_{i\in N(\gamma)} act_i(\gamma)$. Let

$$d_i(\gamma)=\begin{cases}\dfrac{\beta_{F_i}}{-\dfrac{\partial\beta_{F_i}}{\partial\gamma}}+\gamma & \text{if } \dfrac{\partial\beta_{F_i}}{\partial\gamma}<0, \beta_{F_i}>0\\ \dfrac{-\beta_{F_i}}{\dfrac{\partial\beta_{F_i}}{\partial\gamma}}+\gamma & \text{if } \dfrac{\partial\beta_{F_i}}{\partial\gamma}>0, \beta_{F_i}<0\\ \infty & \text{otherwise}\end{cases}$$

be the deactivation time of the active collection F_i. Then the deactivation time $d(\gamma)$ is defined by $d(\gamma)=\min_{i\in A(\gamma)} d_i(\gamma)$.

Combining these together, Δ can be computed by

$$\Delta(\gamma)=\min\{h(\gamma),v(\gamma),act(\gamma),d(\gamma)\}-\gamma.$$

D. The Rule of Dividing and Fusing, and Description of the Algorithm

The last technology is to determine the rule on how to divide and fuse collections of variables as well as how to activate or inactivate a collection.

If there are collections F_i and F_j such that there are exist $k\in F_i$ and $l\in F_j$ with $(v_k,v_l)\in E$, $\beta_{F_i}=\beta_{F_j}$ and $s_{F_i}=s_{F_j}$, then these collections can be fused into a new collection $\tilde{F}_{ij}=F_i\cup F_j$ if $\dfrac{\partial s_{F_i}}{\partial\gamma}-\dfrac{\partial s_{F_j}}{\partial\gamma}\le 0$, $\dfrac{\partial\beta_{F_i}}{\partial\gamma}-\dfrac{\partial\beta_{F_j}}{\partial\gamma}\le 0$ and $t_{v_iv_j}=1$. If there is a collection F_i such that not all edges raising from the source vertex have maximal capacity in the associated maximal flow graph, then divide F_i in the two sub-collections R_i and s_i as presented above. The above iterate steps stop until nothing changes.

The rule for inactivating and activating the collections is given as follows: If $\dfrac{\partial s_{F_i}}{\partial\gamma}>0$ and $s_{F_i}=1$ or $\dfrac{\partial s_{F_i}}{\partial\gamma}<0$ and $s_{F_i}=-1$ for an inactive collection F_i, then we activate collection F_i; If $\beta_{F_i}=0$ and $s_{F_i}=1$, $\dfrac{\partial\beta_{F_i}}{\partial\gamma}<0$ or $s_{F_i}=-1$, $\dfrac{\partial\beta_{F_i}}{\partial\gamma}>0$ for an active collection F_i, then we deactivate the collection.

For our ontology problem, searching the starting value requires an extra step. For $\gamma=0$, our ontology model is

$$R_{\lambda,0}(\boldsymbol{\beta})=\frac{1}{2}(\boldsymbol{y}-\boldsymbol{V\beta})^{\mathrm{T}}(\boldsymbol{y}-\boldsymbol{V\beta})+\lambda\sum_{k=1}^{p}|\beta_k|.$$

Therefore, we can find the starting values for β by solving an ontology problem. For the starting value for *s*, note that the sub-gradient is

$$\frac{\partial R_{\lambda,\gamma}}{\partial\beta_k}=-(\boldsymbol{V}^{\mathrm{T}}\boldsymbol{y})_k+(\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V\beta})_k+\lambda s_k=0$$

and thus

$$s=\frac{1}{\lambda}(\boldsymbol{V}^{\mathrm{T}}\boldsymbol{y}+\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V\beta}).$$

Based on these starting values we now describe the complete algorithm which can be stated as follows.

Algorithm 1: Ontology Sparse Vector Learning via Fuse and Divide Technologies

Initialize: $\gamma=0$; $F_i=\{i\}$ for i=1,…, p; $p_F=p$; determine β_k and s_k for k=1,…, n.

If $p_F>1$, loop

Update β, *s* and *t*;

Determine the derivatives of β_{F_i} and s_{F_i} with respect to γ for i= 1,…, p_F;

Solve maximum flow problem for F_i, i=1,…, $p_F(\gamma)$;

Determine the next hitting time $h(\gamma)$; next violation time $v(\gamma)$; and the next time a collection will be activated $act(\gamma)$ or deactivated $d(\gamma)$

Calculate $\Delta(\gamma) = \min\{h(\gamma), v(\gamma), act(\gamma), d(\gamma)\} - \gamma$;

if $h_{ij}(\gamma) = \Delta(\gamma) + \gamma$ then fuse the two collections F_i and F_j; $p_F = p_F$ -1, $\gamma = h_{ij}(\gamma)$;

else if $v(\gamma) = \Delta(\gamma) + \gamma$ then

if $v(\gamma) = \gamma$ then divide the collection F_i into two smaller collections; $p_F = p_F + 1$; $\gamma = v(\gamma)$;

else if $a_i(\gamma) = \Delta(\gamma) + \gamma$ then activate the collection F_i and $\gamma = a_i(\gamma)$;

else if $d_i(\gamma) = \Delta(\gamma) + \gamma$ then deactivate the collection F_i; $\gamma = d_i(\gamma)$;

Output

IV. Simulation Studies

In this section, we designed four simulation experiments which are related to ontology similarity measure and ontology mapping. A vector with p dimension is used to express each vertex's information with the purpose of being close to the setting of ontology algorithm. The information of name, instance, attribute and structure of vertex is contained in the vector. Here the instance of vertex refers to the set of its reachable vertex in the directed (or, undirected) ontology graph.

In the following four experiments, for convenience, we set that the elements of matrix $\boldsymbol{V}$ are obtained by using a Gaussian distribution with mean 0, and $\boldsymbol{y}$ is determined by experts. After getting the sparse vector $\boldsymbol{\beta}$, the ontology function then is derived by $f_\beta(v) = \sum_{i=1}^{p} v_i \beta_i$ such that the noise term δ is ignored.

A. *Experiment on Biology Data*

"GO" ontology O_1 (which was constructed in http: //www. geneontology. org) is used for our experiment, and Fig. 1 shows the basic structure of O_1. *P@N* (Precision Ratio, see Craswell and Hawking [25] for more detail) is used to judge the equality of the experiment. At first, it is experts that give the closest *N* concepts for every vertex on the ontology graph. Then we get the first *N* concepts for every vertex on ontology graph by the algorithm and compute the precision ratio. Ontology algorithms in Huang et al., [26], Gao and Liang [27] and Gao and Gao [28] are applied into "Go" ontology. In the end, through the precision ratio which we have obtained from the four methods, and some experiment results can be referred to Tab. 1.

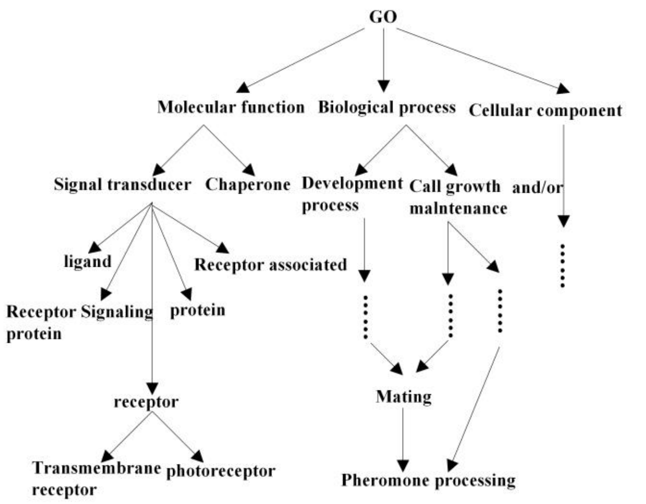

Fig. 1. "GO" ontology

While *N*= 3, 5, 10 or 20, the precision ratio which we get from our algorithm is higher than that determined by algorithms proposed in Huang et al., [26], Gao and Liang [27] and Gao and Gao [28]. Particularly, such precision ratios are increasing clearly as *N* increases. Thus, our algorithm is superior to the method presented by Huang et al., [26], Gao and Liang [27] and Gao and Gao [28].

B. *Experiment on Physical Education Data*

Physical education ontologies O_2 and O_3 (the structures of O_3 and O_3 are presented in Fig. 2 and Fig. 3 respectively) are chosen for our second experiment. This experiment tends to determine the ontology mapping between O_2 and O_3 by means of sparse ontology algorithm. *P@N* criterion works to measure the equality of the experiment. At first, it is experts that help to get the first closest *N* concepts for each vertex on the ontology graph. Then it is by the algorithm that we obtain the first *N* concepts for every vertex on ontology graph and compute the precision ratio. Then, we apply ontology algorithms in Huang et al., [26], Gao and Liang [27] and Gao et al., [11] to "physical education" ontology, too. In the end, by comparing the precision ratio that we have got from four methods, we get some experiment results which can be referred to Table 2.

The experiment results in Table 2 shows that , our algorithm turns out to be more efficient than those proposed in Huang et al., [26], Gao and Liang [27] and Gao et al., [11], especially in the situation where *N* is sufficiently large.

C. *Experiment on Plant Data*

In order to further evaluate the effectiveness of similarity measuring of our algorithm in special applications, we apply it to plant science. "PO" ontology O_4 was constructed in http: / www.plantontology.org) is used for our third experiment, and Fig. 4 manifests the general structure of O_4. Also, the *P@N* is used to measure the equality of the experiment data. Ontology learning algorithms presented in Gao and Zhu [17], Gao et al., [4], and Gao and Gao [29] are also applied into "PO" ontology. Finally, we compare the precision ratios which we have obtained from the four learning techniques in Table 3.

TABLE 1. THE EXPERIMENT DATA FOR ONTOLOGY SIMILARITY MEASURE

	P@3 average precision ratio	P@5 average precision ratio	P@10 average precision ratio	P@20 average precision ratio
Our Algorithm	47.82%	58.94%	68.10%	80.64%
Algorithm in [26]	46.38%	53.48%	62.34%	74.59%
Algorithm in [27]	43.56%	49.38%	56.47%	71.94%
Algorithm in [28]	42.13%	51.83%	60.19%	72.39%

TABLE 2. THE EXPERIMENT DATA FOR ONTOLOGY MAPPING

	P@1 average precision ratio	P@3 average precision ratio	P@5 average precision ratio
Our Algorithm	70.97%	80.65%	92.26%
Algorithm in [26]	61.29%	73.12%	79.35%
Algorithm in [27]	69.13%	75.56%	84.52%
Algorithm in [11]	67.74%	77.42%	89.68%

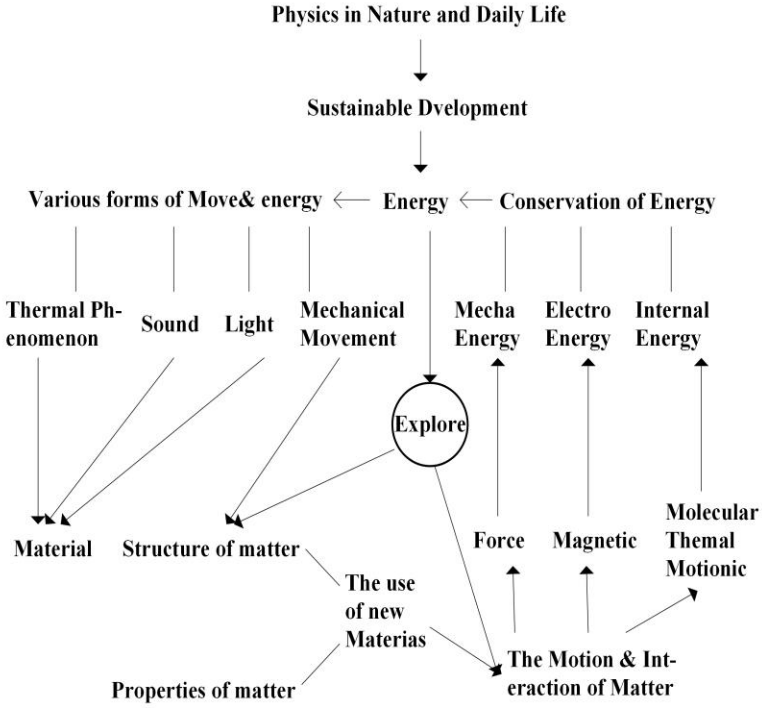

Fig. 2. “Physical Education” Ontology O_2

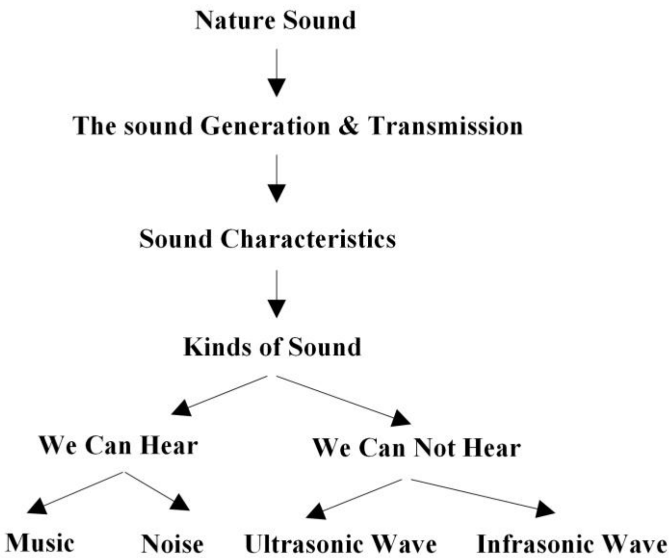

Fig. 3. “Physical Education” Ontology O_3

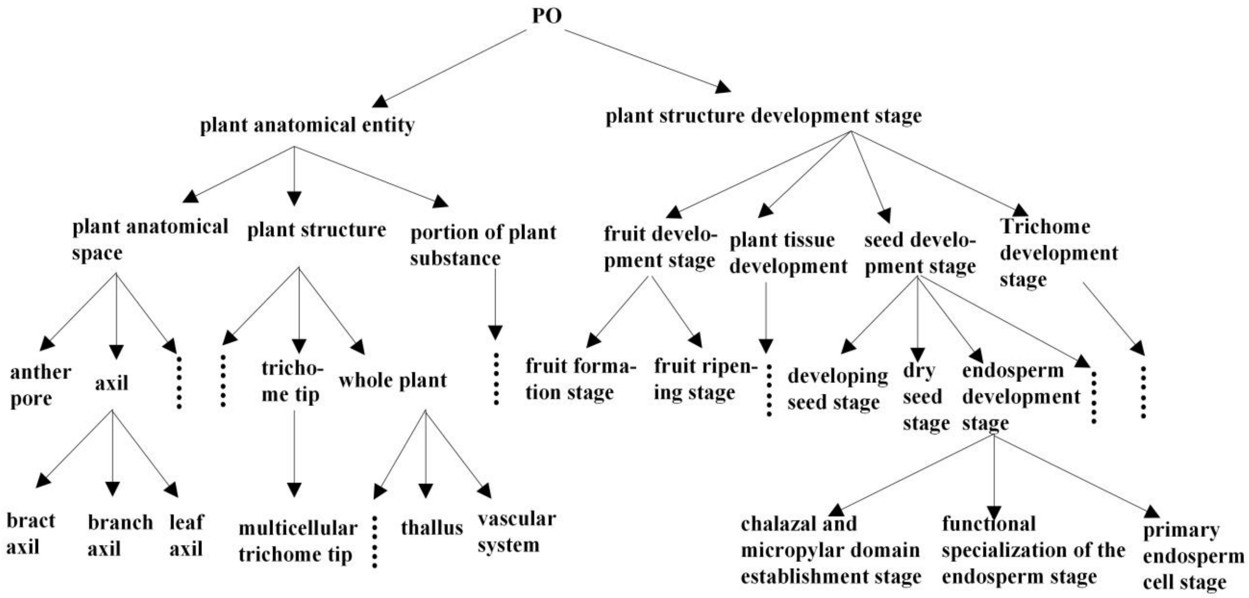

Fig. 4. "PO" Ontology O_4

TABLE 3. THE EXPERIMENT DATA ON PLANT ONTOLOGY

	P@3 average precision ratio	P@5 average precision ratio	P@10 average precision ratio
Our Algorithm	51.95%	63.67%	86.03%
Algorithm in [17]	50.42%	62.16%	78.53%
Algorithm in [4]	51.03%	61.76%	79.04%
Algorithm in [29]	49.04%	57.85%	71.01%

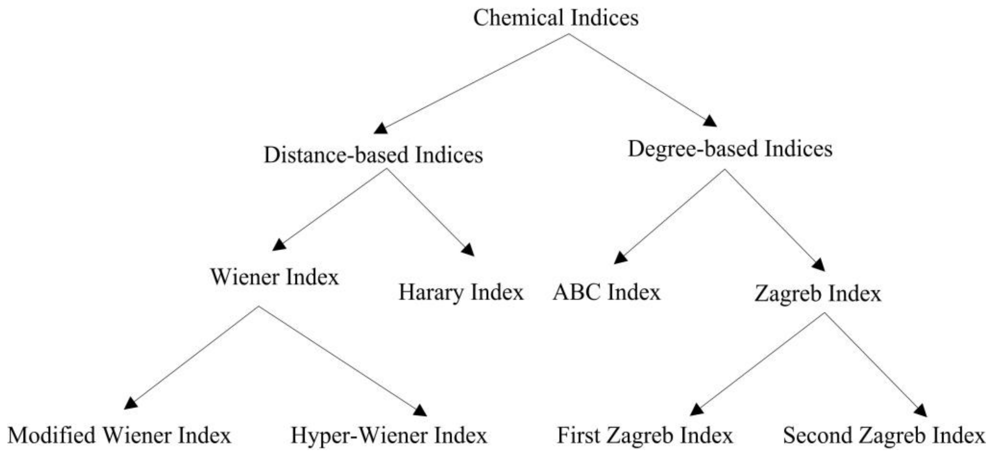

Fig.5. "Chemical Index" Ontology O_5

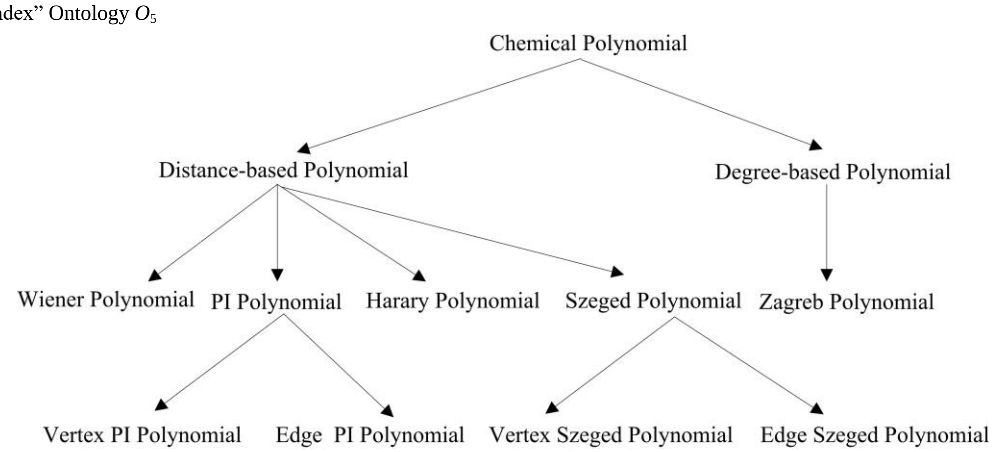

Fig. 6. "Chemical Index" Ontology O_6

TABLE 4. THE EXPERIMENT DATA ON CHEMICAL INDEX DATA

	P@1 average precision ratio	P@3 average precision ratio	P@5 average precision ratio
Our Algorithm	39.47%	52.92%	69.65%
Algorithm in [17]	32.47%	44.15%	56.67%
Algorithm in [4]	35.09%	45.32%	60.35%
Algorithm in [29]	39.47%	49.42%	61.75%

The experiment comparing data in Table 3 reveals that our algorithm turns out to be more efficient than those proposed in Gao and Zhu [17], Gao et al., [4], and Gao and Gao [29] for N=3, 5 and 10. The trends of accuracy of the algorithm efficiency implies that our learning algorithm has more efficiency than other three learning tricks if N is large enough.

D. Experiment on Chemical Index Data

To deep test the effectiveness of ontology mapping of our algorithm in special engineering applications, we finally apply our algorithm on the chemical index data. Our aim is to build the similarity based ontology mapping between two chemical index ontology O_5 and O_6 which are constructed by Wu et al., [30]. Fig. 5 and Fig. 6 present the general structure of O_5 and O_6. We emphasize here that figures 5 and 6 presented only consist of partial vertices of ontologies O_5 and O_6. In fact, chemical index ontology O_5 contains 68 concepts, and the chemical index ontology O_6 contains 46 concepts. Again, the $P@N$ is used to measure the equality of the experiment data. Ontology learning algorithms presented in Gao and Zhu [17], Gao et al., [4], and Gao and Gao [29] are also applied into chemical index ontologies. Finally, we compare the precision ratios which we have obtained from the four learning techniques in Tab. 4.

The experiment comparing results presented in Table 4 implies that our sparse ontology learning algorithm has more efficiency than those proposed in Gao and Zhu [17], Gao et al., [4], and Gao and Gao [29]. Our algorithm is suitable to constructing the ontology mapping for chemical index ontologies.

V. Conclusions

In this paper, the fuse and divide technologies are presented for ontology sparse vector computation. The new iterative computation algorithm is based on these ontology technologies and sub-gradient computation. At last, simulation data shows that our new algorithm has high efficiency in biology, physical education, plant science and chemical index ontologies. The ontology sparse algorithm raised in our paper illustrates the promising application prospects for multiple disciplines.

Acknowledgment

We thank the reviewers for their constructive comments in improving the quality of this paper.

References

[1] E. Raad and J. Evermann, “The Role of Analogy in Ontology Alignment: A Study on LISA,” Cognitive Systems Research, vol. 33, pp. 1-16, 2015.

[2] F. Ali, E. K. Kim, and Y. G. Kim, “Type-2 Fuzzy Ontology-based Semantic Knowledge for Collision Avoidance of Autonomous Underwater Vehicles,” Information Sciences, vol. 295, pp. 441-464, 2015.

[3] W. Gao and L. Shi, “Szeged Related Indices of Unilateral Polyomino Chain and Unilateral Hexagonal Chain,” IAENG International Journal of Applied Mathematics, vol. 45, no.2, pp. 138-150, 2015.

[4] W. Gao, L. L. Zhu and Y. Guo, “Multi-dividing Infinite Push Ontology Algorithm,” Engineering Letters, vol. 23, no. 3, pp. 132-139, 2015.

[5] V. Gopal and N. S. Gowri Ganesh, “Ontology Based Search Engine Enhancer,” IAENG International Journal of Computer Science, vol. 35, no. 3, pp. 413-420, 2008.

[6] B. Amini, R. Ibrahim, M. S. Othman, and M. A. Nematbakhsh, “A Reference Ontology for Profiling Scholar's Background Knowledge in Recommender Systems,” Expert Systems with Applications, vol. 42, pp. 913-928, 2015.

[7] S. T. Sun, L. Z. Wang, R. Ranjan, and A. Z. Wu, “Semantic Analysis and Retrieval of Spatial Data Based on the Uncertain Ontology Model in Digital Earth,” International Journal of Digital Earth, vol. 8, no. 1, pp. 1-14, 2015.

[8] X. Q. Zhong, H. G. Fu, H. D. Xia, L. N. Yang, and M. S. Shang, “A Hybrid Cognitive Assessment Based on Ontology Knowledge Map and Skills,” Knowledge-Based Systems, vol. 73, pp. 52-60, 2015.

[9] M. Mahfoudh, G. Forestier, L. Thiry, and M. Hassenforder, “Algebraic Graph Transformations for Formalizing Ontology Changes and Evolving,” Knowledge-Based Systems, vol. 73, pp. 212-226, 2015.

[10] W. Gao and L. Liang, “Sparse Vector Learning in Ontology Algorithms and Applied to Multiple Disciplines,” Wulfenia, vol. 22, no. 1, pp. 420-435, 2015.

[11] W. Gao, Y. Gao, and L. Liang, “Diffusion and Harmonic Analysis on Hypergraph and Application in Ontology Similarity Measure and Ontology Mapping,” Journal of Chemical and Pharmaceutical Research, vol. 5, no. 9, pp. 592-598, 2013.

[12] W. Gao and L. Shi, “Ontology Similarity Measure Algorithm with Operational Cost and Application in Biology Science,” BioTechnology: An Indian Journal, vol. 8, no. 11, pp. 1572-1577, 2013.

[13] W. Gao and T. W. Xu, “Ontology Similarity Measuring and Ontology Mapping Algorithm Based on MEE Criterion,” Energy Education Science and Technology Part A: Energy Science and Research, vol. 32, no. 5, pp. 3793-3806, 2014.

[14] J. Z. Wu, X. Yu, and W. Gao, “Bilinear Model for Ontology Mapping,” Computer Modelling & New Technologies, vol. 18, no. 8, pp. 368-372, 2014.

[15] W. Gao, Y. Gao, and Y. Zhang, “Strong and Weak Stability of *k*-Partite Ranking Algorithm,” Information, vol. 15, no. 11A, pp. 4585-4590, 2012.

[16] W. Gao and T. W. Xu, “Stability Analysis of Learning Algorithms for Ontology Similarity Computation,” Abstract and Applied Analysis, 2013, 9 pages, http://dx.doi.org/10.1155/2013/174802.

[17] W. Gao and L. Zhu, “Gradient Learning Algorithms for Ontology Computing,” Computational Intelligence and Neuroscience, 2014, 12 pages, http://dx.doi.org/10.1155/2014/438291.

[18] W. Gao, L. Yan, and 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, vol. 50, no. 3/4, pp. 226-231, 2014.

[19] G. Mancinelli, S. Vizzini, A. Mazzola, S. Maci, and A. Basset, “Cross-validation of Delta N-15 and FishBase Estimates of Fish Trophic Position in A Mediterranean Lagoon: The Importance of the Isotopic Baseline,” Estuaring Coastal and Shelf Science, vol. 135, pp. 77-85, 2013.

[20] Z. Zhu, P. Chen, and J. Zhuang, “Predicting Chinese Children and Youth's Energy Expenditure Using ActiGraph Accelerometers: A Calibration and Cross-validation Study,” Research Quarterly for Exercise and Sport, vol. 84, no. 2, pp. S56-S63, 2013.

[21] S. Mukhopadhyay and S. Bhattacharya, “Cross-validation Based Assessment of A New Bayesian Palaeoclimate Model,” Environmetrics, vol. 24, no. 8, pp. 550-568, 2013.

[22] H. Ishibuchi and Y. Nojima, “Repeated Double Cross-validation for Choosing A Single Solution in Evolutionary Multi-objective Fuzzy Classifier Design,” Knowledge-based Systems, vol. 54. pp. 22-31, 2013.

[23] Y. Zhang, X. Yu, D. Guo, Y. Yin, and Z. Zhang, “Weights and Structure Determination of Multiple-input Feed-forward Neural Network Activated by Chebyshev Polynomials of Class 2 via Cross-validation,” Neural Computing & Applications, vol. 25, pp. 1761-1770, 2014.

[24] K. Varmuza, P. Filzmoser, M. Hilchenbach, H. Krüger, and J. Silén, “KNN Classification-evaluated by Repeated Double Cross Validation: Recognition of Minerals Relevant for Comet Dust,” Chemometrics and Intelligent Laboratory System, vol. 138, pp. 64-71, 2014.

[25] N. Craswell and D. Hawking, “Overview of the TREC 2003 Web Track,” Proceeding of the Twelfth Text Retrieval Conference, Gaithersburg, Maryland, NIST Special Publication, 2003, pp. 78-92.

[26] X. Huang, T. Xu, W. Gao, and Z. Jia, “Ontology Similarity Measure and Ontology Mapping via Fast Ranking Method,” International Journal of Applied Physics and Mathematics, vol. 1, pp. 54-59, 2011.

[27] W. Gao and L. Liang, “Ontology Similarity Measure by Optimizing NDCG Measure and Application in Physics Education,” Future

Communication, Computing, Control and Management, vol. 142, pp. 415-421, 2011.
[28] Y. Gao and W. Gao, "Ontology Similarity Measure and Ontology Mapping via Learning Optimization Similarity Function," International Journal of Machine Learning and Computing, vol. 2, no. 2, pp. 107-112, 2012.
[29] Y. Gao and W. Gao, "Ontology Sparse Vector Learning Based on Accelerated First-Order Method," The Open Cybernetics & Systemics Journal, vol. 9, pp. 657-662, 2015.
[30] J. Z. Wu, X. Yu, W. Gao, "Disequilibrium Multi-dividing Ontology Learning Algorithm", Manuscript.

Xiao Yu, male, was born in the city of Nanjing, Jiangsu Province, China on Feb. 3, 1973. She got a bachelor degree of Engineering on Computer Software from Southeast University in 1995. Then, she got Master degree of Engineering on Computer Organism and Architecture from Southeast University in 2002.

She acted as lecturer in the department of computer science and engineering, Southeast University from July 1995 to November 2000. Now, she acts as lecturer in the department of distance education college, Southeast University. As a researcher in computer science and mathematics, her interests are covering two disciplines: System Management, WEB Technology, Information retrieval, and Cloud Computing.

Jianzhang Wu, male, was born in the city of Wanzhou, Sichuan Province, China on Oct. 24, 1972. He got two bachelor degrees on control science and engineering from Southeast University in 1995 and omputer science from Southeast University in 2002.

He acted as lecturer in the department of Computer Science and Engineering, Southeast University from July 1995. As a researcher in computer science and mathematics, his interests are covering three disciplines: Statistical learning theory, Cloud computing, and Artificial Intelligence.

Wei Gao, male, was born in the city of Shaoxing, Zhejiang Province, China on Feb.13, 1981. He got two bachelor degrees on computer science from Zhejiang industrial university in 2004 and mathematics education from College of Zhejiang education in 2006. Then, he was enrolled in department of computer science and information technology, Yunnan normal university, and got Master degree there in 2009. In 2012, he got PhD degree in department of Mathematics, Soochow University, China.

He acted as lecturer in the department of information, Yunnan Normal University from July 2012 to December 2015. Now, he acts as associate professor in the department of information, Yunnan Normal University. As a researcher in computer science and mathematics, his interests are covering two disciplines: Graph theory, Statistical learning theory, Information retrieval, and Artificial Intelligence.