Multi-dividing Infinite Push Ontology Algorithm

Wei Gao, Linli Zhu, and Yun Guo

***Abstract*—Along with the arrival of the era of large amount of data, many machine learning methods have been applied to the ontology similarity calculation and ontology mapping. In this paper, we raise an infinite push model for ontology similarity measuring and ontology mapping in multi-dividing setting. The iterative algorithm and generalization bound are given by virtue of the dual solution for optimization and the trick of covering number approach. Furthermore, the fast ontology algorithm for standard ontology SVM by virtue of infinite push multi-dividing ontology algorithm is obtained. Finally, four experiments presented on various fields verify the efficiency of the new computation model for ontology similarity measuring and ontology mapping applications in multi-dividing setting.**

***Index Terms*—ontology, infinite push, similarity measure, ontology mapping, multi-dividing setting.**

I. Introduction

THE term "ontology" is originally from the field of philosophy and it is used to describe the natural connection of things and the inherent hidden connections of their components. In information and computer science, ontology is a model for knowledge storing and representation, and has been widely applied in knowledge management, machine learning, information systems, image retrieval, information retrieval search extension, collaboration and intelligent information integration. In the past decade, as an effective concept semantic model and a powerful analysis tool, ontology has been widely applied in pharmacology science, biology science, medical science, geographic information system and social sciences (for instance, see Przydzial et al., [1], Koehler et al., [2], Ivanovic and Budimac [3], Hristoskova et al., [4], and Kabir [5]).

The structure of ontology can be expressed as a simple graph. Each concept, object or element in ontology corresponds to a vertex and each (directed or undirected) edge on an ontology graph represents a relationship (or potential link) between two concepts (objects or elements). Let O be an ontology and G be a simple graph correspond to G. The nature of ontology engineer application can be attributed to getting the similarity calculating function which is to compute the similarities between ontology vertices. These similarities represent the intrinsic link between vertices in ontology graph. The goal of ontology mapping is to get the ontology similarity measuring function by measuring the similarity between vertices from different ontologies. Such mapping is a bridge between different ontologies, and get a potential association between the objects or elements from different ontologies. Specifically, the ontology similarity function $Sim : V \times V \to \mathbb{R}^+ \cup \{0\}$ is a semi-positive score function which maps each pair of vertices to a non-negative real number.

Very recently, ontology technologies have been employed in a variety of applications. Ma et al., [6] presented a graph derivation representation based technology for stable semantic measurement. Li et al., [7] raised an ontology representation method for online shopping customers knowledge in enterprise information. Santodomingo et al., [8] proposed an innovative ontology matching system that finds complex correspondences by processing expert knowledge from external domain ontologies and in terms of using novel matching tricks. Pizzuti et al., [9] described the main features of the food ontology and some examples of application for traceability purposes. Lasierra et al., [10] argued that ontologies can be used in designing an architecture for monitoring patients at home. More ontology applications on various engineering can refer to [11], [12] and [13].

The advanced idea to deal with the ontology similarity computation is using ontology learning algorithm which gets a ontology function $f : V \to \mathbb{R}$. By virtue of the ontology function, the ontology graph is mapped into a line which consists of real numbers. The similarity between two concepts then can be measured by comparing the difference between their corresponding real numbers. The essence of this idea is dimensionality reduction. In order to associate the ontology function with ontology application, for vertex v, we use a vector to express all its information. In order to facilitate the representation, we slightly confuse the notations and use v to denote both the ontology vertex and its corresponding vector. The vector is mapped to a real number by ontology function $f : V \to \mathbb{R}$, and the ontology function is a dimensionality reduction operator which maps multi-dimensional vectors into one dimensional vectors.

There are several effective methods of getting efficient ontology similarity measure or ontology mapping algorithm in terms of ontology function. Wang et al., [14] considered the ontology similarity calculation in terms of ranking learning technology. Huang et al., [15] raised the fast ontology algorithm in order to cut the time complexity for ontology application. Gao and Liang [16] presented a ontology optimizing model such that the ontology function is determined by virtue of NDCG measure, and it is successfully applied in physics education. Since the large parts of ontology structure is the tree, researchers explored the learning theory approach for ontology similarity calculating and ontology mapping in specific setting when the structure of ontology graph has no cycle. In the multi-dividing ontology setting, all vertices in ontology graph or multi-ontology graph are divided into k parts corresponding to the k classes of rates. The rate

Manuscript received March 31, 2015; revised May 12, 2015. This work was supported in part by the Key Laboratory of Educational Informatization for Nationalities, Ministry of Education, the National Natural Science Foundation of China (60903131), the College Natural Science Foundation of Jiangsu Province in China (10KJD520002) and the PhD initial funding of the first author.

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

L. L. Zhu is with School of Computer Engineering, Jiangsu University of Technology, Changzhou, 213001, China, e-mail: zhulinli@jsut.edu.cn.

Y. Guo is with School of Computer Science and Engineering, Soochow University, Suzhou 215006, China, e-mail: guoyun@suda.edu.cn.

values of all classes are determined by experts. In this way, a vertex in a rate a has larger score than any vertex in rate b (if $1 \leq a < b \leq k$) under the multi-dividing ontology function $f: V \to \mathbb{R}$. Finally, the similarity between two ontology vertices corresponding to two concepts (or elements) is judged by the difference of two real numbers which they correspond to. Hence, the multi-dividing ontology setting is suitable to get a score ontology function for an ontology application if the ontology is drawn into a non-cycle structure.

In this article, we raise a new multi-dividing ontology learning algorithm for ontology similarity measuring and ontology mapping by means of infinite push. The rest of the paper is arranged as follows: in Section 2, the detailed description of infinite push multi-dividing ontology algorithms is showed, and the generalization bound is also yielded via covering number trick; in Section 3, we obtain the fast algorithm for ontology SVM training based on infinite push multi-dividing algorithm; in Section 4, four respective simulation experiments on plant science, humanoid robotics, biology and physics education are designed to test the efficiency of our infinite push based ontology algorithm, and the data results reveal that our algorithm has high precision ratio for these applications.

II. The Multi-dividing Infinite Push Ontology Algorithm

Let $V \subset \mathbb{R}^d$ ($d \geq 1$) be a vertex space (or the instance space) for ontology graph, and the vertices (or, instances) in V are drawn randomly and independently according to some (unknown) distribution $\mathcal{D}$. Given a training set $S = \{v_1, \cdots, v_n\}$ of size n in V, the goal of ontology learning algorithms is to obtain a score function $f: V \to \mathbb{R}$, which assigns a score to each vertex, and ranks all the instances according to their scores. The multi-dividing ontology problem is a special kind of ontology learning problem in which vertices come from k categories and the learner is given examples of vertices labeled as there k rates.

Formally, the settings of multi-dividing ontology problems can be described as follows. There is an instance space V from which vertices are drawn, and the learner is given a training sample $(S_1, S_2, \cdots, S_k) \in V^{n_1} \times V^{n_2} \times \cdots \times V^{n_k}$ consisting of a sequence of training sample $S_a = (v_1^a, \cdots, v_{n_a}^a)$ ($1 \leq a \leq k$). The goal is to learn from these samples a real-valued ontology score function $f: V \to \mathbb{R}$ that orders the future S_a vertices rank higher than S_b where $a < b$. Hence, the empirical model for measuring the quality of ontology function f can be expressed as

$$R(f;S) = \sum_{a=1}^{k-1} \sum_{b=a+1}^{k} \max_{1 \leq j \leq n_b} \left(\frac{1}{n_a} \sum_{i=1}^{n_a} \mathrm{I}(f(v_i^a) < f(v_j^b))\right), \tag{1}$$

where $\mathrm{I}(\cdot) = 1$ if the argument is held and 0 otherwise. The basic idea of (1) is to search an ontology function that guarantees the accuracy of top vertices for each rate a ($1 \leq a \leq k$). Such technology (concern the top vertices for each rate of the goal list) is called infinite push.

Since the function $\mathrm{I}(\cdot)$ is non-differentiable, we should minimize (1) by using a continuous convex function instead of I. Specially, we consider the hinge ontology loss, which for f, v^a, v^b is denoted as

$$l_H(f, v^a, v^b) = (1 - f(v^a - f^b))_+, \tag{2}$$

where $u_+ = \max(u, 0)$. Hence, we shall minimize a regularized version of

$$R^H(f;S) = \sum_{a=1}^{k-1} \sum_{b=a+1}^{k} \max_{1 \leq j \leq n_b} \left(\frac{1}{n_a} \sum_{i=1}^{n_a} l_H(f(v_i^a) < f(v_j^b))\right), \tag{3}$$

Let $\mathcal{F}$ be a real-valued ontology function space on V associated with a certain reproducing kernel. Let $\|f\|_{\mathcal{F}}$ be the RKHS norm of f in $\mathcal{F}$ and the positive number λ be a regularization parameter. Then the optimization problem can be stated as:

$$\min_{f \in \mathcal{F}} [R^H(f;S) + \frac{\lambda}{2}\|f\|_{\mathcal{F}}^2]. \tag{4}$$

In particular, we consider $V = \mathbb{R}^d$ and $\mathcal{F}$ as the class of ontology functions $f: \mathbb{R}^d \to \mathbb{R}$ with $f(v) = w \cdot v$ for some $w \in \mathbb{R}^d$. By using $\|w\|$ as the Euclidean l_2 norm of w and employing w instead of f in $R^H(w;S)$, we infer that (3) becomes

$$\min_{w \in \mathbb{R}^d} [R^H(w;S) + \frac{\lambda}{2}\|w\|_{\mathcal{F}}^2]. \tag{5}$$

Conjunction with the definition of $R^H(w;S)$, we deduce

$$\min_{w \in \mathbb{R}^d} [\frac{\lambda}{2}\|f\|_{\mathcal{F}}^2 + \sum_{a=1}^{k-1} \sum_{b=a+1}^{k} \max_{1 \leq j \leq n_b} (\frac{1}{n_a} \sum_{i=1}^{n_a} (1 - w \cdot (v_i^a - v_j^b)))]. \tag{6}$$

For each pair of (a, b) ($1 \leq a < b \leq k$), let $\xi_{ij}^{a,b}$ ($1 \leq i \leq n_a, 1 \leq j \leq n_b$) be the slack variables corresponding to the max in the hinge loss terms $(1 - w \cdot (v_i^a - v_j^b))_+ = \max((1 - w \cdot (v_i^a - v_j^b)), 0)$. Denote $C = \frac{1}{\lambda}$, then (6) can be re-written as

$$\min_{w, \xi_{ij}^{a,b}} [\frac{\lambda}{2}\|f\|_{\mathcal{F}}^2 + C \sum_{a=1}^{k-1} \sum_{b=a+1}^{k} \max_{1 \leq j \leq n_b} (\frac{1}{n_a} \sum_{i=1}^{n_a} \xi_{ij}^{a,b})]$$

$$\text{subject to} \quad \xi_{ij}^{a,b} \geq 1 - w \cdot (v_i^a - v_j^b) \quad \forall i, j, (a,b)$$

$$\xi_{ij}^{a,b} \geq 0 \quad \forall i, j, (a,b).$$

Moreover, by introducing further slack variables $\xi^{a,b}$ corresponding to this max for each pair of (a, b), the above optimization model is transformed as

$$\min_{w, \xi^{a,b}, \xi_{ij}^{a,b}} [\frac{\lambda}{2}\|f\|_{\mathcal{F}}^2 + \sum_{a=1}^{k-1} \sum_{b=a+1}^{k} \frac{C}{n_a} \xi^{a,b}]$$

$$\text{subject to} \quad \xi^{a,b} \geq \sum_{i=1}^{n_a} \xi_{ij}^{a,b}, \quad \text{for} \quad j = 1, \cdots, n_b$$

$$\xi_{ij}^{a,b} \geq 1 - w \cdot (v_i^a - v_j^b) \quad \forall i, j, (a,b)$$

$$\xi_{ij}^{a,b} \geq 0 \quad \forall i, j, (a,b). \tag{7}$$

In terms of Lagrange multipliers and the trick of dual, the optimization problem can be further re-written by variables $\{\alpha_{ij}^{a,b} : 1 \leq n_a, 1 \leq j \leq n_b\}$ for each pair of (a, b):

$$\min_{\alpha_{ij}^{a,b}} [\frac{1}{2} \sum_{a=1}^{k-1} \sum_{b=a+1}^{k} \sum_{i,j} \sum_{k,l} \alpha_{ij}^{a,b} \alpha_{kl}^{a,b} (v_i^a - v_j^b)$$

$$-\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{i,j}\alpha_{ij}]$$

$$\text{subject to} \quad \sum_{j=1}^{n_b}(\max_{1\le i\le n_a}\alpha_{ij}) \le \frac{C}{n_a}$$

$$\alpha_{ij}^{a,b} \ge 0 \qquad \forall i,j,(a,b). \tag{8}$$

Let $Q^{a,b}(\alpha)$ be the (quadratic) objective function in (8), and let $\Omega^{a,b} \subseteq \mathbb{R}^{mn}$ be the feasible set for each pair of (a,b), i.e., the set of $\alpha^{a,b} \in \mathbb{R}^{n_a n_b}$ satisfying the constraints in (8). Then the projection algorithm starts with some initial value $(\alpha^{a,b})^{(1)}$ for $\alpha^{a,b}$, and on each iteration t for special pair (a,b), updates $(\alpha^{a,b})^{(t)}$ using a gradient and projection step:

$$(\alpha^{a,b})^{(t+1)} \leftarrow \mathcal{P}_{\Omega^{a,b}}((\alpha^{a,b})^{(t)} - \eta_t^{a,b}\nabla Q^{a,b}((\alpha^{a,b})^{(t)})),$$

where $\eta_t^{a,b} > 0$ denotes a learning rate, $\nabla Q^{a,b}$ denotes the gradient of $Q^{a,b}$, and $\mathcal{P}_{\Omega^{a,b}}$ denotes Euclidean projection onto $\Omega^{a,b}$. In the linear case, this gradient computation can be completed in $O(d\sum_{a=1}^{k-1}\sum_{b=a+1}^{k} n_a n_b)$ time. Moreover, it takes $O(\sum_{a=1}^{k-1}\sum_{b=a+1}^{k} n_a n_b \log(n_a n_b))$ time in the projection step. On solving $\alpha^{a,b}$, the solution of (5) can be stated as

$$w = \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{i,j}\alpha_{ij}^{a,b}(v_i^a - v_j^b).$$

Returning to the general circumstances, V is any vertex space and $\mathcal{F}$ is an RKHS corresponding to a kernel function $K: V\times V \to \mathbb{R}$. Denote $\phi_K(x,y,z,u) = K(x,z) - K_{x,u} - K(y,z) + K(y,u)$. By view of similar derivation, we derive the following kernel version of optimization problem:

$$\min_{\alpha_{ij}^{a,b}}[\frac{1}{2}\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{i,j}\sum_{k,l}\alpha_{ij}\alpha_{kl}\phi_K(v_i^a, v_j^b, v_k^a, v_l^b)$$

$$-\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{i,j}\alpha_{ij}]$$

$$\text{subject to} \quad \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{j=1}^{n_b}(\max_{1\le i\le n_a}\alpha_{ij}) \le \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\frac{C}{n_a}$$

$$\alpha_{ij}^{a,b} \ge 0 \qquad \forall i,j,(a,b). \tag{9}$$

On solving $\alpha^{a,b}$ for each pair of (a,b) $(1 \le a < b \le k)$, the solution to the original multi-dividing ontology problem is then obtained by

$$f(v) = \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{i,j}\alpha_{ij}^{a,b}(K(v_i^a, v) - K(v_j^b, v)). \tag{10}$$

Algorithm 1. Multi-dividing infinite push ontology algorithm:
Input: training sample $S = (S_1, S_2, \cdots, S_k) \in V^{n_1}\times V^{n_2}\times\cdots\times V^{n_k}$ and a kernel function $K: V\times V\to\mathbb{R}$
For $a = 1$ to $k-1$:
For $b = a+1$ to k:
Parameters $C^{a,b}, t_{max}^{a,b}, \eta_0^{a,b}$
Initialize: $\alpha_{ij}^{(1)} \leftarrow \frac{C^{a,b}}{n_a n_b} \quad \forall 1\le i\le n_a, 1\le j\le n_b$
For $t = 1$ to $t_{max}^{a,b}$ do: $(\alpha^{a,b})^{\frac{t+1}{2}} \leftarrow (\alpha^{a,b})^{(t)} - \frac{\eta_0^{a,b}}{\sqrt{t}}\nabla Q^{a,b}((\alpha^{a,b})^{(t)})$, $(\alpha^{a,b})^{(t+1)} \leftarrow \mathcal{P}_{\Omega^{a,b}}((\alpha^{a,b})^{(\frac{t+1}{2})})$
End For.
End For.
End For.
Output: $f(v) = \sum_{i,j}\alpha_{ij}^{(t^{a,b})^*}(K(v_i^a, v) - K(v_j^b, v))$, where $(t^{a,b})^* = \text{argmin}_{1\le t\le(t_{max}^{a,b}+1)} Q((\alpha^{a,b})^{(t)})$.

We emphasize here that the p-norm multi-dividing push ontology algorithm minimizes a convex upper bound on the l_p-norm similar as the quantity in (1) for finite p:

$$R_p(f,S) = (\frac{1}{n_b}\sum_{j=1}^{n_b}(\frac{1}{n_a}\sum_{i=1}^{n_a}\Pi(f(v_i^a) < f(v_j^b)))^p)^{\frac{1}{p}}. \tag{11}$$

In the following part of the section, we give some theoretical analysis for Algorithm 1. Assume vertices in rate a (here, $1 \le a \le k$) are drawn randomly and independently according to a distribution $\mathcal{D}_a$ on V. Let $\mathcal{D} = (\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_k)$, and define

$$R(f;\mathcal{D}) = \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sup_{v\in\text{supp}(\mathcal{D}_b)}(\mathbb{E}_{v^a\sim\mathcal{D}_a}[\text{I}(f(v_i^a) - f(v_j^b))]), \tag{12}$$

and

$$R^{a,b}(f;\mathcal{D}) = \sup_{v\in\text{supp}(\mathcal{D}_b)}(\mathbb{E}_{v^a\sim\mathcal{D}_a}[\text{I}(f(v_i^a) - f(v_j^b))]), \tag{13}$$

where $\text{supp}(\mathcal{D}_b)$ denotes the support of $\mathcal{D}_b$.

For any $\gamma > 0$, define the margin ontology loss as

$$l_\gamma(f, v_i^a, v_j^b) = \text{I}(f(v_i^a) - f(v_j^b) \le \gamma),$$

and some notations are defined as follows:

$$R^\gamma(f;S) = \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\max_{1\le j\le n_b}(\frac{1}{n_a}\sum_{i=1}^{n_a} l_\gamma(f, v_i^a, v_j^b)), \tag{14}$$

$$R^\gamma(f;\mathcal{D}) = \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sup_{v\in\text{supp}(\mathcal{D}_b)}(\mathbb{E}_{v^a\sim\mathcal{D}_a}[l_\gamma(f(v_i^a) - f(v_j^b))]), \tag{15}$$

$$R^{\gamma,a,b}(f;S) = \max_{1\le j\le n_b}(\frac{1}{n_a}\sum_{i=1}^{n_a} l_\gamma(f, v_i^a, v_j^b)), \tag{16}$$

$$R^{\gamma,a,b}(f;\mathcal{D}) = \sup_{v\in\text{supp}(\mathcal{D}_b)}(\mathbb{E}_{v^a\sim\mathcal{D}_a}[l_\gamma(f(v_i^a) - f(v_j^b))]), \tag{17}$$

$$l_\gamma(f, \mathcal{D}_a, v^b) = \mathbb{E}_{v^a\sim\mathcal{D}_a}[l_\gamma(f(v_i^a) - f(v_j^b))]. \tag{18}$$

Let $\mathcal{N}(\mathcal{F}, \varepsilon)$ be the l_∞ covering number of $\mathcal{F}$ for radius ε, i.e., the minimum number of l_∞ balls with radius ε cover $\mathcal{F}$. The generalization bound for the multi-dividing infinite push ontology algorithm is stated in the following conclusion.

Theorem. Let $\mathcal{F}$ be a class of real-valued ontology functions on V. Let $\varepsilon, \gamma > 0$, and

$$\delta_{\mathcal{D}}^{a,b}(f,\gamma,\varepsilon) = \mathbb{P}_{v^b\sim\mathcal{D}_b}[l_\gamma(f,\mathcal{D}_a, v^b) < R^\gamma(f;\mathcal{D}) - \varepsilon].$$

Then, we have

$$\begin{aligned}&\mathbb{P}_{S\sim\mathcal{D}_1^{n_1}\times\mathcal{D}_2^{n_2}\times\cdots\times\mathcal{D}_k^{n_k}}[\exists f\in\mathcal{F}: R^\gamma(f;\mathcal{D}) \\ &> R^\gamma(f;S) + \varepsilon] \\ \le\ & \mathcal{N}(\mathcal{F}, \frac{\varepsilon\gamma}{8})\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}(2n_b e^{-\frac{n_a\varepsilon^2}{8}} \\ &+(\sup_{f\in\mathcal{F}}\delta_{\mathcal{D}}^{a,b}(f,\gamma,\varepsilon/4))^{n_b}).\end{aligned}$$

Proof. The most important step in the proof is to bound the following probability

$$\mathbb{P}_{S\sim\mathcal{D}_1^{n_1}\times\mathcal{D}_2^{n_2}\times\cdots\times\mathcal{D}_k^{n_k}}[R^{\gamma}(f;\mathcal{D})>R^{\gamma}(f;S)+\frac{\varepsilon}{2}]$$

for given $f\in\mathcal{F}$. On this purpose, we define

$$R^{\gamma,a,b}(f;\mathcal{D}_a,S_b)=\max_{1\le j\le n_b}l_{\gamma}(f,\mathcal{D}_a,v_j^b).$$

Then, we infer

$$\begin{aligned}
&\mathbb{P}_{S\sim\mathcal{D}_1^{n_1}\times\mathcal{D}_2^{n_2}\times\cdots\times\mathcal{D}_k^{n_k}}[R^{\gamma}(f;\mathcal{D})-R^{\gamma}(f;S)\\
&>\frac{\varepsilon}{2}]\\
=\ &\mathbb{P}_S[R^{\gamma}(f;\mathcal{D})-\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}R^{\gamma,a,b}(f;\mathcal{D}_a,S_b)\\
&+\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}R^{\gamma,a,b}(f;\mathcal{D}_a,S_b)-R^{\gamma}(f;S)>\frac{\varepsilon}{2}]\\
\le\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\mathbb{P}_{S^b}[R^{\gamma,a,b}(f;\mathcal{D})-R^{\gamma,a,b}(f;\mathcal{D}_a,S_b)\\
&>\frac{\varepsilon}{4}]+\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\mathbb{P}_S[R^{\gamma,a,b}(f;\mathcal{D}_a,S_b)\\
&-R^{\gamma,a,b}(f;S)>\frac{\varepsilon}{4}]\\
=\ &I+II.
\end{aligned}$$

We need to bound I and II. To bound II, we define

$$l_{\gamma}(f,S_a,v^b)=\frac{1}{n_a}\sum_{i=1}^{n_a}l_{\gamma}(f,v_i^a,v_j^b).$$

By Hoeffding's inequality, we deduce

$$\begin{aligned}
II\ =\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\mathbb{P}_{S_a\cup S_b}[\max_{1\le j\le n_b}l_{\gamma}(f,\mathcal{D}_a,v_j^b\\
&-\max_{1\le j\le n_b}l_{\gamma}(f,S_a,v_j^b)>\frac{\varepsilon}{4}]\\
\le\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\mathbb{P}_{S_a\cup S_b}[\max_{1\le j\le n_b}|l_{\gamma}(f,\mathcal{D}_a,v_j^b)\\
&-l_{\gamma}(f,S_a,v_j^b)|>\frac{\varepsilon}{4}]\\
\le\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{j=1}^{n_b}\mathbb{P}_{S_a\cup S_b}[|l_{\gamma}(f,\mathcal{D}_a,v_j^b)\\
&-l_{\gamma}(f,S_a,v_j^b)|>\frac{\varepsilon}{4}]\\
\le\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}n_b\sup_{v^b}\mathbb{P}_{S^a}[|l_{\gamma}(f,\mathcal{D}_a,v_j^b)\\
&-l_{\gamma}(f,S_a,v_j^b)|>\frac{\varepsilon}{4}]\\
\le\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}2n_be^{-\frac{n_a\varepsilon^2}{8}}.
\end{aligned}$$

For I, we yield

$$\begin{aligned}
I\ =\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\mathbb{P}_{S_b}[\sup_{v^b}l_{\gamma}(f,\mathcal{D}_a,v^b)\\
&-\max_{1\le j\le n_b}l_{\gamma}(f,\mathcal{D}_a,v_j^b)>\frac{\varepsilon}{4}]\\
=\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\mathbb{P}_{S_b}[\max_{1\le j\le n_b}l_{\gamma}(f,\mathcal{D}_a,v_j^b)\\
&<\sup_{v^b}l_{\gamma}(f,\mathcal{D}_a,v^b)-\frac{\varepsilon}{4}]\\
=\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\Pi_{j=1}^{n_b}\mathbb{P}_{v_j^b}[l_{\gamma}(f,\mathcal{D}_a,v_j^b)\\
&<\sup_{v^b}l_{\gamma}(f,\mathcal{D}_a,v^b)-\frac{\varepsilon}{4}]\\
=\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}(\delta_{\mathcal{D}}^{a,b}(f,\gamma,\frac{\varepsilon}{4}))^{n_b}.
\end{aligned}$$

For any fixed $f\in\mathcal{F}$, by combining I and II, we get

$$\begin{aligned}
&\mathbb{P}_{S\sim\mathcal{D}_1^{n_1}\times\mathcal{D}_2^{n_2}\times\cdots\times\mathcal{D}_k^{n_k}}[R^{\gamma}(f;\mathcal{D})-R^{\gamma}(f;S)>\frac{\varepsilon}{2}]\\
\le\ &\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}(2n_be^{-\frac{n_a\varepsilon^2}{8}}+(\sup_{f\in\mathcal{F}}\delta_{\mathcal{D}}^{a,b}(f,\gamma,\varepsilon/4))^{n_b}).
\end{aligned}$$

The remaining proof from an application of standard tricks converts the above bound to a uniform convergence result that established uniformly over all $f\in\mathcal{F}$ using covering number approach. □

III. Fast Multi-dividing Ontology Algorithm for Ontology SVM

The aim of this section is to present a fast training algorithm for standard multi-dividing ontology SVM by means of the same infinite push framework showed in the above section. The multi-dividing ontology SVM algorithm minimizes a regularized bound in (11) with $p=1$ over an RKHS $\mathcal{F}$ can be expressed as:

$$\min_{f\in\mathcal{F}}[R_1^H(f;S)+\frac{\lambda}{2}\|f\|_{\mathcal{F}}^2],\tag{19}$$

where $R_1^H(f;S)$ is defined in (11) with $p=1$ and we use hinge loss $l_H(f,v_i^a,v_j^b)$ to replace the $\mathrm{I}(f(v_i^a)<f(v_j^b))$ terms. By expanding (19) in terms of the definition of $R_1^H(f;S)$, we get

$$\min_{f\in\mathcal{F}}[\frac{\lambda}{2}\|f\|_{\mathcal{F}}^2+\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\frac{1}{\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}n_an_b}\sum_{i=1}^{n_a}\sum_{j=1}^{n_b}(1-(f(v_i^a)-f(v_j^b)))_+].\tag{20}$$

By introducing slack variables $\xi_{ij}^{a,b}$ for each pair of (a,b) corresponding to the max in the hinge loss terms $(1-(f(v_i^a)-f(v_j^b)))_+=\max(1-(f(v_i^a)-f(v_j^b)),0)$ and setting $C=\frac{1}{\lambda}$, we obtain

$$\min_{f,\xi_{ij}^{a,b}}[\frac{\lambda}{2}\|f\|_{\mathcal{F}}^2+\frac{1}{\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}n_an_b}\sum_{i=1}^{n_a}\sum_{j=1}^{n_b}\xi_{ij}^{a,b}]$$

subject to $\quad\xi_{ij}^{a,b}\ge1-(f(v_i^a)-f(v_j^b))\qquad\forall i,j,(a,b)$

$$\xi_{ij}^{a,b} \geq 0 \qquad \forall i, j, (a,b).$$

By means of Lagrange multipliers and the dual technology, we get the following optimization problem in the $n_a n_b$ variables $\{\alpha_{ij}^{a,b} : 1 \leq i \leq n_a, 1 \leq j \leq n_b\}$ for each pair of (a,b):

$$\min_{\alpha_{ij}^{a,b}}[\frac{1}{2}\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{i,j}\sum_{k,l}\alpha_{ij}\alpha_{kl}\phi_K(v_i^a, v_j^b, v_k^a, v_l^b)$$

$$-\sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\sum_{i,j}\alpha_{ij}]$$

$$\text{subject to} \quad 0 \leq \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\alpha_{ij} \leq \sum_{a=1}^{k-1}\sum_{b=a+1}^{k}\frac{C}{n_a n_b},$$

$$\forall i, j, (a,b).$$

Algorithm 2. Fast multi-dividing training ontology algorithm via projection method

Inputs: training sample $S = (S_1, S_2, \cdots, S_k) \in V^{n_1} \times V^{n_2} \times \cdots \times V^{n_k}$ and a kernel function $K : V \times V \to \mathbb{R}$

For $a = 1$ to $k-1$:

For $b = a+1$ to k:

Parameters $C^{a,b}, t_{max}^{a,b}, \eta_0^{a,b}$

Initialize: $(\alpha_{ij}^{a,b})^{(1)} \leftarrow \frac{C^{a,b}}{n_a n_b} \quad \forall 1 \leq i \leq n_a, 1 \leq j \leq n_b$

For $t = 1$ to $t_{max}^{a,b}$ do: $(\alpha^{a,b})^{\frac{t+1}{2}} \leftarrow (\alpha^{a,b})^{(t)} - \frac{\eta_0^{a,b}}{\sqrt{t}}\nabla Q^{a,b}((\alpha^{a,b})^{(t)})$

For $1 \leq i \leq n_a, 1 \leq j \leq n_b$:

If $((\alpha_{ij}^{a,b})^{(\frac{t+1}{2})} < 0)$ Then $(\alpha_{ij}^{a,b})^{(t+1)} \leftarrow 0$

Else if $((\alpha_{ij}^{a,b})^{(\frac{t+1}{2})} > \frac{C^{a,b}}{n_a n_b})$ Then $(\alpha_{ij}^{a,b})^{(t+1)} \leftarrow \frac{C^{a,b}}{n_a n_b}$

Else $\frac{C^{a,b}}{n_a n_b} \leftarrow (\alpha_{ij}^{a,b})^{(\frac{t+1}{2})}$

End For.

End For.

End For.

End For.

Output: $f(v) = \sum_{i,j}(\alpha_{ij}^{a,b})^{t^*}(K(v_i^a, v) - K(v_j^b, v))$, where $(t^{a,b})^* = \text{argmin}_{1 \leq t \leq (t_{max}^{a,b}+1)} Q^{a,b}((\alpha^{a,b})^{(t)})$.

IV. Experiments

The above ontology learning algorithms can be used in ontology concepts similarity measurement and ontology mapping. The basic idea is: via the ontology gradient computation model, the ontology graph is mapped into a real line consisting of real numbers. The similarity between two concepts then can be measured by comparing the difference between their corresponding real numbers. To show the effectiveness of our new ontology algorithms, four experiments concerning ontology measure and ontology mapping are designed below.

A. Ontology similarity measure experiment on plant data

In the first experiment, we use plant "PO" ontology O_1 which was constructed in the website www.plantontology.org. The structure of O_1 is presented in Fig. 1. $P@N$ (Precision Ratio see Craswell and Hawking [17] is used to measure the quality of the experiment data.

We first give the closest N concepts for every vertex on the ontology graph by experts in plant field, and then we obtain the first N concepts for every vertex on ontology graph by the algorithm 1 and 2, and compute the precision ratio. Specifically, for vertex v and given integer $N > 0$. Let $Sim_v^{N,\text{expert}}$ be the set of vertices determined by experts and it contains N vertices having the most similarity of v. Let

$$v_v^1 = \underset{v' \in V(G)-v}{\text{argmin}}\{|f(v) - f(v')|\},$$

$$v_v^2 = \underset{v' \in V(G)-\{v, v_v^1\}}{\text{argmin}}\{|f(v) - f(v')|\},$$

$$\cdots$$

$$v_v^N = \underset{v' \in V(G)-\{v, v_v^1, \cdots, v_v^{N-1}\}}{\text{argmin}}\{|f(v) - f(v')|\},$$

and

$$Sim_v^{N,\text{algorithm}} = \{v_v^1, v_v^2, \cdots, v_v^N\}.$$

Then the precision ratio for vertex v is denoted by

$$\text{Pre}_v^N = \frac{|Sim_v^{N,\text{algorithm}} \cap Sim_v^{N,\text{expert}}|}{N}.$$

The $P@N$ average precision ratio for ontology graph G is then stated as

$$\text{Pre}_G^N = \frac{\sum_{v \in V(G)} \text{Pre}_v^N}{|V(G)|}.$$

At the same time, we apply ontology methods in [14], [15] and [16] to the "PO" ontology. Then we calculate the average precision ratio by these three algorithms and compare the results with algorithm 1 and algorithm 2 in the paper. Part of the data refer to Table 1.

When N= 3, 5 or 10, the precision ratio by means of our algorithms are higher than the precision ratio determined by algorithms proposed in [14], [15] and [16]. In particular, when N increases, such precision ratios are increasing apparent. Hence, the algorithms described in our paper are superior to the method proposed by [14], [15] and [16].

B. Ontology mapping experiment on humanoid robotics data

For the second experiment, we use "humanoid robotics" ontologies O_2 and O_3. The structure of O_2 and O_3 are showed in Fig. 2 and Fig. 3, respectively. The ontology O_2 presents the leg joint structure of bionic walking device for six-legged robot, while the ontology O_3 presents the exoskeleton frame of a robot with wearable and power-assisted lower extremities.

The goal of this experiment is to give ontology mapping between O_2 and O_3. We also use $P@N$ Precision Ratio to measure the quality of experiment. Again, we apply ontology algorithms in [18], [15] and [16] on "humanoid robotics" ontology, and compare the precision ratio which are gotten from three methods. Some results refer to Tab. 2.

Taking N= 1, 3 or 5, the precision ratio in terms of our infinite push based ontology algorithms are higher than the precision ratio determined by algorithms proposed in [18], [15] and [16]. Specially, as N increases, the precision ratios in view of our algorithms are increasing apparently. In this point of view, the algorithms described in our paper are superior to the method proposed by [18], [15] and [16].

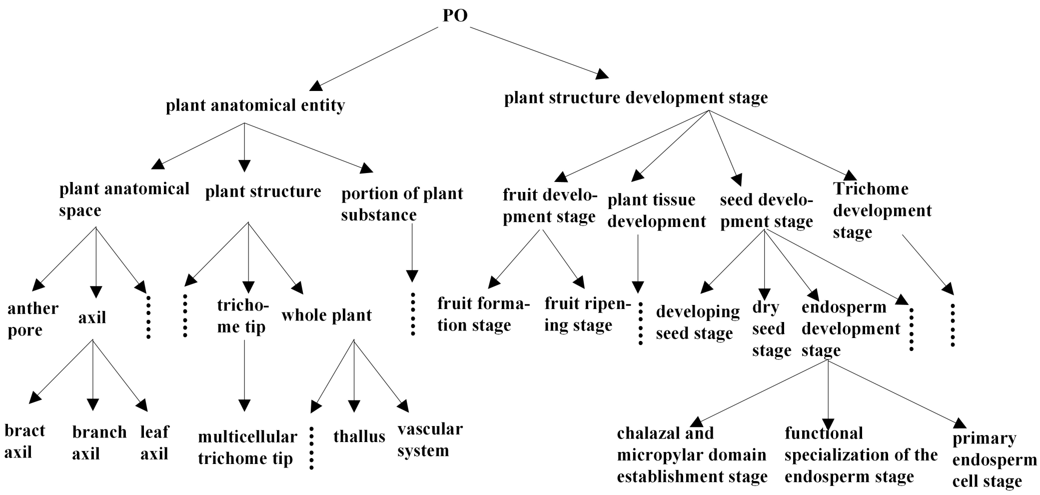

Fig. 1. The Structure of "PO" Ontology.

TABLE I
TAB. 1.THE EXPERIMENT RESULTS OF ONTOLOGY SIMILARITY MEASURE

	P@3 average precision ratio	P@5 average precision ratio	P@10 average precision ratio
Algorithm 1 in our paper	0.5103	0.6176	0.7904
Algorithm 2 in our paper	0.5232	0.6273	0.8018
Algorithm in [14]	0.4549	0.5117	0.5859
Algorithm in [15]	0.4282	0.4849	0.5632
Algorithm in [16]	0.4831	0.5635	0.6871

TABLE II
TAB. 2. THE EXPERIMENT RESULTS OF ONTOLOGY MAPPING

	P@1 average precision ratio	P@3 average precision ratio	P@5 average precision ratio
Algorithm 1 in our paper	0.4444	0.5557	0.7111
Algorithm 2 in our paper	0.2778	0.5000	0.6889
Algorithm in [18]	0.2778	0.4815	0.5444
Algorithm in [15]	0.2222	0.4074	0.4889
Algorithm in [16]	0.2778	0.4630	0.5333

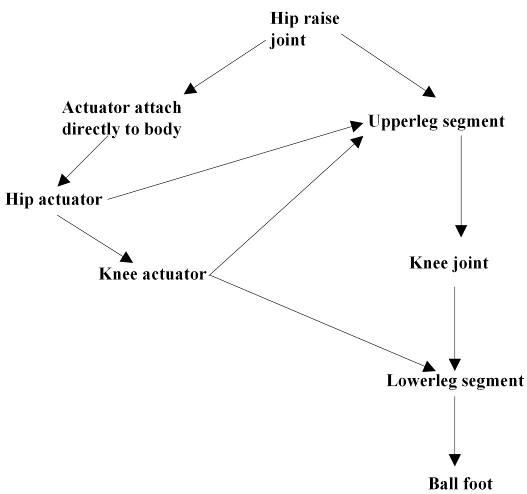

Fig. 2. 'Humanoid Robotics" Ontology O_2.

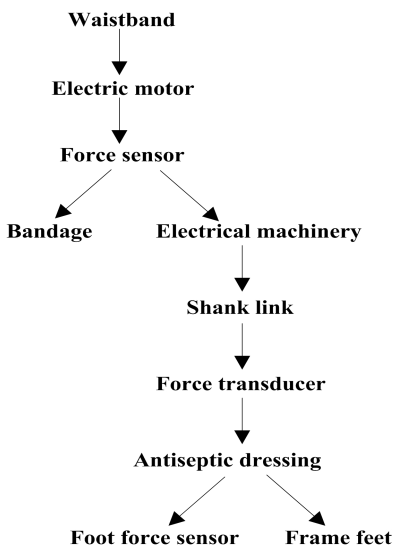

Fig. 3. "Humanoid Robotics" Ontology O_3.

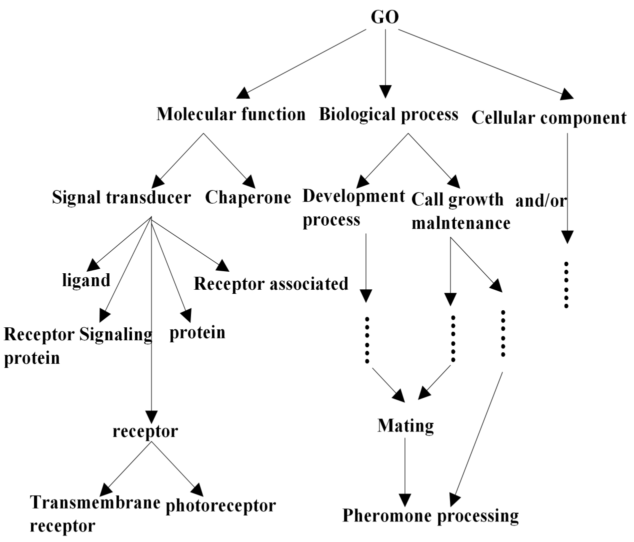

Fig. 4. The Structure of “GO” Ontology.

C. Ontology similarity measure experiment on biology data

In the third experiment, we use gene “GO” ontology O_4 which was constructed in the website http: //www. geneontology. The structure of O_4 is presented in Fig. 4. Again, $P@N$ is used to measure the quality of the experiment data. At the same time, we apply ontology method in [15], [16] and [19] to the “GO” ontology. Calculating the average precision ratio by these three algorithms and comparing the results to algorithm 1 and algorithm 2 rose in our paper, part of the data refer to Table 3.

When N= 3, 5 or 10, the precision ratio by virtue of our ontology algorithms are higher than the precision ratio determined by algorithms proposed in [15], [16] and [19]. In particular, when N increases, such precision ratios are increasing apparent. Therefore, the algorithms described in our paper are superior to the methods proposed by [15], [16] and [19].

D. Ontology mapping experiment on physics education data

For the last experiment, we use “physics education” ontologies O_5 and O_6. The structure of O_5 and O_6 are showed in Fig. 5 and Fig. 6, respectively.

The goal of this experiment is to give ontology mapping between O_5 and O_6. We also use $P@N$ precision ratio to measure the quality of experiment. Again, we apply ontology algorithms in [15], [16] and [20] on “physics education” ontology, and compare the precision ratio which is gotten from three methods. Some results refer to Tab. 4.

Taking N= 1, 3 or 5, the precision ratio in terms of our infinite push based ontology mapping algorithms are higher than the precision ratio determined by algorithms proposed in [15], [16] and [20]. Specially, as N increases, the precision ratios in view of our algorithms are increasing apparently. Therefore, the algorithms described in our paper are superior to the methods proposed by [15], [16] and [20].

V. Conclusions

As a data structural representation and storage model, ontology has been widely used in various fields and proved to

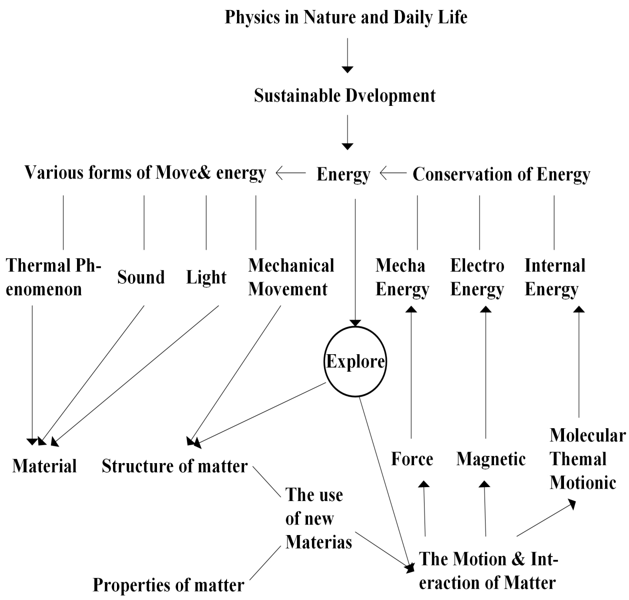

Fig. 5. “Physics Education” Ontology O_5.

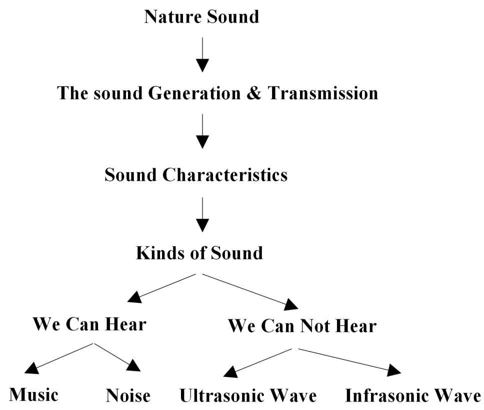

Fig. 6. “Physics Education” Ontology O_6.

have high efficiency. One ontology learning trick is mapping each vertex to a real number, and the similarity is judged by the difference between the real number which the vertices correspond to. In this paper, we raise a infinite push learning model for ontology application in multi-dividing setting. The generalization bound is given by means of covering number approach. The experiments show the effectiveness of the new multi-dividing ontology algorithms. The new technology contributes to the state of art for applications and the result got in our paper illustrates the promising application prospects for multi-dividing ontology algorithm.

References

[1] J. M. Przydzial, B. Bhhatarai, and A. Koleti, “GPCR Ontology: Development and Application of A *G* Protein-Coupled Receptor Pharmacology Knowledge Framework,” *Bioinformatics*, vol. 29, no. 24, pp. 3211-3219, 2013.

[2] S. Koehler, S. C. Doelken, and C. J. Mungall, “The Human Phenotype Ontology Project: Linking Molecular Biology and Disease Through Phenotype Data,” *Nucleic Acids Research*, vol. 42, no. D1, pp. 966-974, 2014.

[3] M. Ivanovic and Z. Budimac, “An Overview of Ontologies and Data Resources in Medical Domains,” *Expert Systerms and Applications*, vol. 41, no. 11, pp. 5158-5166, 2014.

TABLE III
TAB. 3. 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
Algorithm 1 in our paper	0.5171	0.6421	0.7942	0.9054
Algorithm 2 in our paper	0.5093	0.6401	0.7718	0.8653
Algorithm in [15]	0.4638	0.5348	0.6234	0.7459
Algorithm in [16]	0.4356	0.4938	0.5647	0.7194
Algorithm in [19]	0.4213	0.5183	0.6019	0.7239

TABLE IV
TAB. 4. THE EXPERIMENT RESULTS OF ONTOLOGY MAPPING

	P@1 average precision ratio	P@3 average precision ratio	P@5 average precision ratio
Algorithm 1 in our paper	0.6913	0.7849	0.9161
Algorithm 2 in our paper	0.7097	0.8172	0.9032
Algorithm in [15]	0.6129	0.7312	0.7935
Algorithm in [16]	0.6913	0.7556	0.8452
Algorithm in [20]	0.6774	0.7742	0.8968

[4] A. Hristoskova, V. Sakkalis, and G. Zacharioudakis, “Ontology-Driven Monitoring of Patient’s Vital Signs Enabling Personalized Medical Detection and Alert,” *Sensors*, vol. 14, no. 1, pp. 1598-1628, 2014.

[5] M. A. Kabir, J. Han, and J. Yu, “User-Centric Social Context Information Management: An Ontology-Based Approach and Platform,” *Personal and Ubiquitous Computing*, vol. 18, no. 5, pp. 1061-1083, 2014.

[6] Y. L. Ma, L. Liu, K. Lu, B. H. Jin, and X. J. Liu, “A Graph Derivation Based Approach for Measuring and Comparing Structural Semantics of Ontologies,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 5, pp. 1039-1052, 2014.

[7] Z. Li, H. S. Guo, Y. S. Yuan, and L. B. Sun, “Ontology Representation of Online Shopping Customers Knowledge in Enterprise Information,” *Applied Mechanics and Materials*, vol. 483, pp. 603-606, 2014.

[8] R. Santodomingo, S. Rohjans, M. Uslar, J. A. Rodriguez-Mondejar, and M.A. Sanz-Bobi, “Ontology Matching System for Future Energy Smart Grids,” *Engineering Applications of Artificial Intelligence*, vol. 32, pp. 242-257, 2014.

[9] T. Pizzuti, G. Mirabelli, M. A. Sanz-Bobi, and F. Gomez-Gonzalez, “Food Track & Trace Ontology for Helping the Food Traceability Control,” *Journal of Food Engineering*, vol. 120, no. 1, pp. 17-30, 2014.

[10] N. Lasierra, A. Alesanco, and J. Garcia, “Designing An Architecture for Monitoring Patients at Home: Ontologies and Web Services for Clinical and Technical Management Integration,” *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 3, pp. 896-906, 2014.

[11] M. Tovar and D. Pinto, Azucena Montes, Gabriel Gonzalez, and Darnes Vilarino, “Identification of Ontological Relations in Domain Corpus Using Formal Concept Analysis,” *Engineering Letters*, vol. 23, no.2, pp. 72-76, 2015.

[12] 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.

[13] 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.

[14] Y. Y. Wang, W. Gao, Y. G. Zhang and Y. Gao, “Ontology Similarity Computation Use Ranking Learning Method,” *The 3rd International Conference on Computational Intelligence and Industrial Application*, Wuhan, China, 2010, pp. 20–22.

[15] X. Huang, T. W. Xu, W. Gao and Z. Y. Jia, “Ontology Similarity Measure and Ontology Mapping Via Fast Ranking Method,” *International Journal of Applied Physics and Mathematics*, vol. 1, no. 1, pp. 54-59, 2011.

[16] 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.

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

[18] W. Gao and M. H. Lan, “Ontology Mapping Algorithm Based on Ranking Learning Method,” *Microelectronics and Computer*, vol. 28, no. 9, pp. 59-61, 2011.

[19] 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.

[20] 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.

W. 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.

Now, he acts as lecturer 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.

L. L. Zhu, male, was born in the city of Badong, Hubei Province, China on Sep. 20, 1975. He got Master degrees on computer software and theory from Yunnan normal university in 2007. Now, he acts as associate professor in the department of computer engineering, Jiangsu University of Technology. As a researcher in computer science, his interests are covered two disciplines: computer network and artificial intelligence.

Y. Guo, female, is work in School of Computer Science and Engineering, Soochow University, Suzhou, China.