Ontology Similarity Measure and Ontology Mapping Via Fast Ranking Method

Xiao Huang, Tianwei Xu, Wei Gao, and Zhiyang Jia

Abstract—**Ontology similarity calculation and ontology mapping are important research topics in information retrieval. By analyzing the fast algorithm for learning large scale preference relations, we propose the fast algorithm for ontology similarity measure and ontology mapping. Via the ranking learning algorithm, the ontology graph is mapped into a line consists of real numbers. The similarity between two concepts then can be measured by comparing the difference between their corresponding real numbers. The new algorithm has lower compute complexity and two experimental results show that the proposed algorithm has high accuracy and efficiency both on ontology similarity calculation and ontology mapping. Some issues are discussed in the last section as further works.**

***Index Terms*—ontology; similarity computation; ranking; error function; conjugate gradient; ε -exact approximation**

I. Introduction

The problem of information retrieval is how to find the useful information for user’s need from the mass of information. Information retrieval, as one of the most active branches of information theory, refers to cross-cooperation of many fields. At present, the key reason that leads the low-quality of text information retrieval is lacking of semantics retrieval tools. Usually simply and mechanically syntax match based on the syntax layer for understanding of user’s information needs lack of the capacity of semantic understanding.

General speaking, information retrieval technology can be broadly divided into three categories: text retrieval, data retrieval and subject retrieval.

Text retrieval is to compare the user's query request to the form of keywords, and every word in the full text regardness the semantic matching between the query request and documents. This retrieval mode mainly based on word frequency analysis techniques. Since the mode is only aim at text matching, the searched information always large and without human intervention. The drawback is that the return of information overloads, and there are a lot of irrelevant information, users must filter from the results.

Data retrieval is mainly aim at structured information system, query requests and data follow a certain format, and with a certain structure. It allowing to search specific fields, such as: disease = "fever". More representative as a variety of commercial databases. Data retrieval depends on the quality of encoding, has large retrieval cost, and the searched information is relative accurate, but easily missed relevant data. Overall, it has great limitations.

Subject retrieval is to collect information manually or semi-automatic. After accessing to documents, write the document description, which will be added to the appropriate pre-defined subject category. Users from the entrance of the provided basic several categories, level to the next level visits, and obtain more satisfied results. The advantage of the subject retrieve is that it can make the unclear original information request become gradually clear along with tips of the level categories. The disadvantage is that it spends a longer time on searching, there are no suitable relevant categories for retrieval to the new emerging concepts.

These three forms of query retrieval can not build the intrinsic link between information. The results often cause a large amount of non-related information, while it may be missing something important. To the real search requirements of users, or it is difficult to use "keywords" to honestly express, or as a result of the different expression habits. Different subject areas have the different expresssion form to the same concept, so that search results can not be accurate and comprehensive to reflect the users’ requirements.

The essence of difficulties of the above-mentioned information retrieval is that the traditional information retrieval technologies are lack of knowledge processing capacity and understanding capacity.

With the people's growing demand of knowledge organization and information reuse, ontology as a semantic and knowledge model has aroused the concern of many researchers, and widely applied in many areas of the computer science such as knowledge engineering, digital libraries, software reuse, information retrieval and Semantic Web. From the point of knowledge-sharing view, the ontology can be conceptualized as an explicit description of the objective existence, and the relationship between the concept and description can be well representation. It

Manuscript received June 2, 2011; revised June 28, 2011.

This work was supported in part by National Natural Science Foundation of China 60903131, Key Laboratory of National Education Informatization, Ministry of Education, Yunnan Normal University, China and Zhejiang Provincial Education Department of China Y201018272.

Xiao Huang is with the Department of Mathematics, Physics and Information Engineering and the Department of Education, Zhejiang Normal University, Zhejiang, Jinhua 321004, P.R. China. (e-mail: huangxiao@zjnu.cn)

Tianwei Xu is with the Department of Information, Yunnan Normal University, Yunnan, Kunming, 650092, P.R. China. Also, he is PhD student in the institute of higher education, Huazhong University of science and technology, Hubei, Wuhan, 430074, P.R. China. (E-mail: xutianwei@ynnu.edu. cn).

Wei Gao is with the Department of Information, Yunnan Normal University, Yunnan, Kunming, 650092, P.R. China. Also he is PhD student in the Department of Mathematics, Soochow University, Jiangsu, Suzhou, 215006, P.R. China. (E-mail: gaowei@ynnu.edu.cn).

Zhiyang Jia is with the Tourism and Literature College, Yunnan University, Yunnan, Lijiang, 674100, P.R. China. (E-mail: jdyjazz@gmail.com).

originated from the philosophical concept, the philosopher used to describe the nature of things, was later well introduced to the field of artificial intelligence research.

Ontology is a conceptualization clear description, it abstractes certain application field of the real world into a set of concepts and relationships of concepts. Integrating the ontology into the technology of text information retrieval not only inherit the advantages of information retrieval but also overcome the limitations that concepts information retrieval can not deal with the relationships of the concepts. It raise the accurate ratio and recall ratio of information retrieval.

As the ontology has the ability to express concept semantics through the relationship between concepts, portray the intrinsic link between concepts, and excavate those hidden and not clear concepts and information. So, it can better meet user requirements in the recall and precision aspects, and realize the retrieval intelligentize. At the same time, ontology-based retrieval methods are more in line with the of human thought, it can overcome the shortcomings of the information redundancy or information missing caused by the traditional information retrieval methods, and the query results can be more reasonable.

So, the reason for the ontology research so common. There isn't a standard communication of grammar or semantics between computer systems and people. As a model of formal shown in the conceptualization of shared, ontology provided a good way and resolved the problem to some extent. As a semantics communication method between people and machines, machinery and machines, ontology is exactly an agreement. Also, ontology is the foundation of the semantic understanding. Now, ontology similarity computation is widely used in medical science biology science [see 1] and social science [see 2]. As ontology used in information [see 3], every vertex as a concept of ontology, measure the similarity of vertices using the attraction of ontology graph.

For this purpose, we do some research on the ontology-based text information retrieval based on ontology to express the text and query, to expand the specific semantic meaning of the information to be searched, to solve the problems exist in the traditional information retrieval process, to enhance the quality and efficiency of information retrieval. In this paper, we focus on the algorithm to ontology similarity measure and ontology mapping, not other technology problem.

Let graphs $G_1, G_2, \cdots, G_k$ corresponding to ontologies $O_1, O_2, \cdots, O_k$, respectively, and $G=G_1+G_2+\cdots+G_k$. For every vertex $v \in V(G_i)$, where $1 \le i \le k$, the goal of ontology mapping is finding similarity vertices from G-G_i. So, the ontology mapping problem is also ontology similarity measure problem.

Choose the parameter $M \in [0,1]$, let A, B are two concepts on ontology and Sim(A,B)>M, then return B as retrieval expand when search concept A. The traditional method to compute similarity of A and B usually via following computation model:

$$\mathrm{Sim}(A,B)=\alpha_1\,\mathrm{Sim}_{\mathrm{name}}(A,B)+\alpha_2\,\mathrm{Sim}_{\mathrm{instance}}(A,B)$$
$$+\alpha_3\,\mathrm{Sim}_{\mathrm{attribute}}(A,B)\ +\alpha_4\,\mathrm{Sim}_{\mathrm{structure}}(A,B)$$

where α_i $(1 \le i \le 4)$ is real number and satisfy $\alpha_1+\alpha_2+\alpha_3+\alpha_4=1$. This method has some disadvantages, such as:

- In order to compute the similarity, there are lots of parameters have to choose. These parameters mainly contained in the computational formula of $\mathrm{Sim}_{\mathrm{name}}(A,B)$, $\mathrm{Sim}_{\mathrm{instance}}(A,B)$, $\mathrm{Sim}_{\mathrm{attribute}}(A,B)$ and $\mathrm{Sim}_{\mathrm{structure}}(A,B)$, not only α_i.
- High compute complexity.
- One technology to solve this problem is application ranking method to ontology similarity measure.

Largely motivated by applications in information retrieva1, collaborative filtering, and search engines, ranking has recently received significant attention in the statistical machine learning and information retrieval communities. In the typical formulation, it compares two instances and determines which one is ranking higher than other. Then, the instances are ranked according to the desired preference relations. Other ranking algorithm can be seen [4-9], and the analysis for ranking algorithm can be seen [10-13].

Ranking learning problem has been formalized in many ways. We adopt the most general formulation based on directed preference graphs [see 14, 15, and 17]. This provides flexibility to learn different kinds of preference relations by changing the graph. Given training data A, $G^*=(V^*;E^*)$ is directed preference graph encoding the preference relations, and ranking function f choose form a function class F.

The main contribution of our paper is propose a new ontology similarity measure method and ontology mapping using the technology of fast algorithm for learning large scale preference relations. The organization of this paper is as follows: we describe the fast algorithm raised in [17] in Section II, and apply this trick, we give the new ontology similarity measure and ontology mapping algorithm in Section III. Two experiments are obtained in Section IV which shows that the new algorithm have high quality.

II. Fast Algorithm for Learning Large Scale Preference Relations Raised in [17]

A. Setting

Let $A = \cup_{j=1}^{S}$ ($A^j = \{x_i^j \in R^d\}_{i=1}^{m_j}$) is training data which contains S classes (sets); each class A^j contains m_j samples, and A has total $m=\sum_{j=1}^{S} m_j$ samples. Each vertex of $G^*=(V^*;E^*)$ corresponds to a class A^j. The existence of a directed edge E_{ij} from $A^i \to A^j$ means that all training examples in A^j should be preferred or ranked higher than any training example in A^i, i.e. , $\forall$ ($x_k^i \in A^i$, $x_l^j \in A^j$), $x_k^i \succ x_l^j$.

The preference relation $x \succ y$ means x is ranked higher than y. $f: R^d \to R$ is a ranking function representing the preference relation $\succ$ if $\forall x, y \in R^d$, $x \succ y \Leftrightarrow f(x) \ge f(y)$. The goal of algorithm is to learn a ranking function $f: R^d \to R$ for as many pairs as possible in both A and other unseen examples. The output $f(x_k)$ can be sorted to obtain a ranking for a set of test samples $\{f(x_k) \in R^d\}$.

The quality of the ranking for arbitrary preference graphs measured by generalized version of the Wilcoxon-Mann-Whitney (WMW) statistic [see 11 and 12] that is averaged over pairs of samples:

$$\text{WMW}(f,A,G^*)=\frac{\sum_{E_{ij}}\sum_{k=1}^{m_i}\sum_{l=1}^{m_j}1_{f(x_l^j)\geq f(x_k^i)}}{\sum_{E_{ij}}\sum_{k=1}^{m_i}\sum_{l=1}^{m_j}1}, \qquad (1)$$

where $1_{a\geq b}$ = 1 if $a \geq b$, and 0 otherwise. The WMW is an estimate of the probability of correct pairwise ordering Pr[$f(x_i)$ ≥ $f(x_j)$], for a randomly drawn pair (x_i, x_j) such that $x_i \succ x_j$.

It is a discrete optimization problem for Maximizing the WMW. Most ranking algorithms optimize a continuous relaxation instead. The WMW can be computed in $O(md + m\log m)$ time, but previous algorithms will take $O(m^2)$ time for evaluating the relaxed version or its gradient.

A. The MAP Estimator

Consider the family of linear ranking functions: $F = \{f_w\}$, where for any $x,w \in R^d$, $fw(x) = w^Tx$; w are the weights to be learnt. Choose w to maximize the generalized WMW, while for computational efficiency, it shall instead maximize a continuous surrogate via the log-likelihood $L(fw;A;\ G^*)$ = log Pr [correct ranking|w]:

$$L(fw;A;\ G^*) = \log\prod_{E_{ij}}\prod_{k=1}^{m_i}\prod_{l=1}^{m_j}\mathrm{P_r}[f_w(x_l^j) > f_w(x_k^i)|w]\,.$$

Assume each pair (x_l^j , x_k^i) is drawn independently, but only the original samples are drawn independently. Sigmoid function is used the to model the pairwise probability:

$$\mathrm{P_r}[f_w(x_l^j) > f_w(x_k^i)|w] = \sigma[w^T(x_l^j - x_k^i)]\,, \qquad (2)$$

here $\sigma(z) = (1+\exp(-z))^{-1}$ is the sigmoid function. Let $p(w) = N(w|0, \lambda^{-1})$ on the weights w, the optimal maximum posteriori (MAP) estimator is that the $\hat{w}_{MAP}$ = arg max$_w$ $L(w)$, where $L(w)$ is the penalized log-likelihood:

$$L(w) = -\frac{\lambda}{2}\|w\|^2 + \sum_{E_{ij}}\sum_{k=1}^{m_i}\sum_{l=1}^{m_j}\log\sigma[w^T(x_l^j - x_k^i)]\,.$$

B. Gradient Based Learning

Using the Polak-Ribiµere variant of the nonlinear conjugate gradients (CG) algorithm [see 18] to find the w that maximizes $L(w)$. This CG method only needs the gradient $g(w)$ and regardless evaluation of either $L(w)$ or the second derivative (Hessian) matrix. The gradient vector w.r.t. w is:

$$g(w) = -\lambda w - \sum_{E_{ij}}\sum_{k=1}^{m_i}\sum_{l=1}^{m_j}(x_l^j - x_k^i)\sigma[w^T(x_l^j - x_k^i)]\,.$$

Using the approximation $\sigma(z) \approx 1-\frac{1}{2}\mathrm{erfc}(\frac{\sqrt{3}z}{\sqrt{2\pi}})$, where the complementary error function is defined by erfc(z) $=\frac{2}{\sqrt{\pi}}\int_z^\infty e^{-t^2}dt$. Then, the approximate gradient (still $O(dM^2)$) can be represented as:

$$g(w) = \nabla_w L(w) \approx -\lambda w - \sum_{E_{ij}}\sum_{k=1}^{m_i}\sum_{l=1}^{m_j}(x_l^j - x_k^i)$$

$$[1-\frac{1}{2}\mathrm{erfc}(\frac{\sqrt{3}w^T(x_l^j - x_k^i)}{\sqrt{2\pi}})]. \qquad (3)$$

C. Fast Summation of erfc Function

Let $E_-^j(y)$ and $E_+^i(y)$ be the sum of N erfc functions which centered at $zi \in R$, with weights $q_i \in R$:

$$E(y) = \sum_{i=1}^{N} q_i \mathrm{erfc}(y - z_i)\,. \qquad (4)$$

Direct computation of (4) at M points $\{y_j \in R\}_{j=1}^{M}$ is $O(MN)$. Derive an ε -exact approximation algorithm to compute this in $O(M + N)$ time. For any $\varepsilon > 0$, $\hat{E}$ is an ε *-exact* approximation to E if the maximum absolute error relative to the total weight $Q_{abs} = \sum_{i=1}^{N}|q_i|$ is upper bounded by a specified ε , i.e. , $\max_{y_j}[|\hat{E}(y_j) - E(y_j)|/Q_{abs}] \leq \varepsilon$. The constant in $O(M + N)$ for algorithm depends on ε , and it can be arbitrary. At machine precision have no difference between the direct and the fast methods. This approach depends on retaining only the first few terms of an infinite series expansion for the erfc function.

For the truncated Fourier series representation derived in [see 19,20]:

$$\mathrm{Erfc}(z) = 1-\frac{4}{\pi}\sum_{n=1,n\equiv1(\mathrm{mod}\,2)}^{2p-1}\frac{e^{-n^2h^2}}{n}\sin(2nhz) + \mathrm{error}(z) \qquad (5)$$

Where $|\mathrm{error}(z)| < \left|\frac{4}{\pi}\sum_{n=2p+1,n\equiv1(\mathrm{mod}\,2)}^{\infty}\frac{e^{-n^2h^2}}{n}\sin(2nhz)\right| +$ Erfc($\frac{\pi}{2h} - |z|$). Here p is the truncation number and $h \in R$ related to the sampling interval. The fast algorithm to compute $E(y)$ according to (5) can be written as

$$E(y) = \sum_{i=1}^{N} q_i \mathrm{erfc}(y - z_i) =$$

$$\sum_{i=1}^{N} q_i[1-\frac{4}{\pi}\sum_{n=1,n\equiv1(\mathrm{mod}\,2)}^{2p-1}\frac{e^{-n^2h^2}}{n}\sin(2nhz) + \mathrm{error(z)}]\,. \qquad (6)$$

Regardless the error term of the time being, the sum $E(y)$ can be represented as:

$$\hat{E}(y) = \mathrm{Q}-\frac{4}{\pi}\sum_{i=1}^{N} q_i \sum_{n=1,n\equiv1(\mathrm{mod}\,2)}^{2p-1}\frac{e^{-n^2h^2}}{n}\sin(2nh(y - z_i))\,, \qquad (7)$$

Where $Q = \sum_{i=1}^{N} q_i$. y and z_i are entangled in the argument of the sin function, leading to a quadratic complexity. The crux of the algorithm is to separate them using the trigonometric identity:

$$\sin\{2nh(y - z_i)\} = \sin\{2nh(y - z^*)\}\cos\{2nh(zi - z^*)\}$$
$$- \cos\{2nh(y - z^*)\}\sin\{2nh(zi - z^*)\} \qquad (8)$$

It has changed all the points by z^*. Substituting the separated representation (8), exchanging the order of summation, and regrouping terms in (7),

$$\hat{E}(y) = \mathrm{Q}-\frac{4}{\pi}\sum_{n=1,n\equiv1(\mathrm{mod}\,2)}^{2p-1}A_n\sin(2nh(y - z^*)) +$$

$$\frac{4}{\pi}\sum_{n=1,n\equiv1(\mathrm{mod}\,2)}^{2p-1}B_n\cos(2nh(y - z^*))\,. \qquad (9)$$

Where

$$A_n = \frac{e^{-n^2h^2}}{n}\sum_{i=1}^{N} q_i\cos(2nh(y - z^*))\,,$$

$$B_n = \frac{e^{-n^2h^2}}{n}\sum_{i=1}^{N} q_i\sin(2nh(y - z^*))\,. \qquad (10)$$

The coefficients $\{A_n, B_n\}$ do not depend on y. So, every A_n,

B_n can be precomputed in $O(N)$ time. Since there are p such coefficients the total complexity to compute them is $O(pN)$. The term Q can also be pre-computed in $O(N)$ time. Once A, B, and Q are known, evaluation of $\hat{E}(y)$ at M points requires $O(pM)$ operations. Therefore, the computational complexity has reduced from the quadratic $O(NM)$ to the linear $O(p(N+M))$. It needs space to store the points and the coefficients A and B. Hence, the storage complexity is $O(N+M+p)$.

Obverse (6), we see that if we fixed h and p, then the error will increases with the $|z|$ increases. Therefore, as $|z|$ increases, h will decrease and consequently the series converges slower leading to a large truncation number. Since $s = (y-z_i) \in [-\infty, +\infty]$, p required to approximate erfc(s) can be very large for large $|s|$. Luckily, when $s \to +\infty$, then erfc(s) $\to 2$ and when $s \to \infty$, then erfc(s) $\to 0$ very quickly. Since a precision ε is we only demand, we can approximate:

$$\operatorname{erfc}(s) \approx \begin{cases} 2 & s < -r \\ p-\text{truncated} \quad \text{series} & -r \le s \le r \\ 0 & s > r \end{cases} \tag{11}$$

The truncation number p and the bound r have to be chosen so that the error is always less than ε for any s, e.g. , for error of the order 10^{-15} we should use the series expansion for $-6 \le s \le 6$. However, the value of $(y-z_i)$ for all pairs of z_i and y cannot be check by us. This is bad case that would lead us back to the quadratic complexity. In order to avoid this case, we should subdivide the points into clusters.

The domain can be sub-divided uniformly into K intervals of length $2r_x$. The N source points are assigned into K clusters, S_k for $k = 1, \cdots, K$ such that the center of each cluster is c_k. Computing the aggregated coefficients for each cluster and summing up the total contribution from all the influential clusters. For each cluster, we will use the series coefficients if $|y-c_k| \le r_y$. If $(y-c_k) < -r_y$, a contribution of $2Q_k$ will be included; if $(y-c_k) > r_y$, that cluster can be ignored. We choose the cut off radius r_y to achieve a given accuracy.

Since each z_i only belongs to one cluster, the cost to compute A, B, and Q is still $O(pN)$. Let l be the number of influential clusters, i.e., the clusters for which $|y-c_k| \le r_y$. Evaluating $\hat{E}(y)$ at M points due to these l clusters is $O(plM)$. Let m be the number of clusters for which $(y-c_k) < -r_y$,. Evaluating $\hat{E}(y)$ at M points due to these m clusters is $O(mM)$. Hence the total time is $O(pN + (pl + m)M)$. The storage complexity is $O(N + M + pK)$.

About choosing parameters: for given any $\varepsilon > 0$, we choose the parameters: r_x (the interval length), r_y (the cut off radius), p (the truncation number) and h, such that for any target y, $|\hat{E}(y) - E(y)| \le Q_{abs}\varepsilon$. The following choice guarantees that error< ε: (1) $r_x = 0.1\operatorname{erfc}^{-1}(\varepsilon)$, (2) $r_y = \operatorname{erfc}^{-1}(\varepsilon) + 2r_x$, (3) $h = \dfrac{\pi}{3(r + \operatorname{erfc}^{-1}(\varepsilon/2))}$, and (4) $p = \left\lceil \dfrac{1}{2h}\operatorname{erfc}^{-1}\left(\dfrac{\sqrt{\pi}h\varepsilon}{4}\right) \right\rceil$ (see [21] for details).

III. New Algorithm Via Fast Alghorithm

D. The Ideal of New Algorithm

The ranking learning algorithm can be used in ontology concept similarity measure. The based ideal is that: Via the ranking learning algorithm, the ontology graph is mapped into a line consists of real numbers. The similarity between two concepts then can be measured by comparing the difference between their corresponding real numbers.

E. Ontology Similarity Measure Design

For $v \in V(G)$. We use the one of following methods to obtain the similarity vertices and return the outcome to the users.

Method 1：Choose parameter M, return set $\{u \in V(G), |f(u) - f(v)| \le M\}$.

Method 2：Choose integer N, return the closest N concepts on the ranking list in $V(G)$.

Clearly, method 1 looks like more fair and method 2 can control the number of vertices that return to the users.

F. Ontology Mapping Design

For $v \in V(G_i)$，where $1 \le i \le k$. We use the one of following methods to obtain the similarity vertices and return the outcome to the users.

Method 1：Choose parameter M, return set $\{u \in V(G-G_i), |f(u) - f(v)| \le M\}$.

Method 2：Choose integer N, return the closest N concepts on the ranking list in $V(G-G_i)$.

Also, method 1 looks like more fair and method 2 can control the number of vertices that return to the users.

II. Experiments

Two experiments concern ontology measure and ontology mapping are desired follow.

A. Dimensionality reduction method for ontology information representation

To connect ontology to fast ranking algorithm, we should use a vector to express the vertex of information. This vector contains the information of name, instance, attribute and structure of vertex, where the instance of vertex is the set of its reachable vertex in the directed ontology graph.

Sometimes, these vectors have high dimension, thus lead high compute complexity. The method to solve this problem is using the technology of dimensionality reduction method, and this method raised by [22].

The generic problem of dimensionality reduction is the following. Given a set $x_1, \cdots, x_k$ of k points in R^l, find a set of points $y_1, \cdots, y_k$ in R^m ($m \ll l$), such that y_i "represents" x_i.

For an ontology graph G, $W_{ij} > 0$ if there exist an edge between vertex i and vertex j, where the value of W_{ij} is the value of edge; otherwise, $W_{ij} = 0$. Note that one of method to computer the value of W_{ij} is using the Heat kernel:

$$W_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}} \tag{12}$$

where parameter $t \in R$. Or, inverse multiquadric kernel:

$$W_{ij} = (c^2 + \|x_i - x_j\|^2)^{-\alpha} \tag{13}$$

where c and α are positive parameters.

Let W be weight matrix, and D is diagonal weight matrix, its entries are column (or row, since W is symmetric) sum of W, $D_{ii}=\sum_j W_{ji}$. $L=D-W$ is the Laplacian matrix. This is a symmetric, positive, semidefinite matrix which can be view as an operator on functions defined on vertices of G. Also, 0 is smallest eigenvalue of L. Then reduces to find the solution of

$$\arg\min_{Y^T DY=I} tr(Y^T LY)\ . \tag{14}$$

That is to say, compute the mth eigenvectors corresponding to mth smallest eigenvectors beyond 0, and note $\boldsymbol{y}_1,\cdots,\boldsymbol{y}_k$. Then the matrix $\boldsymbol{Y}=[\boldsymbol{y}_1\boldsymbol{y}_2\cdots\boldsymbol{y}_m]$ is the solution. $\boldsymbol{x}_i\rightarrow(f_1(i),f_2(i),\cdots,f_m(i))$.

However, in the ontology mapping, we should considering other problem concern computing edge weight. Since there are at least two connected components in graph, and each connected component represent ontology. Even two vertex i and j in different connected component are much closed, they are no edge on G, thus W_{ij}=0. But it is not reasonable in our application. Our method to deal with this problem is using the technology of δ -neighborhood method. For mutil-ontology graph G, there is an edge between vertex i and vertex j if and only if $\|x_i-x_j\|\le\delta$. The we get a new graph G^*, such that $V(G)=V(G^*)$ and the edge set of G^* satisfy δ -neighborhood condition. Then we use G^* supersede to G. After that, the weight matrix W is reasonable for the ontology mapping application.

Usually, we need a connected graph G^*. For this purpose, choose of parameter δ is very important. If δ is large, then $|E(G^*)|$ is large thus lead a high compute complexity. If δ is small, then the graph G^* have more connect components. So, the value of δ should be considered more carefully.

B. Ontology measure using fast algorithm

In this experiment, we construct following Computer Ontology O_1 as Fig. 1. The goal of algorithm is mapping the vertices on graph into a line consists of real numbers. The similarity between two concepts then can be measured by comparing the difference between their corresponding real numbers. Thus, the similarities we get are indirect similarity measure not direct one. We use $P@N$ (see [23]) Precision Ratio to measure the equality of experiment. First, the expert give the first N concepts for every vertex on ontology graph and also give the preference graph for learning, then we obtain the first N concepts for every vertex on ontology graph by algorithm and computer the precision ratio.

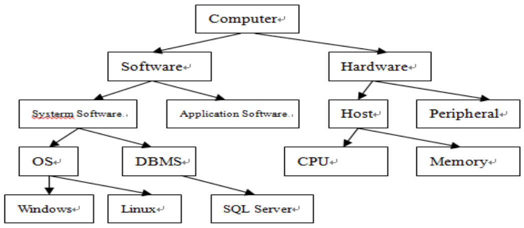


Fig.1 "Computer" Ontology O_1

The experiment shows that, $P@1$ Precision Ratio is 71.43%, $P@3$ Precision Ratio is 76.19%, $P@5$ Precision Ratio is 80%. Thus the algorithm have high efficient.

C. Ontology mapping using fast algorithm

In this experiment, we construct other Computer Ontologies O_2 as Fig. 2. The goal of algorithm is mapping the vertices on $G=G_1+G_2$ into a line consists of real numbers. We also use $P@N$ Precision Ratio to measure the equality of experiment.

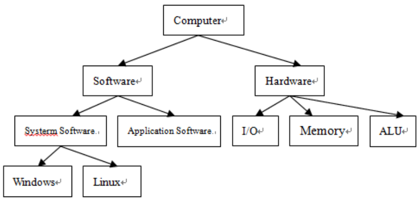


Fig.2 "Computer" Ontology O_2

The experiment shows that, $P@1$ Precision Ratio is 62.50%, $P@3$ Precision Ratio is 76.39%, $P@5$ Precision Ratio is 84.17%. Thus the algorithm have high efficient.

III. CONCLUSION AND FURTHER WORK

With the advent of the information age, human's ability to produce information is becoming rapidly grown. How to find out useful information for user from the mass of information becomes a very important issue. The increasing development of information retrieval technology will provide possibility for solving this problem. Applying ontology into information retrieval can maks semantic and knowledge of the information retrieval, this can raise the ratio of recall and precise of information retrieval system. Constructuring domain ontology and using it into information system can improve efficiency and the quality of retrieval system.

In this paper, we give a new algorithm for measuring the ontology similarity and ontology mapping using fast algorithm for learning large scale preference relations. The new algorithm has less complexity and also has high equality according to the experiment above.

There are many contents for ontology need to be further studied, including:

1) Domain ontology construction and improve. In this thesis, the domain ontology in a smaller scale, all of them are builded artificial. How to realize the ontology automatic, semi-automatic construction will be the focus on the machine learning.

2) How to eliminate the ambiguity of query terms, and further to define the purpose of user queries.

3) How to add agent technology into information retrieval is to inprove the efficiency of information retrieval.

4) Personalized query. In this thesis, the text information retrieval system is aim to specific field. It will to add user interest in order to achieve the personal query in the field.

ACKNOWLEDGMENT

We would like to thank the anonymous referees for providing us with constructive comments and suggestions.

REFERENCES

[1] P. Lambrix, and A. Edberg, "Evaluation of ontology development tools in bioinformatics," *Bioinformatics*, vol. 19, no.12, pp. 1564-1571, Jan 2003,.

[2] A. *Bouzeghoub*, and A. Elbyed, "Ontology mapping for web-based educational systems interoperability," *IBIS*, vol. 1, no.1, pp.73-84, May 2006,.

[3] X. Su, and J. A. Gulla, "Semantic enrichment for ontology mapping," Natural Language Processing and Information Systems Lecture Notes in Computer Science, 2004, Vol, 3136/2004, 21-44, DOI: 10.1007/978-3-540-27779-8_19

[4] T. Joachims, "Optimizing search engines using clickthrough data," in *Proc.The 8th ACM SIGKDD Intl Conference on Knowledge Discovery and Data Mining.* New York, USA: ACM Press, 2002, pp. 133-142.

[5] T. S. Chua, S.Y. Neo, and H.K. Goh,et al, "Trecvid 2005 by nus pris," NIST TRECVID, 2005. http://www.mingzhao.org /Publications/ZM_2005_TRECV

[6] C. Corinna, M. Mehryar, and R. Ashish, "Magnitude-Preserving Ranking Algorithms," *in Proc. The 24th International Conference on Machine Learning.* Corvallis, OR, USA, June, pp.169-176, 2007.

[7] C. David, and Z. Tong, "Subset Ranking Using Regression," COLT 2006, LNAI 4005, 2006, pp. 605-619.

[8] Y. Rong, Alexander, and D. Hauptmann, "Efficient margin-based rank learning *algorithms* for information retrieval," CIVR, pp.113-122, 2006.

[9] R. Cynthia, "*Ranking* with a P-Norm Push," COLT 2006, LNAI 4005, 2006, pp.589-604.

[10] S. Kutin, and P. Niyogi, "The interaction of stability and weakness in AdaBoost," Technical Report TR-2001-30, Computer Science Department, University of Chicago, 2001.

[11] S. Agarwal, *and* P. Niyogi, "Stability and generalization of bipartite ranking algorithms," in *proc. The 18th Annual Conference on Learning Theory,* Bertinoro, Italy, 27-30, Jun 2005.

[12] S. Agarwal, and P. Niyogi, "Generalization bounds for ranking algorithms via algorithmic stability," *Journal of Machine Learning Research,* vol.10, pp. 441-474, Jan 2009.

[13] R. Cynthia, "The *P*-Norm Push: A simple convex ranking algorithm that concentrates at the top of the list," *Journal of Machine Learning Research,* vol. 10, pp. 2233-2271, Jan 2009.

[14] O. Dekel, C. Manning, and Y. Singer, "Log-linear models for label ranking," In *NIPS* 16, 2004.

[15] G. Fung, R. Rosales, and B. Krishnapuram, "Learning rankings via convex hull separation," Neural Information Processing Systems - NIPS , 2005.

[16] H. B. Mann *and* D. R. Whitney, "On a Test of whether one of two random variables is stochastically larger than the other," Ann. Math. Stat., vol.18, no.1, pp. 50-60, 1947.

[17] V. C. Raykar and R. Duraiswami, . "A fast algorithm for learning large scale *preference* relations," in *Proc. The Eleventh International Conference on Artificial Intelligence and Statistics*, San Juan, Puerto Rico March, 2007.

[18] J. Nocedal *and* S. J. Wright, "Numerical Optimization," Springer-Verlag, 1999, pp.57-64.

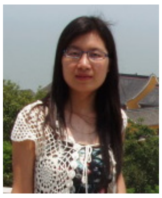
[19] N. C. Beauliu, "A *simple* series for personal computer computation of the error function Q(.)," *IEEE Trans. Comm.,* vol. 37, no. 9, pp. 989-991, May 1989.

[20] C. Yang, R. Duraiswami, and L. Davis, "Efficient kernel machines *using* the improved fast Gauss transform," In NIPS 17, 2005.

[21] V. C. Raykar, R. Duraiswami, and B. Krishnapuram. "Fast weighted summation of erfc functions," CS-TR- 4848, Dept. of Comp. Science, Univ. of Maryland CollegePark, 2007.

[22] M. Bklkin, P. Niyogi, "Laplacian eigenmaps for *dimensionality* reduction and data representation," *Neural Comput.* vol.15, pp. 1373-1396, Sec 2003.

[23] N. Craswell, and D. Hawking, "Overview of the TREC 2003 web track," in *Proc. The 12th Text Retrieval Conference. Gaithersburg,* Maryland, NIST Special Publication, 2003, pp. 78-92.



Xiao Huang, female, was born in Jinhua City, Zhejiang Province, China on July.30, 1978. She got bachelor degree on physics from the physics department of Zhejiang Normal University, 1999. Then, she got Master degree of physics education major from mathematics and Physics College of Zhejiang Normal University, 2002. Also, she got PhD of Science Education major from the East China Normal University in June, 2010. In 2008, she went to HKIED for cooperative research about RFID technology and education.

Act as an associate professor of Zhejiang Normal University; she is a physics teaching Committee member now. She has presided over several kinds of projects and published one book, about ten journal papers and several conference papers. Now she is principally interested in thermodynamics and statistical physics, the nature of science, integration information technology into science teaching.

Dr. Huang has won the Outstanding Graduate Scholarship and outstanding doctoral Fund of the East China Normal University, have got the title of outstanding instructor by the National Physical Education Committee. And she is also a member of IEEE and IACSIT.

Tianwei Xu, male, was born in Maguan City, Yunnan Province, China in 1970. He got bachelor degree on physics from the department of physics, Yunnan normal university, 1992. Then, he got Master degree of computer science major from the Yunnan Normal University in June, 2006. Now he is PhD student in the institute of higher education, Huazhong University of science and technology.

Act as a lecturer in the department of physics, Yunnan normal university, from July 1992 to June 1997. Then, act as associate secretary in the department of computer science and information technology, Yunnan normal university, from July 1997 to June 2003, and associate minister in the department of organization, Yunnan normal university, from July 2003 to June 2007. He also visited Curtin University of Technology in Austrian from Dec 2005 to June 2006. From 2007, he worked as secretary and associate professor in the department of Yunnan normal university, and now, he also a PHD student in the institute of higher education, Huazhong University of science and technology. He also acts as leader in the Key Laboratory of National Education Informatization, Ministry of Education, Yunnan Normal University, China.

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 form College of Zhejiang education in 2006. Then, he enrolled in department of computer science and information technology, Yunnan normal university, and got Master degree there in 2009. Now, he is PhD student in department of Mathematics, Soochow University, China.

During the school years in Soochow University, he also acts as lecturer in the department of information, Yunnan Normal University. As a researcher in computer science and mathematics, his interests are covered two disciplines: Graph theory, Statistical learning theory, Information retrieval, and artificial intelligence.

Zhiyang Jia, male, was born in the city of Jilin, Jilin Province, China on Jan.29, 1980. He got a Bachelor Degree of Mechanical Engineering from Harbin University of Science and Technology in 2003. He enrolled in department of computer science and information technology, Yunnan Normal University in 2006, and got Master Degree of Computer Software and Theory from there in 2009.

During 2003 to 2006, he had worked in China Academy of Machinery Science & Technology as an Information Engineer for three years. He entered Tourism and Literature College of Yunnan University which located in Lijing City of China in 2009, and acts as an Information Teacher from then on. His major fields of study are machine learning and knowledge management. Mr. Jia is a member of IEEE and ACM.