Ontology Algorithm Based on Semi-supervised Eigenmap Learning

Bo Peng, Tian-wei Xu, Zhen Li and Wei Gao∗

School of Information Science and Technology, Yunnan Normal University, Kumming 650500, China

Received: 20 Nov. 2012, Revised: 27 Mar. 2013, Accepted: 28 Mar. 2013
Published online: 1 May. 2013

Abstract: Ontology similarity calculation is an important research topic in information retrieval and it is also widely used in computer science. In this paper we propose new algorithms for ontology similarity measurement and ontology mapping using Semi-supervised Eigenmap Learning method. Via the 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. Two experimental results show that the proposed new algorithm has high accuracy and efficiency on ontology similarity calculation and ontology mapping.

Keywords: ontology, ontology mapping, semi-supervised learning, spectral clustering, graph Laplacian, similarity function

1 Clustering and Graph Laplacian

Clustering is an important topic in statistics, computer science and many other fields. Spectral clustering is a useful family of clustering algorithms which have been applied to some areas involving data analysis and processing. These algorithms are based on graph Laplacians which is a classical topic in spectral graph theory [2] and is introduced to learning theory in [1]. Their goal is to cluster data points according to the values on the data points of a function which is taken to be an eigenfunction of the graph Laplacian with the second smallest eigenvalue.

Let \( \mathcal{X} \) be a compact metric space which contains data points, and \( P \) be a probability measure on \( \mathcal{X} \). A similarity function \( k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_+ \) is symmetric, continuous and bounded away from 0 by a positive constant \( l \). It is used to measure similarity between different data points. Let \( \mathbf{x} = (x_i)_{i=1}^n \) be a random sample drawn independent according to the probability measure \( P \). Then the data similarity matrix is defined by \( K_n = [k(x_i, x_j)]_{i,j=1}^n \), and the degree matrix \( D_n \) is defined as the \( n \times n \) diagonal matrix with diagonal entries \( (D_n)_{i,i} = \sum_{j=1}^n k(x_i, x_j) \). The (normalized) graph Laplacian is defined as the \( n \times n \) matrix

\[
L_n = I - D_n^{-1/2} K_n D_n^{-1/2}.
\]

Observe that \( L_n \) has a trivial smallest eigenvalue 0 with eigenvector \( D_n^{1/2} [1, \ldots, 1]^T \) and it is positive semi-definite [1]. The “second smallest eigenvalue” refers to the smallest eigenvalue beyond the trivial eigenvalue 0, and a corresponding eigenvector may be used for clustering data. To see this, we need to introduce some linear operators on \( C(\mathcal{X}) \), the space of continuous functions on \( \mathcal{X} \) with norm \( \| \cdot \|_\infty \). Denote \( P_n := \frac{1}{n} \sum_{i=1}^n \delta_{x_i} \).

Let \( p_x := \int_{\mathcal{X}} k(\cdot, x) dP_n(y) = \frac{1}{n} \sum_{i=1}^n k(\cdot, x_i) \in C(\mathcal{X}) \) and \( p := \int_{\mathcal{X}} k(\cdot, y) dP(y) \in C(\mathcal{X}) \) be degree functions. Then the integral operator \( T : C(\mathcal{X}) \to C(\mathcal{X}) \) associated with \( (k, P) \) is defined by

\[
Tf(x) = \int_{\mathcal{X}} \frac{k(x, y)}{\sqrt{p(x) p(y)}} f(y) dP(y),
\]

where \( x \in \mathcal{X}, f \in C(\mathcal{X}) \). The empirical integral operator \( T_n : C(\mathcal{X}) \to C(\mathcal{X}) \) is defined by

\[
T_nf(x) = \int_{\mathcal{X}} \frac{k(x, y)}{\sqrt{p_x(x) p(y)}} f(y) dP_n(y)
\]

\[
= \frac{1}{n} \sum_{j=1}^n \frac{k(x, x_j)}{\sqrt{p_x(x) p(x_j)}} f(x_j).
\]
Note that \( p_k(x_i) = \frac{1}{n} \sum_{j=1}^{n} k(x_i, x_j) = \frac{1}{n} (D_k)_{i,j} \) and 
\[
\frac{1}{\sqrt{p_k(x_i)}} = \sqrt{n} (D_k^{-\frac{1}{2}})_{i,j}.
\]
Hence 
\[
[T_k f(x_i)]_{i=1}^{n} = \left[ \frac{1}{n} \sum_{j=1}^{n} \frac{k(x_i, x_j)}{\sqrt{p_k(x_i)} \sqrt{p_k(x_j)}} f(x_j) \right]_{i=1}^{n}.
\]
It follows that if \((\lambda_n, u_n)\) is an eigenpair of \(T_k\), \(u_n(x_i)\) is an eigenvector of the graph Laplacian \(L_k\) with eigenvalue \(1 - \lambda_n\).

Convergence of the spectral clustering considered in this paper means that of \(\{u_k\}_n\) (after normalization by \(\|u_k\|_2 = 1\)) to an eigenfunction \(u\) of \(T\) associated with a simple eigenvalue \(\lambda \neq 0\) if \(\{\lambda_n\}\) tends to \(\lambda\). This convergence was verified when \(k\) is a Gaussian similarity function \(k(x,y) = \exp\left(\frac{-\|x-y\|^2}{2\sigma^2}\right)\) on \(\mathcal{X} \subset \mathbb{R}^n\) in [5] and when \(k\) is a general Mercer kernel similarity function (with \(p > 0\) but without positivity of \(k\)) in [3, 2], both with convergence rate \(O\left(\frac{1}{n}\right)\). A general estimate was also presented in [5], but the convergence associated with a general similarity function \(k\) was not proved. The purpose of this paper is to verify the convergence of the spectral clustering associated with a general similarity function, and give quantitative estimates for the convergence. The convergence rates are stated in terms of the regularity of the similarity function \(k\) and the capacity of the input space \(\mathcal{X}\).

### 2 Ontology Measure and Ontology Mapping

The term of ontology comes from philosophy, it is used to describe the nature of things. Ontology similarity computation is widely used in medical science (see, for instance, [14] and [11]), biology science (see, for instance, [8]) and social science (see, for instance, [6]). Especially, in computer science, ontology is defined as a model for the sharing formal concepts have been applied in intelligent information integration, cooperative information systems, information retrieval, electronic commerce and knowledge management. As an effective model with hierarchical structure and semantics for the concepts, ontology technology has matured after nearly a decade of development. Now it has more systematic, comprehensive engineering theory, representation and construction tools. Particularly, in information retrieval, it has been used to compute semantic similarity (see [12]) and search extensions for concepts. Every vertex on an ontology graph represents a concept: A user searches for a concept \(A\), will return similarities concepts of \(A\) as search extensions to the user. Therefore, the quality of similarity functions play an important role in such applications. Moreover, ontology is also used in image retrieval (see, for instance, [7], [9], [10] and [13]).

When ontology is used in information retrieval, every vertex in ontology graph acts as a concept of ontology, the attraction of ontology graph can be used to measure the similarity of vertices. Let \(G\) be a graph corresponding to ontology \(O\), the goal of ontology similarity measure is to approach a similarity function which maps each pair of vertices to a real number. Choose the parameter \(M \in \mathbb{R}_+\), the concepts \(A\) and \(B\) have high similarity if \(Sim(A, B) > M\). The problem of ontology mapping can be described as follows: Let \(G_1, G_2, \ldots, G_k\) be graphs corresponding to ontologies \(O_1, O_2, \ldots, O_k\), respectively. Let \(G = G_1 + G_2 + \ldots + G_k\). For every vertex \(v \in V(G_i)\), where \(1 \leq i \leq k\), the goal of ontology mapping is to find similarity vertices from \(G - G_i\). Therefore, the ontology mapping problem is also an ontology similarity measure problem. Choose the parameter \(M \in \mathbb{R}_+\), let \(A, B\) be two concepts on ontology and \(Sim(A, B) > M\), then return \(B\) as retrieval expand when search concept \(A\).

### 3 New Ontology Algorithm

Very recently, Gao et al. [15, 16] posed new algorithm for ontology measure and ontology mapping based on graph Laplacian. By using regularization framework of graph, the optimized function is obtained. Thus, all vertices in ontology graph are mapped into real numbers. The ontology measure and ontology mapping is obtained by comparing the difference of the corresponding values.

However, in many cases, ontology are changeable from time to time. There often new vertices added in ontology graph which regard as new concepts, and some old vertices may be deleted. In this situation, new vertices may be unlabelled. Thus, we need new tricks to deal with changeable ontology setting. One method is using semi-supervised learning (SSL). It is a graph Laplacian regularization for semi-supervised Laplacian eigenmap learning (see [17]).

The distribution on input label pairs \((v, y)\) is unknown, where \(v \in \mathbb{R}^N\) and \(y \in \mathbb{R}\). The aim of SSL is to learn a best predictor \(f(v)\) for \(y\) with few labeled and unlabeled examples. It should be considered SSL in a transductive setting, where we are given a labeled sample \((v_1, y_1), \ldots, (v_n, y_n)\) of \(n\) labeled points, and an unlabeled sample \(v_{n+1}, \ldots, v_u\) of \(u - n\) unlabeled points. The \(u\) pairs \((v_i, y_i)\) are drawn i.i.d. from the source distribution. We use \(V_L\) to denote the sequence of labeled vertices in ontology graphs, \(V_U\) to denote the sequence of unlabeled vertices in ontology graphs, and \(V = V_L \cup V_U\). The label sequences are denoted as \(Y_L \in \mathbb{R}^n, Y_U \in \mathbb{R}^u\) and \(Y_U \in \mathbb{R}^u\), respectively. Transductive SSL is to estimate \(Y_U\) according to \(V_L, Y_L\) and \(V_U\). We write \(f(.)\) as a \(u\)-dimensional vector \(f(V) = (f(v_1), \ldots, f(v_n)) \in \mathbb{R}^u\).

For a weighted ontology graph \(G\) with \(u\) vertices corresponding to the unlabeled and labeled vertices \(v_1, \ldots, v_u\), and with edge weights \(w_{ij}\). Let \(W \in \mathbb{R}^{u \times u}\) denote the weight matrix and its diagonal degree matrix \(D = \sum w_{ij}\).
Let \( \hat{W} = D^{-1/2}WD^{-1/2} \) be a normalization of the weight matrix, and denote by \( \hat{D} \) its diagonal degree matrix \( \hat{D}_{ii} = \sum_j \hat{w}_{ij} \). Then

\[
\hat{L}_e = I - \hat{D}^{-1/2} \hat{W}.
\]

For a given weight matrix \( \hat{W} \), the unnormalized graph Laplacian defined as

\[
\hat{L}_u = \hat{D} - \hat{W}.
\]

and the symmetric normalized graph Laplacian

\[
\hat{L}_a = \hat{D}^{-1/2} \hat{L}_u \hat{D}^{-1/2} = I - \hat{D}^{-1/2} \hat{W} \hat{D}^{-1/2}.
\]

The eigenvectors of \( \hat{L}_e \) just as following the generalized eigenfunctions problem

\[
\hat{L}_e \pi_i = \lambda_i \hat{D} \pi_i,
\]

where \( \hat{L}_r = \hat{D}^{-1/2} \hat{L}_a \hat{D}^{-1/2} \). The eigenvectors of \( \hat{L}_e \) and the right eigenvectors of \( \hat{L}_a \) have a one to one mapping. If \( v_{ij} \) is a right eigenvector of \( \hat{L}_e \) with eigenvalue \( \lambda_i \), then \( v_{ij} = \hat{D}^{1/2} v_{ij} \) is an eigenvector of \( \hat{L}_a \) with the same eigenvalue \( \lambda_i \).

Given the labeled and unlabeled data for certain ontology graph, and a parameter \( t \), find the leading \( t \) right eigenvectors \( v_{1,1}, \ldots, v_{1,t} \) of \( \hat{L}_r \), with the smallest eigenvalues \( \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_t \). By the mapping \( v_i \rightarrow (v_{1,1}(v_i), v_{1,2}(v_i), \ldots, v_{1,t}(v_i)) \), where \( v_{1,j}(v_i) \) is the coordinate of \( v_{1,j} \), it can be performed an ordinary (unregularized) least squares regression in the \( t \)-dimensional space. Especially, the least squares predictor

\[
\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n} (y_i - \sum_{j=1}^{k} \beta_j v_{1,j}(v_i))^2
\]

and predict

\[
\hat{f}(v_i) = \sum_{j=1}^{k} \hat{\beta} v_{1,j}(v_i).
\]

The semi-supervised eigemap learning algorithm can be used in ontology concepts similarity measurement. The basic idea is: via the semi-supervised eigemap 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.

Algorithm for ontology similarity measure: For \( v \in V(G) \), we use one of the following methods to obtain the similar vertices and return the outcome to the users.

Choose a parameter \( M \) return set \( \{u \in V(G), |f(u) - f(v)| \leq M \} \).

Algorithm for ontology mapping: For \( v \in V(G_i) \), where \( 1 \leq i \leq k \), we use one of following methods to obtain the similar vertices and return the outcome to the users.

Choose a parameter \( M \), return set \( \{u \in V(G - G_i), |f(u) - f(v)| \leq M \} \).

4 Experiments

To connect ontology to this semi-supervised eigemap learning algorithm, we should use a vector to express each vertex of information. This vector contains the information of name, instance, attribute and structure of the vertex, where the instance of vertex is the set of its reachable vertices in the directed ontology graph.

The first experiment concerns ontology similarity measurement is described as follows. In this experiment we use computer ontology \( O_1 \) which was constructed in [20]. The goal of the algorithm is to map the vertices on the 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 obtained are indirect similarity measure not the direct ones. We use \( P@N \) (Precision Ratio see [21]) to measure the equality of the experiment. First the expert gives the first \( N \) concepts for every vertex on the ontology graph then we obtain the first \( N \) concepts for every vertex on ontology graph by the algorithm and compute the precision ratio.

The experiment shows that, \( P@1 \) Precision Ratio is 0.39, \( P@3 \) Precision Ratio is 0.55, \( P@5 \) Precision Ratio is 0.70. Thus the proposed algorithm has high efficiency.

For the second experiment we use another Computer Ontologies \( O_2 \) which constructed in [20] shows \( O_2 \). The goal of the algorithm is to map 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.

The experiment shows that, \( P@1 \) Precision Ratio is 0.33, \( P@3 \) Precision Ratio is 0.48, \( P@5 \) Precision Ratio is 0.63. Thus the algorithm has high efficiency.

5 Conclusion

In this paper, we give a new algorithm for measuring the ontology similarity and ontology mapping using semi-supervised eigemap learning. The new algorithms have high quality according to the experiments above.

Acknowledgement

This work was supported in part by Key Laboratory of Educational Informatization for Nationalities, Ministry of Education, the National Natural Science Foundation of China (60903131) and Key Science and Technology Research Project of Education Ministry (210210). The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References


