

RESEARCH

Open Access



Approximation analysis of ontology learning algorithm in linear combination setting

Wei Gao^{1*†} and Yaojun Chen^{2†}

Abstract

In the past ten years, researchers have always attached great importance to the application of ontology to its relevant specific fields. At the same time, applying learning algorithms to many ontology algorithms is also a hot topic. For example, ontology learning technology and knowledge are used in the field of semantic retrieval and machine translation. The field of discovery and information systems can also be integrated with ontology learning techniques. Among several ontology learning tricks, multi-dividing ontology learning is the most popular one which proved to be in high efficiency for the similarity calculation of tree structure ontology. In this work, we study the multi-dividing ontology learning algorithm from the mathematical point of view, and an approximation conclusion is presented under the linear representation assumption. The theoretical result obtained here has constructive meaning for the similarity calculation and concrete engineering applications of tree-shaped ontologies. Finally, linear combination multi-dividing ontology learning is applied to university ontologies and mathematical ontologies, and the experimental results imply that the higher efficiency of the proposed approach in actual applications.

Keywords: Ontology, Learning algorithm, Multi-dividing setting, Linear combination

Introduction

The term “ontology” was originally used in the field of philosophy, meaning “the essence of things, itself”, and the abstract nature of real things. Others describe ontology as: “Ontology defines the relevant terms in the field, associations and rules of domain lexical epitaxy”. Ontology theory was first introduced into the field of computer artificial intelligence. Later, the ontology was widely used in the computer field, and more and more domain experts defined the ontology. At present, the most extensive and most popular ontology is defined as “the explicit formal specification of sharing conceptualization”, whose meaning includes four aspects:

- sharing, means people agree on the expression of ontology;
- conceptualization, means that ontology is an abstract expression of the real world;
- explicit, means concepts and conceptual relationships are accurately and clearly defined;
- formal, means that concepts and relationships in ontology are described as machine-recognizable forms.

Domain ontology is an ontology type that is developed through ontology research, and the ontology is classified according to the degree of domain dependence. Specifically, ontologies can be divided into four categories: top-level ontology, domain ontology, task ontology, application ontology. Since ontology can be regarded as a structured collection of concepts, the inter-relationship between concepts and the structural features of ontology

*Correspondence: gaoweiy@ynnu.edu.cn

[†]Wei Gao and Yaojun Chen contributed equally to this work.

¹School of Information Science and Technology, Yunnan Normal University, 650500 Kunming, China

Full list of author information is available at the end of the article



© The Author(s). 2020 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

are the essential problems of various applications of ontology, and thus the semantic similarity calculation between ontology concepts becomes the core of different ontology algorithms.

As a hot topic in the field of computer science and information technology, ontology algorithms have always been the key of data retrieval, image analysis, and applied to many frontier fields such as big data, Internet of Things, and deep learning. Subramaniaswamy et al. [1] provided a personalized food recommender system in IoT-based healthcare system in which ontologies are used to bridge the gap between descriptions and heterogeneous user profiles. Mohammadi et al. [2] examined the ontology alignment systems using statistical inference where some mathematical tricks like Wilcoxon signed-rank and asymptotic tests are recommended based on their statistical safety and robustness in different settings. Morente-Molinera et al. [3] suggested a trick that uses sentiment analysis procedures to automatically obtain fuzzy ontologies, and also multi-granular fuzzy linguistic modelling are employed to choose the optimal representation mean to store the information in fuzzy ontology. Sacha et al. [4] raised an ontology VIS4ML to describe and understand existing VA workflows used in machine learning. Kamsu-Foguem et al. [5] pointed out that different flows and their combinations can be dealt with by means of semantic Web concepts and conceptual graph theories which permit rules to be imbued to improve reasoning. Janowicz et al. [6] focus on how to fully use SOSA, including integration with the new release of the SSN ontology. Sulthana and Ramasamy [7] proposed a neuro-fuzzy classification trick in light of fuzzy rules, and ontology facilitates a systematic and hierarchical methodology to manage the context. To deal with syntactic evolution in the sources, Nadal et al. [8] introduced a technology that the ontology upon new releases are adapted semi-automatically. Scarpato et al. [9] presented the reachability matrix ontology to describe the networks and the cybersecurity domain, and then to compute the reachability matrix. Karimi and Kamandi [10] raised Inductive Logic Programming (ILP) based ontology learning algorithm and used it to solve the ontology mapping problem.

Various ontology algorithms are widely employed in different engineering fields. Koehler et al. [11] introduced the expansion of the Human Phenotype Ontology (HPO). Chhim et al. [12] presented an efficacious product design and manufacturing process based ontology for manufacturing knowledge reuse. Ali et al. [13] manifested a consensus-based Additive Manufacturing Ontology (AMO) and presented how to use it for promoting reusability in dentistry product manufacturing. Neveu et al. [14] proposed open-source Phenotyping Hybrid Information System (PHIS) with its ontology-driven architecture for building relationships between objects and enriching

datasets with knowledge and metadata. Kiefer and Hall [15] updated gene ontology analysis for stimulate further research and possible treatment. Jaervenpaeae et al. [16] described the systematic development process of an OWL-based manufacturing resource capability ontology and its capabilities of manufacturing resources. Serra et al. [17] demonstrated a proof of concept for leveraging the built-in logical axioms of the ontology to classify patient surface marker data into appropriate diagnostic classes. Di Noia et al. [18] proposed to structure the knowledge associated with NFRs in terms of fuzzy ontology for tool-supported decision making in architectural design. Ledvinka et al. [19] determined the implementation of an ontology-based information system for aviation safety data integration. Aameri et al. [20] raised an ontology to specify shapes, parthood and connection in mechanical assemblies such that the constraints of feasible configurations can be logically expressed and used during generative design.

With the proliferation of ontology processing concepts, machine learning algorithms are applied to ontology similarity calculations (some ontology learning tricks can be referred to Gao et al. [21–23] and [24]). Among them, the ontology learning algorithm in multi-dividing setting has proved to be more efficient for the similarity calculation under the tree-shaped ontology structures (see Gao et al. [25], Gao and Farahani [26], Wu et al. [27], Sangaiah et al. [28] and Gao and Xu [29] for more details). Due to the engineering accuracy of the multi-dividing ontology learning algorithm has confirmed by different ontology applications, this paper no longer gives the experimental results of the algorithm under special ontology data, but from the statistical point of view. The approximation property of the multi-dividing ontology learning algorithm in a special expression setting is given.

In recent years, cloud computing has received widespread attention, and the number and types of cloud services it provides been increasing year by year (see Bryce [30] and Song et al. [31]). Scholars are considering how to quickly find the cloud services that users need and effectively provide them to users (see Dimitri [32]). The traditional cloud service is based on the search of keywords, and the query results will contain irrelevant information. At the same time, due to the defect of keyword matching, it is easy to miss related services, which is mainly because the traditional cloud query services don't have keywords concept expand query function. In order to solve the above-mentioned problems in existing cloud services, cloud ontology-based semantic networks can provide users with more accurate cloud services for different needs information. Therefore, applying ontology to cloud computing and cloud services is definitely worth of looking forward to the future, which encourages us to design specific ontology algorithms in cloud ontology

according to specific cloud service requirements (see Sangaiah et al. [33] and [34]).

The rest of paper is organised as follows: first we introduce the setting of ontology learning and in particular multi-dividing ontology learning; then the main theoretical result and its detailed proof are determined; finally, we manifest two experiments on university and mathematical ontology data to demonstrate the efficiency of the algorithm.

Ontology learning problem

Throughout our paper, we use a graph to represent the structure of the ontology. The vertices in the graph represent the concept of the ontology, while the edges between the vertices express a direct subordinate relationship (or inclusion relationship, affiliated relationship) between the two concepts. In order to facilitate the mathematical representation of ontology learning setting, we need to do some processing and specification on the ontology data in the beginning stage, which has met the requirements of the later mathematical expression.

First of all, we numerically denote the semantic information, knowledge background, structure, instance, attribute and classification information corresponding to a concept, and then encapsulate it in a fixed-dimensional vector. Through a certain technical means, we can unify the dimensions of the vectors corresponding to the vertices of all ontology concepts, and specify the same type of information to be expressed of the corresponding sequence number of components in the vectors. In this way, the ontology information is represented by the corresponding vector space, and thus the processing and calculation of ontology data can be transformed into the processing and calculation of multi-dimensional vectors. In what follows, on the premise of not causing confusion, in order to facilitate the representation, we use $v = (v_1, \dots, v_p)$ (assume $v \in \mathbb{R}^p$) to simultaneously represent the ontology concept, the vertex in the ontology graph corresponding to the concept, and the vector corresponding to this vertex.

As a conceptual model, the main task of ontology is to manage concepts and information mining. Therefore, the similarity calculation between concepts is the core of ontology application in various engineering fields. Specifically, for the ontology vertices v_1 and v_2 , it is necessary to characterize the measurement of $sim(v_1, v_2)$. Since the vertices are denoted by vectors, the similarity between vertices can be regarded as the similarity between two vectors in high-dimensional space.

A learning technique based on the dimension descent method is to map each ontology vector into a real number, thereby mapping the entire ontology graph to the real axis, and the similarity between the ontology vertices is obtained by their one-dimensional distance on the real axis. Specifically, let $f : \mathbb{R}^p \rightarrow \mathbb{R}$ be an ontology function

that maps ontology concept vectors into real numbers. The similarity between the two ontology vertices v_1 and v_2 is measured by $|f(v_1) - f(v_2)|$, and the larger the value is, the smaller the similarity between v_1 and v_2 becomes, and on the contrary, the smaller the value of $|f(v_1) - f(v_2)|$, the larger the similarity between two vertices.

Therefore, in the standard ontology learning setting, the ontology procedure can be described as follows. Let $V \subset \mathbb{R}^p$ ($p \geq 1$) be an instance space (or called the input space) for ontology graph, and the vertices in V are drawn independently and randomly follow to a certain unknown distribution \mathcal{D} . The aim of ontology learning algorithms is to deduce an optimal ontology function $f : V \rightarrow \mathbb{R}$ using the given ontology training set $S = \{v_1, \dots, v_n\}$ of size n .

Multi-dividing ontology setting

This framework of multi-dividing ontology learning algorithm is based on the fact that most ontology graph structures are tree structures (acyclic graphs). Forming several branches of the tree below the topmost vertex, if we classify all ontology concepts using a classification algorithm, we find that the vertices in each branch correspond exactly to a class of vertex classification. It means that the similarity between the vertices of the same branch is higher than the similarity between the vertices from different branches. After mapping all ontology vertices to the real axis, it can be observed that the real numbers corresponding to the vertices of the same branch have an aggregation effect on the one-dimensional axis (it can be understood that the vertices of each branch form a one-dimensional cluster on the axis). In light of this observation, we have reason to assume that the real numbers corresponding to the same branch vertices are in the same interval of the real axis. We can imagine cutting the entire real axis into k -breaks (here k represents the number of branches under the top vertex of the ontology graph), with all the vertices of each branch happening within a certain break. In the following contexts, we always assume k, a, b are positive integers.

Now, we formally describe the multi-dividing ontology algorithm. All the ontology vertices are divided into k parts which is corresponding to k branches in the ontology graph, and we endow rate number of these k parts of vertices, denoted by $1, 2, \dots, k$ (note that the rate values of all parts are determined by domain experts who have deep domain knowledge related to ontology in certain engineering application). Assume that $f(v^a) > f(v^b)$ where f is an ontology function, v^a belongs to rate a vertices, v^b belongs to rate b vertices, and $1 \leq a < b \leq k$. It reveals that under the target ontology function, the value of high rate vertex is bigger than the value of low rate vertex.

Correspondingly, the ontology training sample in multi-dividing ontology setting is denoted as $S = (S_1, S_2, \dots, S_k) \in V^{n_1} \times V^{n_2} \times \dots \times V^{n_k}$ which consists of

a sequence of training sample $S_a = (v_1^a, \dots, v_{n_a}^a)$ belongs to rate a (here $1 \leq a \leq k$). The ontology learner is given such an ontology sample S and aim to learn a real-valued ontology score function $f : V \rightarrow \mathbb{R}$ (or $f : \mathbb{R}^p \rightarrow \mathbb{R}$) that the value of S_a vertices are bigger than the value of S_b vertices if $1 \leq a < b \leq k$. Suppose that vertices in each S_a (here $1 \leq a \leq k$) are drawn independently and randomly according to certain unknown distribution \mathcal{D}_a on the instance space V respectively. On the other hand, since each vertex v_i^a or v_j^b is a p -dimensional vector, we set

$$v_i^a = ((v_i^a)_1, \dots, (v_i^a)_p), v_j^b = ((v_j^b)_1, \dots, (v_j^b)_p) \text{ with } i \in \{1, \dots, n_a\} \text{ and } j \in \{1, \dots, n_b\}.$$

Let $I(\cdot)$ be the binary truth function (it also named 0-1 function or 0-1 loss). Then, the ontology learning algorithm in area under the receiver operating characteristic curve criterion can be formulated by

$$\widehat{A}(f, S) = \sum_{a=1}^{k-1} \sum_{b=a+1}^k \frac{1}{n_a n_b} \sum_{i=1}^{n_a} \sum_{j=1}^{n_b} I(f(v_i^a) > f(v_j^b)). \quad (1)$$

Here we need to explain and state the following points:

- (1) The optimal ontology function is obtained by maximizing $\widehat{A}(f, S)$.
- (2) The standard multi-dividing learning algorithm can be stated as

$$\widehat{A}(f, S) = \sum_{a=1}^{k-1} \sum_{b=a+1}^k \frac{1}{n_a n_b} \sum_{i=1}^{n_a} \sum_{j=1}^{n_b} \left\{ I(f(v_i^a) > f(v_j^b)) + \frac{1}{2} I(f(v_i^a) = f(v_j^b)) \right\}.$$

Clearly, our ontology framework omitted the $\frac{1}{2} I(f(v_i^a) = f(v_j^b))$ part in each of the accumulated items.

- (3) The expected ontology model of (1) is denoted as

$$A(f) = \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{E}_{V_a \sim \mathcal{D}_a, V_b \sim \mathcal{D}_b} I(f(V_a) > f(V_b)). \quad (2)$$

Remember that in the ontology sparse vector setting, the ontology function can be concrete represented as

$$f(v) = \sum_{t=1}^p v_t \beta_t + \beta_0 = v \beta^T + \beta_0, \quad (3)$$

where $\beta = (\beta_1, \dots, \beta_p)$ is ontology sparse vector such that most of its components are supposed to be zero, and β_0 is an offset item. In many circumstances, we ignore β_0 and consider $f(v) = \sum_{t=1}^p v_t \beta_t$. It's general expand expression can be stated as $f(v) = \sum_{t=1}^p g_t(v_t)$, where g_t is some kind of function (obviously, in the very special case of ontology sparse vector setting, $g_t(v_t) = v_t \beta_t$).

Return to the standard framework with offset term, the general ontology model (3) can be written by

$$f(v) = \sum_{t=1}^p g_t(v_t) + \beta_0. \quad (4)$$

In this paper, we consider the linear combination setting where function g_t can be formulated as $g_t(\cdot) = \sum_{q=1}^d \beta_{tq} \phi_q(\cdot)$ where $\phi_q(\cdot)$ are basis functions from (4). Set

$$\begin{aligned} \Delta(v_i^a, v_j^b) &= f(v_i^a) - f(v_j^b) \\ &= \sum_{t=1}^p (g_t((v_i^a)_t) - g_t((v_j^b)_t)) \\ &= \sum_{t=1}^p \sum_{q=1}^d \beta_{tq} (\phi_q((v_i^a)_t) - \phi_q((v_j^b)_t)) \end{aligned} \quad (5)$$

as the difference between value of two ontology functions of v_i^a and v_j^b . Thus, the expected version can be re-stated as

$$A(f) = \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}(\Delta(V_a, V_b) > 0), \quad (6)$$

where $V_a \sim \mathcal{D}_a$ belongs to rate a and $V_b \sim \mathcal{D}_b$ belongs to rate b . Once a special combination of (a, b) is fixed, then denote $A^{a,b}(f) = \mathbb{P}(\Delta(V_a, V_b) > 0)$.

Accordingly, the ontology empirical framework with ontology sample set $S = (S_1, S_2, \dots, S_k) \in V^{n_1} \times V^{n_2} \times \dots \times V^{n_k}$ is re-formulated by

$$\widehat{A}(f, S) = \sum_{a=1}^{k-1} \sum_{b=a+1}^k \frac{1}{n_a n_b} \sum_{i=1}^{n_a} \sum_{j=1}^{n_b} I(\Delta(v_i^a, v_j^b) > 0). \quad (7)$$

It is not hard to verify that $\widehat{A}(f, S)$ is an unbiased estimator of $A(f)$, that is to say, $\mathbb{E}[\widehat{A}(f, S)] = A(f)$. Therefore, the ontology risk and ontology empirical risk are defined as $\binom{k}{2} - A(f)$ and $\binom{k}{2} - \widehat{A}(f, S)$, respectively.

Our main result will be presented in next section which characterizes the convergence property under condition that each n_i is a big number (here $i \in \{1, \dots, k\}$) in the multi-dividing ontology setting.

Main result and proof

In this section, we manifest our result and the detailed proof is based on the statistical skills.

Hypothesis space is an important factor in statistical learning theory. The ontology algorithm can't converge if the space is too large, while the resulting optimal ontology function doesn't have excellent statistical properties if the hypothesis space is too small. A crucial point in the proof technique is to control the measure of the hypothesis space to achieve a certain degree of balance. Here, we set

$$\mathcal{F}_d = \{f : f(v) = \sum_{t=1}^p g_t(v_t) = \sum_{t=1}^p \sum_{q=1}^d \beta_{tq} \phi_q(v_t)\} \quad (8)$$

as hypothesis space in our setting where $\phi_q(\cdot)$ with $q \in \{1, \dots, d\}$ are basis functions.

For each pair of (a, b) with $1 \leq a < b \leq k$, set $\frac{n_a}{n_b} \rightarrow c^{a,b}$. Our main result is stated as follows.

Theorem 1 Assume $c^{a,b} > 0$ for each pair of (a, b) with $1 \leq a < b \leq k$ and $\sum_{n_a=1}^\infty n_a^{2dp} \exp\{-\frac{n_a \varepsilon^2}{8}\} < \infty$ for any $a \in \{1, \dots, k-1\}$ and $\varepsilon > 0$, then

$$|\sup_{f \in \mathcal{F}_d} \widehat{A}(f, S) - \sup_{f \in \mathcal{F}_d} A(f)| \rightarrow 0 \quad (9)$$

holds almost everywhere.

Proof of Theorem 1. Our proof techniques depend heavily on Hoeffding inequality, Borel-Cantelli lemma and statistical property of shatter coefficient. For any combination (a, b) with $1 \leq a < b \leq k$, set $A_i^{a,b} = \frac{1}{n_b} \sum_{j=1}^{n_b} I(\Delta(v_i^a, v_j^b) > 0)$ where $i \in \{1, \dots, n_a\}$. Hence, the ontology empirical version can be re-written as

$$\widehat{A}(f, S) = \sum_{a=1}^{k-1} \sum_{b=a+1}^k \frac{1}{n_a} \sum_{i=1}^{n_a} A_i^{a,b}.$$

Notice that $A_i^{a,b}$ are all independent with $A_i^{a,b} \in [0, 1]$ for any combination (a, b) with fixed $v_1^b, \dots, v_{n_b}^b$, where $1 \leq a < b \leq k$. Thus, in terms of Hoeffding Theorem, we infer

$$\begin{aligned} & \mathbb{P}(|\widehat{A}(f, S) - A(f)| > \varepsilon) \\ &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}\left(|\frac{1}{n_a} \sum_{i=1}^{n_a} A_i^{a,b} - A^{a,b}(f)| > \varepsilon\right) \\ &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}\left(|\sum_{i=1}^{n_a} (A_i^{a,b} - A^{a,b}(f))| > n_a \varepsilon\right) \\ &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}\left(|\sum_{i=1}^{n_a} (A_i^{a,b} - \mathbb{E}(A_i^{a,b} | v_1^b, \dots, v_{n_b}^b))\right. \\ & \quad \left.+ \sum_{i=1}^{n_a} (\mathbb{E}(A_i^{a,b} | v_1^b, \dots, v_{n_b}^b) - A^{a,b}(f))| > n_a \varepsilon\right) \\ &\leq \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}\left(|\sum_{i=1}^{n_a} (A_i^{a,b} - \mathbb{E}(A_i^{a,b} | v_1^b, \dots, v_{n_b}^b))| \geq \frac{n_a \varepsilon}{2}\right) \\ & \quad + \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}\left(|\sum_{i=1}^{n_a} (\mathbb{E}(A_i^{a,b} | v_1^b, \dots, v_{n_b}^b) - A^{a,b}(f))| > \frac{n_a \varepsilon}{2}\right) \\ &= (I) + (II). \end{aligned}$$

For the first part (I), in view of Hoeffding inequality, we deduce

$$\begin{aligned} (I) &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}\left(\left|\sum_{i=1}^{n_a} (A_i^{a,b} - \mathbb{E}(A_i^{a,b} | v_1^b, \dots, v_{n_b}^b))\right| \geq \frac{n_a \varepsilon}{2}\right) \\ &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{E}\left\{\mathbb{P}\left(\left|\sum_{i=1}^{n_a} (A_i^{a,b} - \mathbb{E}(A_i^{a,b} | v_1^b, \dots, v_{n_b}^b))\right| \geq \frac{n_a \varepsilon}{2} \mid v_1^b, \dots, v_{n_b}^b\right)\right\} \\ &\leq \sum_{a=1}^{k-1} \sum_{b=a+1}^k 2\mathbb{E}\left\{\exp\left\{-\frac{n_a \varepsilon^2}{8}\right\} \mid v_1^b, \dots, v_{n_b}^b\right\} \\ &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k 2 \exp\left\{-\frac{n_a \varepsilon^2}{8}\right\}. \end{aligned}$$

Using the same fashion, the second part (II) can be similar bounded to

$$\begin{aligned} (II) &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \mathbb{P}\left(\left|\sum_{i=1}^{n_a} (\mathbb{E}(A_i^{a,b} | v_1^b, \dots, v_{n_b}^b) - A^{a,b}(f))\right| > \frac{n_a \varepsilon}{2}\right) \\ &\leq \sum_{a=1}^{k-1} \sum_{b=a+1}^k 2 \exp\left\{-\frac{n_a \varepsilon^2}{8}\right\}. \end{aligned}$$

By combining above two parts, we yield

$$\begin{aligned} \mathbb{P}(|\widehat{A}(f, S) - A(f)| > \varepsilon) &\leq (I) + (II) \\ &\leq 4 \sum_{a=1}^{k-1} \sum_{b=a+1}^k \exp\left\{-\frac{n_a \varepsilon^2}{8}\right\}. \end{aligned} \quad (10)$$

In terms of $\sum_{n_a=1}^\infty \mathbb{P}(|\widehat{A}(f, S) - A(f)| > \varepsilon) < \infty$ and Borel-Cantelli lemma, we know that $|\widehat{A}(f, S) - A(f)| \rightarrow 0$ holds almost everywhere. Since the VC dimension of $\mathcal{F}_d = \{f : f = \beta_0 + \sum_{t=1}^p \sum_{q=1}^d \beta_{tq} \phi_q\}$ is $dp + 1$ and there are $\sum_{a=1}^{k-1} \sum_{b=a+1}^k n_a n_b$ U-statistic kind of observations, $\Delta(v_i^a, v_j^b)$ for $1 \leq a < b \leq k, i \in \{1, \dots, n_a\}$ and $j \in \{1, \dots, n_b\}$, the shatter coefficient (the standard definition of shatter coefficient in multi-dividing ontology setting can be stated as the same as it in the k -partite ranking setting, for more details see Gao and Wang [35]) of the linear ontology function space \mathcal{F}_d can be bounded by $\sum_{a=1}^{k-1} \sum_{b=a+1}^k \{2 + 2(n_a n_b - 1)^{dp}\} \leq \sum_{a=1}^{k-1} \sum_{b=a+1}^k 3(n_a n_b)^{dp}$. In light of (10) and this upper bound of shatter coefficient, we verify that

$$\begin{aligned} & \mathbb{P}\left(\sup_{f \in \mathcal{F}_d} |\widehat{A}(f, S) - A(f)| > \varepsilon\right) \\ & \leq 12 \sum_{a=1}^{k-1} \sum_{b=a+1}^k (n_a n_b)^{pd} \exp\left\{-\frac{n_a \varepsilon^2}{8}\right\}. \end{aligned} \quad (11)$$

According to the assumption in theorem that $\frac{n_a}{n_b} \rightarrow c^{a,b} > 0$ with $1 \leq a < b \leq k$, we see that for each pair of (a, b) , n_a increases at the same rate of n_b . It indicates that when the ontology vertex number is increasing, the number of vertices in each branch will grow relatively evenly (in graph theory, such structure called nearly balanced tree). Thus, if $\frac{n_a}{n_b} \rightarrow c^{a,b} > 0$, then $\sum_{a=1}^{k-1} \sum_{b=a+1}^k \sum_{n_a, n_b} (n_a n_b)^{pd} \exp\left\{-\frac{n_a \varepsilon^2}{8}\right\} < \infty$ is equivalent to $\sum_{a=1}^{k-1} \sum_{b=a+1}^k \sum_{n_a} n_a^{2dp} \exp\left\{-\frac{n_a \varepsilon^2}{8}\right\} < \infty$ which is acted as known condition. Therefore, by means of $\frac{n_a}{n_b} \rightarrow c^{a,b} > 0$ for each pair of (a, b) with $1 \leq a < b \leq k$ and Borel-Cantelli Lemma, we confirm that

$$\sum_{f \in \mathcal{F}_d} \mathbb{P}(\sup |\widehat{A}(f, S) - A(f)| > \varepsilon) < \infty \tag{12}$$

and also derive

$$\sup_{f \in \mathcal{F}_d} |\widehat{A}(f, S) - A(f)| \rightarrow 0 \tag{13}$$

holds almost everywhere.

Finally, we need to show that $\sup_{f \in \mathcal{F}_d} \widehat{A}(f, S) - \sup_{f \in \mathcal{F}_d} A(f) \rightarrow 0$ almost surely. Set

$$\widehat{f}^* = \arg \max_{f \in \mathcal{F}_d} \widehat{A}(f, S)$$

and

$$f^* = \sup_{f \in \mathcal{F}_d} A(f).$$

Hence, the corresponding area under the receiver operating characteristic curve criterion are stated as

$$\sup_{f \in \mathcal{F}_d} \widehat{A}(f, S) = \widehat{A}(\widehat{f}^*, S)$$

and

$$\sup_{f \in \mathcal{F}_d} A(f) = A(f^*).$$

Combining all these facts, we get

$$\begin{aligned} & \left| \sup_{f \in \mathcal{F}_d} \widehat{A}(f, S) - \sup_{f \in \mathcal{F}_d} A(f) \right| \\ &= \left| \widehat{A}(\widehat{f}^*, S) - A(f^*) \right| \\ &\leq \left| \widehat{A}(\widehat{f}^*, S) - A(\widehat{f}^*) \right| + \left| A(\widehat{f}^*) - A(f^*) \right|. \end{aligned} \tag{14}$$

By means of $A(\widehat{f}^*) \leq A(f^*)$, the second term in (14) can be decomposed into two terms and then dealt with as follows

$$\begin{aligned} & \left| A(\widehat{f}^*) - A(f^*) \right| \\ &= A(f^*) - A(\widehat{f}^*) \\ &= A(f^*) - \widehat{A}(\widehat{f}^*, S) + \widehat{A}(\widehat{f}^*, S) - A(\widehat{f}^*) \\ &\leq A(f^*) - \widehat{A}(\widehat{f}^*, S) + \widehat{A}(\widehat{f}^*, S) - A(\widehat{f}^*) \\ &\leq 2 \sup_{f \in \mathcal{F}_d} |\widehat{A}(f, S) - A(f)|. \end{aligned} \tag{15}$$

From (13), (14) and (15), we obtain

$$\begin{aligned} & \left| \sup_{f \in \mathcal{F}_d} \widehat{A}(f, S) - \sup_{f \in \mathcal{F}_d} A(f) \right| \\ &\leq \left| \widehat{A}(\widehat{f}^*, S) - A(\widehat{f}^*) \right| + \left| A(\widehat{f}^*) - A(f^*) \right| \\ &\leq \left| \widehat{A}(\widehat{f}^*, S) - A(\widehat{f}^*) \right| + 2 \sup_{f \in \mathcal{F}_d} |\widehat{A}(f, S) - A(f)| \\ &\leq 3 \sup_{f \in \mathcal{F}_d} |\widehat{A}(f, S) - A(f)| \rightarrow 0 \end{aligned}$$

holds almost everywhere. \square

Theorem 1 still establishes even if d and p are admitted to increase relying on the ontology sample capacity. In this case, the hypothesis space \mathcal{F}_d increases as d and p grows. However, the rates of d and p rely on the combination of (n_1, \dots, n_k) , which implies that the dimension p has slower increasing rate than ontology sample capacities. Theorem 1 reveals that if p is much larger than ontology sample capacities, then the ontology empirical risk minimization algorithm framework may not obtain a desired performance. To our delight, in certain structural assumption settings (for instance, sparsity), we have enough reasons to construct an optimal ontology rule if some offset terms are eliminated when p is much larger than ontology sample capacities.

Since 0-1 ontology loss is a non-derivable function, it is difficult minimizing $\binom{k}{2} - \widehat{A}(f, S)$ in practice, and it is natural to apply some approximation tricks based on the smooth ontology function such as logistic ontology function $\Lambda_\tau(x) = \frac{\exp\{-x\tau\}}{1 + \exp\{-x\tau\}}$ where τ is a positive number which is used to control how steep the logistic ontology function is around zero. In light of $\Lambda_\tau(x)$, we consider the $\Phi(f)$ approximation to $\binom{k}{2} - \widehat{A}(f, S)$ in the multi-dividing ontology setting, i.e.,

$$\begin{aligned} \Phi(f) &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \frac{1}{n_a n_b} \sum_{i=1}^{n_a} \sum_{j=1}^{n_b} \Lambda_\tau \left(\Delta(v_i^a, v_j^b) \right) \\ &= \sum_{a=1}^{k-1} \sum_{b=a+1}^k \frac{1}{n_a n_b} \sum_{i=1}^{n_a} \sum_{j=1}^{n_b} \frac{\exp\left\{-\Delta(v_i^a, v_j^b)\tau\right\}}{1 + \exp\left\{-\Delta(v_i^a, v_j^b)\tau\right\}} \end{aligned}$$

for some positive number τ . To Minimize $\Phi(f)$ in \mathcal{F}_d is equivalent to that in terms of $\gamma \in \mathbb{R}^{p \times d}$, and thus it can be re-formulated as $\Phi(f) = \Phi(\gamma)$ where $\gamma = (\gamma_1, \dots, \gamma_p)$ and $\gamma_t = (\gamma_{t1}, \dots, \gamma_{td})$ for $t \in \{1, \dots, p\}$. Therefore, the minimizer of $\Phi(\gamma)$ can be numerically identified since $\Phi(\gamma)$ is the smooth function of γ . Theorem 1 indicates that the optimization of 0-1 ontology loss is ensure to yield the best character with certain restrictions on ontology sample dimensions and capacities. But, generally speaking, it is not practical to deduce the solution of minimizing 0-1 ontology loss, in particular in high dimensional settings. In all, it's a sensible and popular way to apply a smooth ontology function to approximate 0-1 ontology

loss which is thought to be an intelligent approximation although its optimality of the approximation is still unknown.

Now, we elaborately explain the weak points of linear combination multi-dividing ontology learning algorithm, that is, under what circumstances it is not suitable.

- It can be seen from the ontology learning models (1), (2), (6), (7) that the technology to achieve dimensionality reduction comes from the pairwise comparison of the ontology sample vertices. The weakness is that only two ontology vertices can be extracted at one time for comparison, which results in the number of vertex pairs to be compared in the optimization model becoming very large as the sample capacity increases.
- The depth of the vertex in the ontology graph is defined as the distance between this vertex and the top vertex, and the depth of the ontology graph is generally defined as the depth of the deepest vertex. As the depth increases, the concept of the ontology will become more and more detailed, and the similarity between the adjacent vertices of the upper and lower layers will become greater. Conversely, the smaller the number of layers is, the larger the span of the ontology concept is, and the smaller the similarity of the upper and lower concepts will become. This is what we often say about the structure distribution of the ontology graph. Again, back to look at our multi-dividing ontology algorithm in linear combination setting, it cannot reflect this structure characteristic of the ontology graph since each pair of ontology vertices comparison is from different rate of branches.

Experiment

In this section, we mainly focus on the effectiveness of the algorithm in some specific fields from experimental point of view. The ontology data we used here are all

tree-structures (or close to tree-structures) in order to fit the multi-dividing setting, and we aim to investigate the similarity-based ontology mapping between two different ontology trees in the same application field. The entire execution trees can be described as follows: first, for two university ontology graphs or two mathematical ontology graphs, domain experts determine the most similar N vertices in another ontology of each vertex (here $N = 1, 3, 5$), and they are marked as the target similarity vertex set of each vertex; then, by means of our linear combination multi-dividing ontology algorithm, we calculate the real number corresponding to each vertex, and record the most similar N vertices in corresponding ontology; for each vertex, comparing the similarity set given by the expert and obtained by the algorithm calculating, and compute the matching rate; finally, the average matching rate of the entire ontology graph is calculated in light of the matching rate of all vertices in the two ontology graphs.

Experiment on university data

University ontologies are very well-known ontologies, which often appear in some explanations about ontology introductory textbooks and examples, and the structure of two university ontologies O_1 and O_2 are depicted in Fig. 1.

In our multi-dividing linear combination setting, $k = 3$ and three branches correspond to “course”, “student” and “staff”. It is clear that $|V(G)| = 28$ where G is the union of two subgraphs and the same concept in different subgraphs can be regraded as different vertices. We take 14 vertices as ontology sample set from the whole vertex set. In order to compare it with other ontology learning algorithms, we compare the experimental data (some parts of these data have been already presented in the previous articles), and part of the results are as follows.

From the comparison of the data in Table 1, we can see that our linear combination multi-dividing ontology

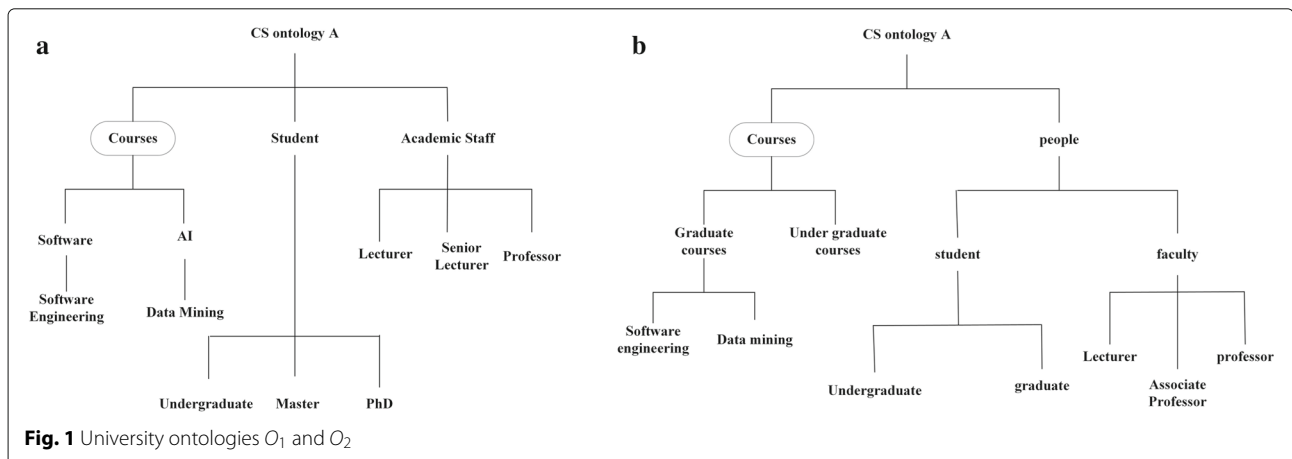


Fig. 1 University ontologies O_1 and O_2

Table 1 Comparing result on university ontology when $N = 1, 3, 5$

	P@1 data	P@3 data	P@5 data
our linear combination ontology learning algorithm	0.5357	0.6785	0.800
ontology learning algorithm in [22]	0.4643	0.5714	0.6642
ontology learning algorithm in [23]	0.5000	0.5962	0.6857
ontology learning algorithm in [36]	0.4286	0.5238	0.5929
ontology learning algorithm in [37]	0.2857	0.3929	0.5500
ontology learning algorithm in [38]	0.4643	0.5952	0.7429

learning algorithm is significantly better than the previous three algorithms for the average accuracy of university ontology.

Experiment on mathematical data

Mathematical ontologies are constructed in mathematical education and used to provide mathematical knowledge for graduate students in the field of discrete mathematics, and as the first experiment, our aim is to build a bridge between the following two mathematical ontology graphs based on similarity computing between ontology vertices. The structure of two mathematical ontologies O_3 and O_4 are depicted in Fig. 2.

Although we found that the graph structures of O_3 and O_4 are not tree, their structures are very close to the tree-shaped acyclic graph structure, and thus can be treated as a tree structure after simple processing. After analysis we take $k = 4$, and it is clear that $|V(G)| = 26$. We take half of vertices as ontology sample, i.e., $|S| = 13$. Similarly, to compare it with other ontology learning algorithms, we directly use the experimental data which were presented in the [24, 25] and [27]. Furthermore, we test the accuracy of “confidence weighted ontology algorithm” presented in

[37] and “weak function based ontology learning algorithm” manifested in [38], and compare to our ontology learning algorithm. Part of the results are as follows.

From the comparison results manifested in Table 2, we acquire that the linear combination multi-dividing ontology learning algorithm proposed in our paper has higher efficiency than the previous three algorithms for the average accuracy of mathematical ontology.

In the above two comparative experiments, we believe that the reason why the data of the ontology learning algorithm in this paper is superior to the data of other algorithms lies in that our algorithm is designed for the tree structure, while focuses and goals achieved in the engineering field of other ontology learning algorithms are differently designed. The “University” ontology is a pure tree structure, and although the “Mathematical” ontology is not strictly acyclic graph, it can also be processed and divided according to the tree structure. In contrast to other ontology learning algorithms compared in experiments, some of them are not designed for tree structures, and some of them use different angles to design algorithms. For instances, (1) although the confidence weighted ontology algorithm in [37] is also designed under a multi-dividing framework, its purpose is to save space complexity, and its core algorithm is a buffer update strategy, not an iteration of ontology functions; (2) disequilibrium ontology learning in [27] is also presented in multi-dividing ontology learning setting, while it focuses on the balance between the data rather than the structure of the ontology graph. In general, the efficiency of the algorithm in this paper reflects its advantages over tree-structured ontology graphs.

Conclusion

As a powerful auxiliary tool, the ontology has penetrated into various research fields such as chemistry, genetics, and pharmacy, providing technical support to scientists

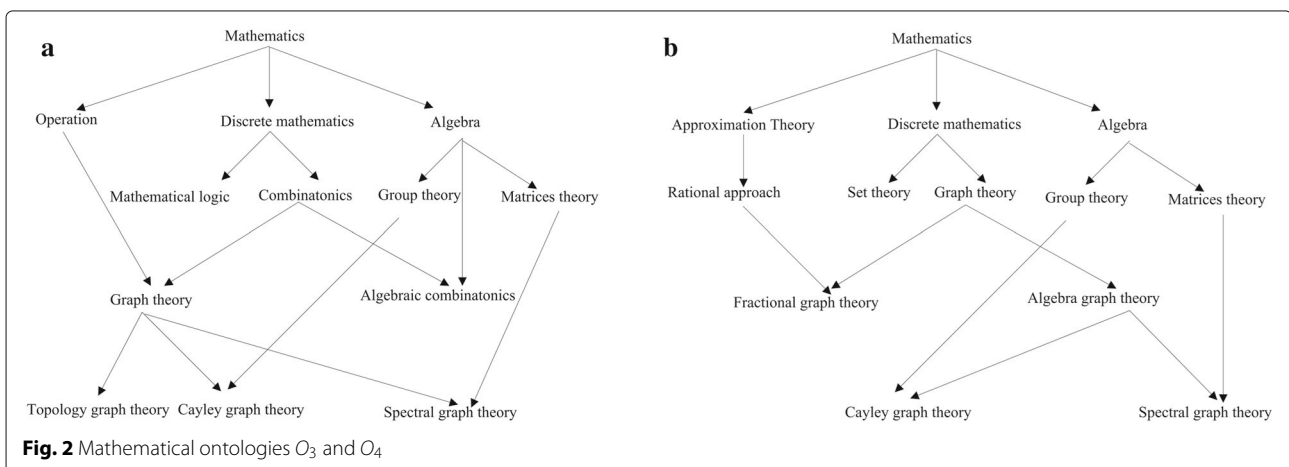


Fig. 2 Mathematical ontologies O_3 and O_4

Table 2 Comparing result on mathematical ontology when $N = 1, 3, 5$

	P@1 data	P@3 data	P@5 data
our linear combination ontology learning algorithm	0.4615	0.6538	0.8077
ontology learning algorithm in [24]	0.3846	0.5128	0.6923
PMDO Algorithm in [25]	0.4231	0.5128	0.7154
disequilibrium ontology learning in [27]	0.3846	0.5000	0.6769
ontology learning algorithm in [37]	0.2692	0.3717	0.4923
ontology learning algorithm in [38]	0.4231	0.5513	0.6846

from all walks of life. In the process of ontology construction, scholars found that most ontology uses tree structure to represent the hierarchical relationship and derivative relationship of concepts. It can be said that tree structure is the most suitable structural representation of ontology concepts. Based on this fact, researchers proposed several multi-dividing ontology learning algorithms, which specifically divide the categories of vertices for multiple branches of the ontology tree structure. The existing experimental data can fully explain that the multi-dividing ontology algorithms have higher efficiency for some well-known application ontologies (such as “GO”, “PO”, etc).

In this article, we only focus on the theoretical analysis of ontology learning algorithm. The approximation property of the multi-dividing ontology learning algorithm is analyzed from the perspective of statistical learning theory, and the result shows that the algorithm has very good approximation properties in the linear combination setting.

We list some open problems as the end of this paper:

- How to use the covering number to characterize the properties of the hypothesis space, and then obtain the theoretical boundary of the covering number approximation in multi-dividing ontology learning setting.
- What will happen if we assume the ontology tree is not balanced (n_i are not increasing at the same rate, where $i \in \{1, \dots, k\}$)?
- Find a suitable assumption to ensure the ontology function satisfy the “Uniform Glivenko-Cantelli” characteristic in multi-dividing ontology learning setting.

Acknowledgements

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

Authors' contributions

The authors have worked equally when writing this paper. All authors read and approved the final manuscript.

Funding

This work was supported in part by National Natural Science Foundation of China (11761083).

Availability of data and materials

The datasets generated during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare no conflict of interest.

Author details

¹School of Information Science and Technology, Yunnan Normal University, 650500 Kunming, China. ²Department of Mathematics, Nanjing University, 210093 Nanjing, China.

Received: 6 November 2019 Accepted: 29 April 2020

Published online: 04 June 2020

References

1. Subramaniaswamy V, Manogaran G, Logesh R, et al. (2019) An ontology-driven personalized food recommendation in IoT-based healthcare system. *J Supercomput* 75:3184–3216
2. Mohammadi M, Hofman W, Tan YH (2019) A comparative study of ontology matching systems via inferential statistics. *IEEE T Knowl Data En* 31:615–628
3. Morente-Molinera JA, Kou G, Pang C, et al. (2019) An automatic procedure to create fuzzy ontologies from users' opinions using sentiment analysis procedures and multi-granular fuzzy linguistic modelling methods. *Inform Sci* 476:222–238
4. Sacha D, Kraus M, Keim DA, et al. (2019) VIS4ML: an ontology for visual analytics assisted machine learning. *IEEE Tran Vis Comput Gr* 25:385–395
5. Kamsu-Foguem B, Abanda FH, Doumbouya MB, et al. (2019) Graph-based ontology reasoning for formal verification of BREEAM rules. *Cogn Syst Res* 55:14–33
6. Janowicz K, Haller A, Cox SJD, et al. (2019) SOSA: A lightweight ontology for sensors, observations, samples, and actuators. *J Web Semant* 56:1–10
7. Sulthana AR, Ramasamy S (2019) Ontology and context based recommendation system using Neuro-Fuzzy Classification. *Comput Electr Eng* 74:498–510
8. Nadal S, Romero O, Abello A, et al. (2019) An integration-oriented ontology to govern evolution in Big Data ecosystems. *Inform Syst* 79:3–19
9. Scarpato N, Cilia ND, Romano M (2019) Reachability matrix ontology: a cybersecurity ontology. *Appl Artif Intell* 33:643–655
10. Karimi H, Kamandi A (2019) A learning-based ontology alignment approach using inductive logic programming. *Expert Syst Appl* 125:412–424
11. Koehler S, Carmody L, Vasilevsky N, et al. (2019) Expansion of the Human Phenotype Ontology (HPO) knowledge base and resources. *Nucleic Acids Res* 47:D1018–D1027
12. Chhim P, Chinnam RB, Sadawi N (2019) Product design and manufacturing process based ontology for manufacturing knowledge reuse. *J Intell Manuf* 30:905–916
13. Ali MM, Rai R, Otte JN, et al. (2019) A product life cycle ontology for additive manufacturing. *Comput Ind* 105:191–203
14. Neveu P, Tireau A, Hilgert N, et al. (2019) Dealing with multi-source and multi-scale information in plant phenomics: the ontology-driven Phenotyping Hybrid Information System. *New Phytol* 221:588–601
15. Kiefer J, Hall JG (2019) Gene ontology analysis of arthrogryposis (multiple congenital contractures). *Am J Med Genet C* 181:310–326
16. Jaervenpaeae E, Siltala N, Hylli O, et al. (2019) The development of an ontology for describing the capabilities of manufacturing resources. *J Intell Manuf* 30:959–978
17. Serra LM, Duncan WD, Diehl AD (2019) An ontology for representing hematologic malignancies: the cancer cell ontology. *BMC Bioinformatics* 20:231–236
18. Noia TD, Mongiello M, Nocera F, et al. (2019) A fuzzy ontology-based approach for tool-supported decision making in architectural design. *Knowl Inf Syst* 58:83–112

19. Ledvinka M, Lalis A, Kremen P (2019) Toward data-driven safety: an ontology-based information system. *J Aerospace Inform Syst* 16:22–36
20. Aameri B, Cheong H, Beck JC (2019) Towards an ontology for generative design of mechanical assemblies. *Appl Ontology* 14:127–153
21. Gao W, Chen YJ, Baig AQ, et al. (2018) Ontology geometry distance computation using deep learning technology. *J Intell Fuzzy Syst* 35:4517–4524
22. Gao W, Farahani MR, Aslam A, et al. (2017) Distance learning techniques for ontology similarity measuring and ontology mapping. *Cluster Comput* 20:959–968
23. Gao W, Zhu LL, Wang KY (2015) Ontology sparse vector learning algorithm for ontology similarity measuring and ontology mapping via ADAL technology. *Int J Bifurcat Chaos* 25. <https://doi.org/10.1142/S0218127415400349>
24. Gao W, Zhang YQ, Guirao JLG, et al. (2019) A discrete dynamics approach to sparse calculation and applied in ontology science. *J Differ Equ Appl* 25(9–10):1239–1254
25. Gao W, Guirao JLG, Basavanagoud B, et al. (2018) Partial multi-dividing ontology learning algorithm. *Inform Sci* 467:35–58
26. Gao W, Farahani MR (2017) Generalization bounds and uniform bounds for multi-dividing ontology algorithms with convex ontology loss function. *Comput J* 60:1289–1299
27. Wu JZ, Yu X, Gao W (2017) Disequilibrium multi dividing ontology learning algorithm. *Comm Stat Theory Methods* 46:8925–8942
28. Sangaiah AK, Medhane DV, Han T, et al. (2019) Enforcing position-based confidentiality with machine learning paradigm through mobile edge computing in real-time industrial informatics. *IEEE Tran Ind Inform* 15(7):4189–4196
29. Gao W, Xu TW (2013) Stability analysis of learning algorithms for ontology similarity computation. *Abstr Appl Anal*. <http://dx.doi.org/10.1155/2013/174802>
30. Bryce C (2019) Security governance as a service on the cloud. *J Cloud Comput* 8:23. <https://doi.org/10.1186/s13677-019-0148-5>
31. Song R, Xiao Z, Lin J, et al. (2020) CIES: Cloud-based Intelligent Evaluation Service for video homework using CNNLSTM network. *J Cloud Comput* 9:7. <https://doi.org/10.1186/s13677-020-0156-5>
32. Dimitri N (2020) Pricing cloud IaaS computing services. *J Cloud Comput* 9:14. <https://doi.org/10.1186/s13677-020-00161-2>
33. Sangaiah AK, Medhane DV, Bian GB, et al. (2019) Energy-aware green adversary model for cyber physical security in industrial system. *IEEE Tran Ind Inform*. <https://doi.org/10.1109/TII.2019.2953289>
34. Sangaiah AK, Hosseinabadi AAR, Sadeghilalimi M, et al. (2019) Energy consumption in point-coverage wireless sensor networks via bat algorithm. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2019.2952644>
35. Gao W, Wang WF (2018) Analysis of k -partite ranking algorithm in area under the receiver operating characteristic curve criterion. *Int J Comput Math* 95:1527–1547
36. Gao W, Baig A. Q, Ali H, et al. (2017) Margin based ontology sparse vector learning algorithm and applied in biology science. *Saudi J Biol Sci* 24:132–138
37. Zhu LL, Hua G, Zafar S, et al. (2018) Fundamental ideas and mathematical basis of ontology learning algorithm. *J Intell Fuzzy Syst* 35:4503–4516
38. Zhu LL, Hua G, Aslam A (2018) Ontology learning algorithm using weak functions. *Open Phys* 16:910–916

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
