Thai Journal of Mathematics
Vol. 18, No. 1 (2020),
Pages 394 - 410

ISSN 1686-0209

FOUR-LAYER DISTANCE METRIC AND DISTANCE-BASED KERNEL FUNCTIONS FOR INDUCTIVE LOGIC PROGRAMMING

Nirattaya Khamsemanan[1], **Cholwich Nattee**[1,*], **Masayuki Numao**[2]
[1] *Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani, Thailand*
E-mail: nirattaya@siit.tu.ac.th, cholwich@siit.tu.ac.th
[2] *The Institute of Scientific and Industrial Research, Osaka University, Osaka, Japan*
E-mail: numao@sanken.osaka-u.ac.jp

Abstract Inductive Logic Programming (ILP) is a field of study focusing developing machine learning algorithms using logic programming to describe examples and hypotheses. This makes ILP techniques capable to deal with relational data, i.e. non-vector data. To learn from ILP data, an algorithm must be able to handle non-linear data. Hypotheses generated from ILP techniques are in form of Horn clauses, which can be interpreted by human. This is a benefit over conventional learning algorithms that generate black-box hypotheses or classification models. Nevertheless, learning algorithms used by ILP techniques are based on covering algorithms. It requires high computational power to generate appropriate hypotheses from a set of examples. We propose a distance metric for ILP datasets. Incorporating distances between examples in the hypothesis generation helps improve the performance of an ILP system. We also propose distance-based kernel functions for ILP datasets based on the distance metric. The kernel functions allow us to improve a hypothesis construction algorithm for ILP systems. To evaluate our proposed technique, we conduct experiments on real-world ILP datasets. The results show that the proposed technique outperforms the existing techniques.

MSC: 68T10; 68T27; 68T30; 54E35

Keywords: distance function; metric; first-order logic; multi-relational data mining; instance-based learning

Submission date: 01.12.2019 / Acceptance date: 27.01.2020

1. Introduction

Logic programming provides means for representing knowledge and multi-relational data by using First-Order Logic (FOL). It has advantages over propositional data representation, i.e. data in the vector form, in term of information expressiveness. It allows us to represent datasets composing of multiple data types as well as relationships among them. For example, structures of chemical compounds cannot be represented in form

*Corresponding author.

Published by The Mathematical Association of Thailand.

of vectors without losing information since relationships among atoms that forms each compound cannot be effectively represented using a vector. But, they can be represented by logic programming using two predicate symbols, i.e. `atom` to denote an atom in a compound with its properties and `bond` to denote a bond connecting two atoms including its properties. Other predicates can be introduced to denote substructures of a compound such as an aromatic ring.

Inductive Logic Programming (ILP) [11] is a subfield of machine learning focusing on developing learning techniques using logic programming as the data representation and generating classification models in form of Horn clauses. Therefore, the constructed hypotheses can be comprehended by human. This is a benefit over existing conventional learning techniques that generate typically black-box classification models. Until now, a number of ILP systems have been developed and applied to various application domains. However, the algorithms used in most ILP systems are based on searching and covering algorithms. Advanced optimization techniques developed for the propositional learning cannot applied due to the difference in data representation. This therefore limits the predictive performance of the hypotheses constructed by ILP systems.

In this paper, we propose a novel distance function for ILP datasets called the 'four-layer (4L) distance'. We prove that the proposed 4L distance is a metric. We then propose distance-based kernel functions for ILP datasets from the 4L distance. We can then apply learning techniques developed for the propositional learning, i.e. k-Nearest Neighbors (k-NN) and Support Vector Machines (SVM) on ILP datasets. This yields highly accurate classification models that outperforms models constructed by existing ILP systems.

Definition 1.1. A positive real-valued function $d : X \times X \to [0, \infty)$ is a metric if it satisfies the following properties:

(1) $d(x, y) = 0$ if and only if $x = y$ (Coincidence axiom)
(2) $d(x, y) = d(y, x)$ (Symmetry)
(3) $d(x, z) + d(z, y) \geq d(x, y), \forall z \in X$ (Triangular Inequality)

2. Related Works

Bisson [2] proposes a similarity function for FOL objects that is a weighted sum of similarities between predicates and their arguments. This function is therefore defined recursively since an argument of a predicate can refer to an FOL object. Bisson has shown that calculating a similarity based on the function is equivalent to solving a system of linear equations. However, the similarity is defined as a product of value-based similarities. This may cause a similarity to be 0 when one of similarities is 0. It therefore causes a loss of information.

RIBL [4] is a variant of the non-metric similarity function proposed by Bisson. Since RIBL is not a recursive function, a similarity can be computed without iterations. Thus, it improves computation time.

Ramon and Bruynooghe [14] propose a distance function for FOL objects called 'RB distance'. They also shows that the RB distance function satisfies the metric properties. The RB distance considers each FOL object as a set of predicates. However, it does not take into account the multi-level structure of the FOL objects.

Tobudic and Widmer [16] propose a distance function called 'DISTALL'. It extends RIBL to support multi-level structure in FOL objects. The distance is computed as the solution of the maximum flow, minimum weight problems on sets of predicates where a weight is set to be a Manhattan distance between two predicates. This concept allows DISTALL to capture structural and semantic information of FOL objects. However, DISTALL does not satisfy the metric properties.

Gärtner et al. [7] propose a kernel function for structured data which includes FOL objects. It can be used to compute a similarity between two FOL objects. However, the function does not satisfy the coincidence axiom.

De Raedt and Ramon [13] propose a distance metric based on the generality ordering between objects. Therefore, a generality ordering need to be defined for FOL objects before we can apply this distance function. However, it is not trivial to define an ordering based on the semantics of the objects..

A number of functions have been proposed to measure a distance between two predicates [5, 12, 15] based on vector-based distance functions, e.g. Manhattan and Euclidean distances. Since an FOL object is a set of predicates, these functions cannot be directly applied to measure a distance between two objects.

A distance between two FOL objects can be computed by first transforming the two FOL objects into their equavalent vectors and applying existing distance functions for the distance computation. Propositionalization is a technique to transform an FOL object into a vector. A number of propositionalization techniques have been proposed, e.g. Linus [10] and RRC [1]. However, the obtained vectors cannot represent all information provided by FOL objects. It is therefore not possible to perform a propositionalization process without information loss.

3. Main Results

The proposed 4L distance function is inspired by the Euclidean distance. Each predicate symbol is regarded as a dimension. The 4L distance of two objects is computed by combining distances from 4 layers, i.e. (1) distance between arguments in the same rank of two predicates, (2) distance between two predicates, (3) distance between two predicate symbols, and (4) distance between two FOL objects. To avoid a loop in distance calculation, the structure of FOL objects must be directed acyclic.

Definition 3.1. Suppose X and Y are two FOL objects whose properties are represented in a multi-level structure database $\mathcal{C}$. The objects X and Y are sets of FOL predicates. The 4L distance is defined as follows:

Layer 1:: The Four-layer Distance between two FOL objects: The distance between X and Y is defined as

$$D(X,Y) = \sqrt{\frac{\sum_{r\in\Omega} (D_r(X,Y))^2}{|\Omega|}},$$

where Ω is the set of predicate symbols of predicates in $\mathcal{C}$, and $D_r(\cdot,\cdot)$ is the distance between two FOL objects with respect to a predicate r, is defined below.

Layer 2:: Distance between two FOL objects with respect to a predicate symbol r: Suppose there are p predicates in X with the predicate symbol r, and q predicates in Y with the predicate symbol r, then the r-distance between a set X and a set Y is

$$D_r(X,Y) = \begin{cases} \max\{ & \\ \quad \max_{k=1}^{p} \min_{j=1}^{q} d_r(X^{r_k}, Y^{r_j}), & \\ \quad \max_{j=1}^{q} \min_{k=1}^{p} d_r(X^{r_k}, Y^{r_j}) & \\ \}, & \text{if } p,q \neq 0 \\ 1, & \text{if } p \neq 0, q = 0, \\ & \quad \text{or } p = 0, q \neq 0 \\ 0, & \text{if } p = q = 0 \end{cases}$$

where $d_r(\cdot,\cdot)$, the distance of predicates with respect to a predicate symbol r, is defined below.This distance function preserves the symmetry and triangle inequality properties. This layer of distance calculation is a modified Hausdorff distance.

Layer 3:: Distance of predicates with respect to a predicate symbol r**:** Suppose $X^r = r(x_1, x_2, \cdots, x_n) \in X$ and $Y^r = r(y_1, y_2, \cdots, y_n) \in Y$ are two predicates with the same predicate symbol r. The distance function of two predicates with the same predicate symbol r is defined as

$$d_r(X^r, Y^r) = \sqrt{\frac{\sum_{i=1}^{n} (\delta_{r,i}(x_i, y_i))^2}{n}},$$

where $\delta_{r,i}(\cdot,\cdot)$, the distance of arguments with respect to a predicate symbol r and rank i, is defined below.

Layer 4:: Distance of arguments with respect to a predicate symbol r **and rank** i**:** Suppose $X^r = r(x_1, x_2, \cdots, x_n) \in X$ and $Y^r = r(y_1, y_2, \cdots, y_n) \in Y$ are two predicates with the same predicate symbol r. The distance between x_i and y_i is defined as

$$\delta_{r,i}(x_i, y_i) = \begin{cases} 0 & \text{if } x_i = y_i, \\ 1 & \text{if } x_i \neq y_i, \text{ and } x_i \notin \mathbb{R} \text{ or } y_i \notin \mathbb{R}, \\ \dfrac{|x_i - y_i|}{\max(r,i)} & \text{if } x_i \neq y_i \text{ and } x_i, y_i \in \mathbb{R}. \\ D(x_i, y_i) & \text{if } x_i \neq y_i \text{ and } x_i, y_i \text{ are FOL sets.} \end{cases}$$

where $\max(r,i)$ is the maximum difference of all pairs of arguments in the rank i of predicates with the predicate symbol r, ranging over $\mathcal{C}$, and $D(x_i, y_i)$ is the four-layer distance between sets x_i and y_i as defined in Layer 1 with $\Omega_{(r,i)}$, the set of all predicate symbols that contains in set arguments in rank i of predicate symbol r.

We first prove that the 4L distance function, defined in 3.1, is a metric in a single-level structure where all arguments are treated as strings or numbers. Let

$$\Delta_{r,i}(x_i, y_i) = \begin{cases} 0 & \text{if } x_i = y_i, \\ 1 & \text{if } x_i \neq y_i, \text{ and } x_i \notin \mathbb{R} \text{ or } y_i \notin \mathbb{R}, \\ \dfrac{|x_i - y_i|}{\max(r,i)} & \text{if } x_i \neq y_i \text{ and } x_i, y_i \in \mathbb{R}. \end{cases}$$

This means that $\Delta_{r,i}(x_i, y_i) = \delta_{r,i}(x_i, y_i)$ in the setting where arguments are not sets.

Proposition 3.2. *The function $\Delta_{r,i}$ is a metric.*

Proof. Notice that $\max(r,i) \geq |x_i - y_i|, \forall x_i, y_i$, by its definition. Thus it can be seen that $0 \leq \Delta_{r,i}(x_i, y_i) \leq 1$.

Directly from its definition, $\Delta_{r,i}$ satisfies property 1 and 2 of being a metric. Remaining to prove is the triangular inequality property. Let z_i be an argument in rank i of an predicate with the predicate symbol r.

Case 1 $x_i, y_i \notin \mathbb{R}$:: ,

If: $z_i \in \mathbb{R}$, then $\Delta_{r,i}(x_i, z_i) + \Delta_{r,i}(z_i, y_i) = 2 > 1 \geq \Delta_{r,i}(x_i, y_i)$

If: $z_i \notin \mathbb{R}$, then $\Delta_{r,i}$ is the discrete metric which satisfies the triangular inequality.

Case 2 $x_i, y_i \in \mathbb{R}$:: ,

If: $z_i \in \mathbb{R}$, then $\Delta_{r,i}(x_i, z_i) = \dfrac{|x_i - y_i|}{\max(r,i)}$ is a metric in $\mathbb{R}$ because $\max(r,i)$ is a fixed constant. Hence the triangular inequality is satisfied.

If: $z_i \notin \mathbb{R}$, $\Delta_{r,i}(x_i, z_i) + \Delta_{r,i}(z_i, y_i) = 2 > 1 \geq \Delta_{r,i}(x_i, y_i)$

Case 3 $x_i \notin \mathbb{R}, y_i \in \mathbb{R}$:: ,

If: $z_i \in \mathbb{R}$, $\Delta_{r,i}(x_i, z_i) + \Delta_{r,i}(z_i, y_i) = 1 + \Delta_{r,i}(z_i, y_i) \geq 1 = \Delta_{r,i}(x_i, y_i)$

If: $z_i \notin \mathbb{R}$, $\Delta_{r,i}(x_i, z_i) + \Delta_{r,i}(z_i, y_i) = \Delta_{r,i}(x_i, z_i) + 1 \geq 1 = \Delta_{r,i}(x_i, y_i)$

■

Proposition 3.3. *The function d'_r, defined below, is a metric.*

$$d'_r(X^r, Y^r) = \sqrt{\frac{\sum_{i=1}^{n} \left(\Delta_{r,i}(x_i, y_i)\right)^2}{n}},$$

Proof. : We have,

$$\begin{aligned} 0 \leq \Delta_{r,i}(x_i, y_i) \leq 1 &\Rightarrow \leq \left(\Delta_{r,i}(x_i, y_i)\right)^2 \leq 1 \\ &\Rightarrow \sum_{i=1}^{n} \left(\Delta_{r,i}(x_i, y_i)\right)^2 \leq n \\ &\Rightarrow 0 \leq d_r(X^r, Y) \leq 1. \end{aligned}$$

From Proposition 3.2, the distance $\Delta_{r,i}$ is a metric. It can be seen that d_r satisfies the symmetry property. Yet again, the triangular inequality property must be proven in this case. Notice that:

$$\left(\sqrt{\sum_{i=1}^{n}(\Delta_{r,i}(x_i,z_i))^2}+\sqrt{\sum_{i=1}^{n}(\Delta_{r,i}(z_i,y_i))^2}\right)^2$$
$$=\sum_{i=1}^{n}(\Delta_{r,i}(x_i,z_i))^2+\sum_{i=1}^{n}(\Delta_{r,i}(z_i,y_i))^2+$$
$$2\sqrt{\left(\sum_{i=1}^{n}(\Delta_{r,i}(x_i,z_i))^2\right)\left(\sum_{i=1}^{n}(\Delta_{r,i}(z_i,y_i))^2\right)}$$
$$\geq\sum_{i=1}^{n}(\Delta_{r,i}(x_i,z_i))^2+\sum_{i=1}^{n}(\Delta_{r,i}(z_i,y_i))^2+$$
$$2\sqrt{\left(\sum_{i=1}^{n}\Delta_{r,i}(x_i,z_i)\Delta_{r,i}(z_i,y_i)\right)^2} \tag{3.1}$$
$$=\sum_{i=1}^{n}(\Delta_{r,i}(x_i,z_i))^2+\sum_{i=1}^{n}(\Delta_{r,i}(z_i,y_i))^2+$$
$$2\left(\sum_{i=1}^{n}\Delta_{r,i}(x_i,z_i)\Delta_{r,i}(z_i,y_i)\right)$$
$$=\sum_{i=1}^{n}\Big((\Delta_{r,i}(x_i,z_i))^2+(\Delta_{r,i}(z_i,y_i))^2+2\Delta_{r,i}(x_i,z_i)\Delta_{r,i}(z_i,y_i)\Big)$$
$$=\sum_{i=1}^{n}(\Delta_{r,i}(x_i,z_i)+\Delta_{r,i}(z_i,y_i))^2$$
$$\geq\sum_{i=1}^{n}(\Delta_{r,i}(x_i,y_i))^2 \tag{3.2}$$

Note that (3.1) comes from the Cauchy-Schwarz inequality, and (3.2) comes from Proposition 3.2. Hence, $d'_r(X^r,Z^r)+d'_r(Z^r,Y^r)\geq d'_r(X^r,Y^r)$ as desired.

■

Proposition 3.4. *The function D'_r, defined below, is a metric.*

$$D'_r(X,Y)=\begin{cases}\max\{ & \\ \quad\max_{k=1}^{p}\min_{j=1}^{q}d'_r(X^{r_k},Y^{r_j}), & \\ \quad\max_{j=1}^{q}\min_{k=1}^{p}d'_r(X^{r_k},Y^{r_j}) & \\ \}, & \text{if } p,q\neq 0\\ 1, & \text{if } p\neq 0, q=0,\\ & \text{ or } \ p=0,q\neq 0\\ 0, & \text{if } p=q=0\end{cases}$$

Proof. From Proposition 3.3, it is easy to see that $0\leq D'_r(X,Y)\leq 1$.

It suffices to only show that $D'_r(X,Y)$ satisfies the triangular inequality since the symmetry property is easily derived from its definition.

Let Z be an object. Thus Z is a set of FOL predicates. Suppose that there are m predicates in Z with predicate symbol r.

If: $p,q,m \neq 0$, then D'_r is simply the Hausdorff metric.
If: $p,q \neq 0, m = 0$, then $D'_r(X,Z) + D'_r(Z,Y) = 2 \geq D'_r(X,Y)$.
If: $p \neq 0, q = m = 0$, then $D'_r(X,Z) + D'_r(Z,Y) = 1 + 0 = 1 = D'_r(X,Y)$.
If: $p,q,m = 0$, then $D'_r(X,Z) + D'_r(Z,Y) = 0 = D'_r(X,Y)$.

■

Theorem 3.5. *The function D', defined below, is a metric.*

$$D'(X,Y) = \sqrt{\frac{\sum\limits_{r\in\Omega} \left(D'_r(X,Y)\right)^2}{|\Omega|}},$$

where Ω is the set of predicate symbols of predicates

Proof. Because of its definition, along with Proposition 3.4, one can see that $0 \leq D'_r(X,Y) \leq 1$.

We will now prove all three properties of a metric for $D'(X,Y)$

(1) Coincidence axiom:
($\Leftarrow$) Suppose that $X = Y$. This means that X and Y are the same set. Therefore all predicates in both sets are the same which means that $D'_r(X,Y) = 0$ for all $r \in \Omega$. Hence $D'(X,Y) = 0$
($\Rightarrow$) Suppose that $D'(X,Y) = 0$. Let $X^r = r(x_1,\cdots,x_n) \in X$ and $Y^r = r(y_1,\cdots,y_n) \in Y$. Proof by contradiction, suppose also that $X \neq Y$

$$\begin{aligned}
X \neq Y \quad &\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : X^r \neq Y^r, \text{ for some } r \in \Omega \\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : x_i \neq y_i, \text{ for some rank } i \\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : \Delta(x_i,y_i) > 0, \text{ for some rank } i \\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : d'_r(X^r,Y^r) > 0, \text{ for some } r \in \Omega \\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : D'_r(X^r,Y^r) > 0, \text{ for some } r \in \Omega \\
&\Rightarrow \quad D'(X,Y) > 0,
\end{aligned}$$

which leads to a contradiction.

(2) Symmetry:

$$D'(X,Y) = \sqrt{\frac{\sum\limits_{r\in\Omega} \left(D'_r(X,Y)\right)^2}{|\Omega|}} = \sqrt{\frac{\sum\limits_{r\in\Omega} \left(D'_r(Y,X)\right)^2}{|\Omega|}} = D'(Y,X).$$

(3) Triangular Inequality: Suppose that Z is another set. The following proof is similar to that of Proposition 3.3. First notice that

$$\begin{aligned}
&\left(\sqrt{\sum_{r\in\Omega}\left(D'_r(X,Z)\right)^2}+\sqrt{\sum_{r\in\Omega}\left(D'_r(Y,Z)\right)^2}\right)^2 \\
&=\sum_{r\in\Omega}\left(D'_r(X,Z)\right)^2+\sum_{r\in\Omega}\left(D'_r(Y,Z)\right)^2+ \\
&\quad 2\sqrt{\left(\sum_{r\in\Omega}\left(D'_r(X,Z)\right)^2\right)\left(\sum_{r\in\Omega}\left(D'_r(Y,Z)\right)^2\right)} \\
&\geq\sum_{r\in\Omega}\left(D'_r(X,Z)\right)^2+\sum_{r\in\Omega}\left(D'_r(Y,Z)\right)^2+ \\
&\quad 2\sqrt{\left(\sum_{r\in\Omega}D'_r(X,Z)D'_r(Y,Z)\right)^2} \qquad (3.3)\\
&=\sum_{r\in\Omega}\left[\left(D'_r(X,Z)\right)^2+\left(D'_r(Y,Z)\right)^2+2D'_r(X,Z)D'_r(Y,Z)\right] \\
&=\sum_{r\in\Omega}\left[D'_r(X,Z)+D'_r(Y,Z)\right]^2 \\
&\geq\sum_{r\in\Omega}\left[D'_r(X,Y)\right]^2 \qquad (3.4)
\end{aligned}$$

Note that (3.3) comes from the Cauchy-Schwarz inequality and (3.4) comes from Proposition 3.4. Hence, $D'(X,Z)+D'(Z,Y)\geq D'(X,Y)$ as desired.

■

Theorem 3.6. *The four-layer distance function is a metric on a directed acyclic FOL dataset.*

Proof. Proof by induction: We prove that the four-layer distance function is a metric for a multi-level structure.
Base Case: The base case is a single-level structure, which is proved in Theorem 3.5.
Inductive Step: Induction Hypothesis: Assume that the 4L distance function is a metric for up to an n-level structure.

It remains to prove that D is a metric for an $(n+1)$-level structure which means that we have to prove that D is also a metric when one more level is added to the calculation.

Notice that

$$\delta_{r,i}(x_i,y_i)=\begin{cases}\Delta_{r,i}(x_i,y_i) & \text{if } x_i,y_i \text{ are not both sets}\\ D(x_i,y_i) & \text{if } x_i,y_i \text{ are both sets.}\end{cases}$$

where $\Delta_{r,i}$ is the same function as in Proposition 3.2.

(1) Coincidence axiom:
($\Leftarrow$) Suppose that $X=Y$. This means that X and Y are the same set. Therefore all predicates in both sets are the same which means that $D_r(X,Y)=0$ for all

$r \in \Omega$. Hence, $D(X,Y)=0$
($\Rightarrow$) Suppose that $D(X,Y)=0$. Let $X^r = r(x_1,\cdots,x_n) \in X$ and $Y^r = r(y_1,\cdots,y_n) \in Y$. Proof by contradiction: suppose also that $X \neq Y$

$$\begin{aligned}
X \neq Y \quad &\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : X^r \neq Y^r, \text{ for some } r \in \Omega\\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : x_i \neq y_i, \text{ for some rank } i\\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : D(x_i,y_i) > 0, \text{ if } x_i, y_i \text{ are both sets, or,} \qquad (3.5)\\
& \quad \exists X^r \in X, \forall Y^r \in Y : \Delta_{r,i}(x_i,y_i) > 0, \text{ if } x_i, y_i \text{ are not both sets.}\\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : \delta(x_i,y_i) > 0, \text{ for some rank } i\\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : d_r(X^r,Y^r) > 0, \text{ for some } r \in \Omega\\
&\Rightarrow \quad \exists X^r \in X, \forall Y^r \in Y : D_r(X^r,Y^r) > 0, \text{ for some } r \in \Omega\\
&\Rightarrow \quad D(X,Y) > 0,
\end{aligned}$$

which leads to a contradiction. Note that the line (3.5) is by the Induction Hypothesis.

(2) Symmetry: First, we prove the symmetry property of $d_r(X^r,Y^r)$:
Without loss of generality, suppose that x_i and y_i are sets. For all $1 \leq i \leq k \leq n$, at most one of x_j, y_j is a set for $k+1 \leq j \leq n$. This means that

$$\begin{aligned}
d_r(X^r,Y^r) &= \sqrt{\frac{\sum_{i=1}^{n} (\delta_{r,i}(x_i,y_i))^2}{n}}\\
&= \sqrt{\frac{\sum_{i=1}^{k} (D(x_i,y_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(x_j,y_j))^2}{n}},\\
&= \sqrt{\frac{\sum_{i=1}^{k} (D(y_i,x_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(y_j,x_j))^2}{n}}\\
&= \sqrt{\frac{\sum_{i=1}^{n} (\delta_{r,i}(y_i,x_i))^2}{n}}\\
&= d_r(Y^r,X^r). \qquad (3.6)
\end{aligned}$$

Hence, similar to the proof of Proposition 3.4 and Theorem 3.5,

$$D(X,Y) = \sqrt{\frac{\sum_{r\in\Omega} (D_r(X,Y))^2}{|\Omega|}} = \sqrt{\frac{\sum_{r\in\Omega} (D_r(Y,X))^2}{|\Omega|}} = D(Y,X).$$

(3) Triangular inequality property: First, we prove the triangular inequality property of $d_r(X^r,Y^r)$:
Without loss of generality, suppose that x_i and y_i are sets. For all $1 \leq i \leq k \leq n$,

at most one of x_j, y_j is a set for $k+1 \leq j \leq n$. This means that

$$\begin{aligned} d_r(X^r, Y^r) &= \sqrt{\frac{\sum_{i=1}^{n} (\delta_{r,i}(x_i, y_i))^2}{n}} \\ &= \sqrt{\frac{\sum_{i=1}^{k} (D(x_i, y_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(x_j, y_j))^2}{n}}, \end{aligned}$$

$$\begin{aligned} d_r(X^r, Z^r) &= \sqrt{\frac{\sum_{i=1}^{n} (\delta_{r,i}(x_i, z_i))^2}{n}} \\ &= \sqrt{\frac{\sum_{i=1}^{k} (D(x_i, z_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(x_j, z_j))^2}{n}}, \end{aligned}$$

$$\begin{aligned} d_r(Z^r, Y^r) &= \sqrt{\frac{\sum_{i=1}^{n} (\delta_{r,i}(z_i, y_i))^2}{n}} \\ &= \sqrt{\frac{\sum_{i=1}^{k} (D(z_i, y_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(z_j, y_j))^2}{n}}. \end{aligned}$$

Notice that,

$$\begin{aligned} &\left(\sqrt{\sum_{i=1}^{n} (\delta_{r,i}(x_i, z_i))^2} + \sqrt{\sum_{i=1}^{n} (\delta_{r,i}(z_i, y_i))^2}\right)^2 \\ &= \left(\sqrt{\sum_{i=1}^{k} (D(x_i, z_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(x_j, z_j))^2} + \sqrt{\sum_{i=1}^{k} (D(z_i, y_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(z_j, y_j))^2}\right)^2 \\ &= \left(\sum_{i=1}^{k} (D(x_i, z_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(x_j, z_j))^2\right) + \left(\sum_{i=1}^{k} (D(z_i, y_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(z_j, y_j))^2\right) \\ &\quad + 2\sqrt{\Big(\sum_{i=1}^{k} (D(x_i, z_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(x_j, z_j))^2\Big)\Big(\sum_{i=1}^{k} (D(z_i, y_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(z_j, y_j))^2\Big)} \\ &\geq \Big(\sum_{i=1}^{k} (D(x_i, z_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(x_j, z_j))^2\Big) + \Big(\sum_{i=1}^{k} (D(z_i, y_i))^2 + \sum_{j=k+1}^{n} (\Delta_{r,i}(z_j, y_j))^2\Big) \\ &\quad + 2\sqrt{\left(\sum_{i=1}^{k} D(x_i, z_i)D(z_i, y_i) + \sum_{j=k+1}^{n} \Delta_{r,i}(x_j, z_j)\Delta_{r,i}(z_j, y_j)\right)^2} \end{aligned} \tag{3.7}$$

$$
\begin{aligned}
&= \Big(\sum_{i=1}^{k}(D(x_i,z_i))^2 + \sum_{j=k+1}^{n}(\Delta_{r,i}(x_j,z_j))^2\Big) + \Big(\sum_{i=1}^{k}(D(z_i,y_i))^2 + \sum_{j=k+1}^{n}(\Delta_{r,i}(z_j,y_j))^2\Big) \\
&\quad + 2\left(\sum_{i=1}^{k}D(x_i,z_i)D(z_i,y_i) + \sum_{j=k+1}^{n}\Delta_{r,i}(x_j,z_j)\Delta_{r,i}(z_j,y_j)\right) \\
&= \sum_{i=1}^{k}\Big((D(x_i,z_i))^2 + (D(z_i,y_i))^2 + 2D(x_i,z_i)D(z_i,y_i)\Big) \\
&\quad + \sum_{j=k+1}^{n}\Big((\Delta_{r,i}(x_j,z_j))^2 + (\Delta_{r,i}(z_j,y_j))^2 + 2\Delta_{r,i}(x_j,z_j)\Delta_{r,i}(z_j,y_j)\Big) \\
&= \sum_{i=1}^{k}(D(x_i,z_i)+D(z_i,y_i))^2 + \sum_{j=k+1}^{n}(\Delta_{r,i}(x_j,z_j)+\Delta_{r,i}(z_j,y_j))^2 \\
&\geq \sum_{i=1}^{k}(D(x_i,y_i))^2 + \sum_{j=k+1}^{n}(\Delta_{r,i}(x_j,y_j))^2 \qquad (3.8) \\
&= \sum_{i=1}^{n}(\Delta_{r,i}(x_i,y_i))^2
\end{aligned}
$$

Note that (3.7) comes from the Cauchy-Schwarz inequality and (3.8) is valid because of the induction hypothesis and Proposition 3.2. Hence

$$d_r(X^r, Z^r) + d_r(Z^r, Y^r) \geq d_r(X^r, Y^r).$$

The rest of the proof of the triangular inequality of D is similar to the proof of D' in Theorem 3.5.

Thus, D is also a metric when one level is added to the calculation. This means that the case $n+1$ is also valid. Therefore, by induction, D is a metric for all n-level deep. ■

Algorithm 1 shows how to calculate a 4L distance between two FOL objects.

Definition 3.7. A kernel $k : \mathcal{C} \times \mathcal{C} \to \mathbb{R}$ is a real-valued function taking a cartesian product of elements in $\mathcal{C}$ and returns a real value. If k is positive definite, there exists a map ϕ that isometically embeds $\mathcal{C}$ into a Hilbert space $\mathcal{H}$ such that

$$\langle \phi(X), \phi(Y) \rangle = k(X, Y)$$

The positive definite property of a kernel is required by SVM learning algorithm in order to secure the maximal margin in the Hilbert space $\mathcal{H}$.

Definition 3.8. A distance-based kernel is a kernel created from a distance metric d, such that

$$d(X, Y) = \|\phi(X) - \phi(Y)\|_{\mathcal{H}}$$

From definition 3.8, we define a kernel function as:

$$
\begin{aligned}
k(X, Y) &= k_O(X, Y) \\
&= \frac{1}{2}\left(D(X,Y)^2 - D(X,O)^2 - D(Y,O)^2\right) \qquad (3.9)
\end{aligned}
$$

where $D(X, Y)$ is the 4L distance, and O is a fixed object in a dataset $\mathcal{C}$.

Notice that

Algorithm 1 Four-Layer Distance

```
**Input**: a set of predicates $\mathcal{C}$, and a set of FOL objects (sets) $\mathcal{I}$

$S \leftarrow$ *empty stack*
**function** $D(X, Y, r, i)$
    $\Omega \leftarrow$ PredicateSymbols($\mathcal{C}, r, i$)
    **if** $X = Y$ **then**
        **return** 0
    **else**
        Push($S, \langle X, Y\rangle$)
        $s \leftarrow \sqrt{\left(\sum_{r\in\Omega}(D(r, X, Y)^2)\right)/|\Omega|}$
        Pop($S$)
        **return** $s$
    **end if**
**end function**

**function** $Dr(r, X, Y)$
    $P \leftarrow$ FindPredicatesBySet($X, r$)
    $Q \leftarrow$ FindPredicatesBySet($Y, r$)
    **if** $|P| = |Q| = 0$ **then**
        **return** 0
    **else if** $(|P| \neq 0 \wedge |Q| = 0) \vee (|P| = 0 \wedge |Q| \neq 0)$ **then**
        **return** 1
    **else**
        **return** Hausdorff($d_r, P, Q$)
    **end if**
**end function**

**function** $d(r, (X, x_1, x_2, \ldots, x_n), (Y, y_1, y_2, \ldots, y_n))$
    **return** $\sqrt{\left(\sum_{i=1}^{n} \delta_{r,i}(x_i, y_i)^2\right)/n}$
**end function**
```

$$
\begin{aligned}
\|\phi(X) - \phi(Y)\|^2_{\mathcal{H}} &= \langle \phi(X) - \phi(Y)\rangle\langle \phi(X) - \phi(Y)\rangle \\
&= \langle \phi(X), \phi(X)\rangle - 2\langle \phi(X), \phi(Y) + \langle \phi(Y), \phi(Y)\rangle \\
&= \left(\frac{1}{2}\left(D(X,X)^2 - D(X,O)^2 - D(X,O)^2\right)\right) - \\
&\quad 2\left(\frac{1}{2}\left(D(X,Y)^2 - D(X,O)^2 - D(Y,O)^2\right)\right) + \\
&\quad \left(\frac{1}{2}\left(D(Y,Y)^2 - D(Y,O)^2 - D(Y,O)^2\right)\right) \\
&= D(X,Y)^2.
\end{aligned}
$$

```
function $\delta(r, i, x, y)$
    if $x = y$ then
        return 0
    else if either $x$ or $y$ is a Set then
        return $\Delta(r, i, x, y)$
    else if both $x$ and $y$ is a Set then
        return $D(x, y, r, i)$
    end if
end function

function $\Delta(r, i, x, y)$
    if $x = y$ then
        return 0
    else if $x \neq y \wedge (x \notin \mathbb{R} \vee y \notin \mathbb{R})$ then
        return 1
    else
        return $|x - y| / \max(r, i)$
    end if
end function
```

Haasdonk and Bahlmann [8] present a variant of kernel functions that can be constructed from a distance function. Following the presented kernel functions, we create the following kernel functions from the 4L distance:

(1) The 4L distance-based simple linear kernel:

$$k_{4L}^{lin}(X, Y) = \frac{1}{2}\left(D(X, Y)^2 - D(X, O)^2 - D(Y, O)^2\right)$$

where O is a fixed object in a dataset $\mathcal{C}$.

(2) The 4L distance-based negative-distance kernel:

$$k_{4L}^{nd}(X, Y) = -\left(D(X, Y)^2\right).$$

(3) The 4L distance based polynomial kernel:

$$k_{4L}^{pol}(X, Y) = \left(1 + \gamma\left(D(X, Y)^2 - D(X, O)^2 - D(Y, O)^2\right)\right)^p$$

where $\gamma \in \mathbb{R}^+$ and $p \in \mathbb{N}$.

(4) The 4L distance-based Gaussian kernel:

$$k_{4L}^{gs}(X, Y) = e^{-\gamma D(X, Y)^2}$$

where $\gamma \in \mathbb{R}^+$.

The 4L kernels are not necessarily positive definite functions because the 4L distance function is not conditionally positive definite on some datasets. However, a positive semidefinite kernel matrix is necessary for the SVM learning algorithm. Wu et al. [17] propose *shift spectrum transformation* that transforms a indefinite kernel matrix into a positive semidefinite one. The transformation is done by

Table 1. The datasets used in the experiments

Name	**#Examples**	**#Predicates**
Mutagenesis	188	14,147
Alz amine	686	987
Alz toxic	886	1,187
Alz acetyl	1,326	1,627
Alz memory	642	943
NCI GI50 BT549	2,778	134,578
DSSTox NCTRER	232	9,285
DSSTox CPDBAS	788	23,366

$$\widetilde{K} = U\widetilde{\Lambda}U^T = U(\Lambda + \eta I)U^T = K + \eta I$$

where K is an indefinite kernel matrix. Wu et al. also show that if η is greater than $|\lambda_N|$, where $\lambda \geq \lambda_N$ for all eigenvalues λ of K, then $\widetilde{K}$ is positive semidefinite.

4. Experiments

We conduct experiments to evaluate the proposed 4L distance and 4L kernels using real-world ILP datasets, i.e. Mutegenesis [3], Alzheimer [9], NCI GI50 BT549, EPA's DSSTox NCTRER [6] and EPA's DSSTox CPDBAS. Table 1 shows the number of FOL objects and predicates in each dataset.

We compare the performance of our proposed 4L distance function using k-NN. The performances of the 4L-kernels are evaluated by SVM. We compare the results with the existing distance and kernel functions for FOL objects, i.e. structured data kernel (SK), RB distance (RB), DISTALL (DT), and RIBL.

The experiments are conducted using the nested cross validation technique using 10-fold cross validation in both inner and outer loop. We conduct the grid search in order to find the best combination of hyperparameter values. Table 2 shows the values of the hyperparameters used in the experiments. For each dataset and a kernel function, if the Gram matrix is indefinite, we apply the shift spectrum transformation with $\eta = |\lambda_N|$ where λ_N is the minimum eigenvalue.

Table 2. Ranges of hyperparameter values using the grid search

Alg.	**Hyperparameter**	**Values**
k-NN	The number of neighbors (k)	$1 \ldots 30$
SVM	Types of kernel functions	linear, gaussian, polynomial, negative
	Penalty of errors (C)	10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3, 10^4, 10^5, 10^6
	Degree of the polynomial kernel	$2, 3, 4, 5, 6$
	Kernel coefficient (γ) for polynomial & Gaussian kernels)	10^{-3}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2, 10^3

Table 3 shows the experimental results. Our proposed 4L distance and kernels outperforms the existing techniques when used the same way. The k-NN using the 4L distance performs the best on Mutagenesis and BT549 datasets. SVM with non-linear kernels yield the highest accuracy for all Alz subdatasets, NCTRER, and CPDBAS datasets.

Table 3. Experimental results

Alg.	Dist.	Mutag	Alz	Alz	Alz	Alz toxic	BT549	NCT	CPD
			amine	choline	schopo			RER	BAS
kNN	4L	**87.34**%	92.86%	88.46%	86.92%	93.79%	**63.03**%	63.77%	63.96%
	RB	82.98%	87.19%†	83.86%†	75.08%†	89.73%†	49.43%†	43.88%†	51.66%†
	DK	66.61%†	81.21%†	83.26%†	77.41%†	88.71%†	47.63%†	59.93%	52.80%†
	DT	24.42%†	73.18%†	81.22%†	66.81%†	78.44%†	49.46%†	41.70%†	51.52%†
	RIBL	43.77%†	32.07%†	39.30%†	54.19%†	39.96%†	49.82%†	37.05%†	46.84%†
SVM	4L	85.23%	**96.80**%	**96.15**%	**92.05**%	**98.65**%	61.95%	**68.12**%	**65.86**%
	RB	86.20%	93.74%†	85.90%†	80.53%†	95.26%†	60.40%	60.81%	57.36%†
	DK	63.30%†	93.31%†	94.42%†	87.07%†	97.40%†	58.89%†	60.78%†	57.11%†
	DT	66.61%†	75.08%†	84.16%†	65.11%†	72.81%†	61.01%	59.98%	50.77%†
	RIBL	66.61%†	45.04%†	60.11%†	51.07%†	44.25%†	61.77%	50.89%†	57.36%†

Note: † indicates that the result is significantly different from the best result with $p = 0.05$.

In Alz subdatasets, 4L kernels outperform other techniques significantly since each Alz subdataset contains 21 predicate symbols. In datasets with lower numbers of predicate symbols such as BT549, or Mutagenesis, the proposed 4L techniques still outperform others, but not as significantly as in higher dimension datasets. These results show that proposed 4L kernels perform better with datasets with high numbers of predicate symbols. This validates the concept of "dimension", which the proposed 4L distance function is based on. Existing techniques are based on measuring differences between all predicates, regardless of their predicate symbols. Since 4L techniques perform significantly better in datasets with more predicate symbols than the rest, the results suggests that differences among predicates with the same predicate symbols (dimensions) reveal more distinctions between two FOL objects than measuring all predicates without considering dimensions.

5. Conclusion

We create a novel distance function to measure the difference between FOL objects, called the four-layer (4L) distance function. Each predicate symbol is viewed as a dimension of a space. The first three layers of distance calculations are performed to measure the difference of two FOL objects with respect to a predicate symbol. These distances in all dimensions (predicate symbols) are combined in the last layer to yield the distance between two objects. This also supports multi-level structure datasets. In the experiment, we employ the 4L distance with the k-NN algorithm. We also create distance-based kernel functions from the 4L distance for SVM. The proposed techniques outperform existing techniques ([7], [14],[16], [4]) in datasets Mutagenesis [3], Alzheimer [9], NCI GI50 BT549, EPA's DSSTox NCTRER dataset [6], and EPA's DSSTox CPDBAS dataset. The results show that the techniques using 4L distance and kernels perform well for both linear and non-linear datasets. Moreover, the proposed techniques operate well on datasets with higher number of predicate symbols that endorse the concept of dimensions. Since the 4L function is a metric, a directed acyclic FOL dataset with 4L metric can now be considered as a metric space.

Acknowledgements

This work was performed under the Cooperative Research Program of "Network Joint Research Center for Materials and Devices: Dynamic Alliance for Open Innovation Bridging Human, Environment and Materials."

References

[1] G. Anderson, B. Pfahringer, Clustering relational data based on randomized propositionalization. In ILP 2007, LNAI 4894 (2008) 39–48.

[2] G. Bisson, Learning in fol with a similarity measure. In AAAI-92 Proceedings (1992) 82–87.

[3] A.K. Debnath, R.L. Lopez de Compadre, G. Debnath, A.J. Shusterman, C. Hansch, Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds, correlation with molecular orbital energies and hydrophobicity, J. Med. Chem. 34(1991) 786–797.

[4] W. Emde, D. Wettschereck, Relational instance-based learning, In Proceedings of the International Conference on Machine Learning (ICML) (1996) 122–130.

[5] V. Estruch, C. Ferri, J. Hernández-Orallo, M. José Ramírez-Quintana. An integrated distance for atoms. In Functional and Logic Programming (2010) 150–164.

[6] H. Fang, W. Tong, L.M. Shi, R. Blair, R. Perkins, W. Branham, B.S. Hass, Q. Xie, S.L. Dial, C.L. Moland, D.M. Sheehan, Structure-activity relationships for a large diverse set of natural, synthetic, and environmental estrogens, Chemical Research in Toxicology, 14(3)(2001) 280–294.

[7] T. Gärtner, J.W. Lloyd, P.A. Flach, Kernels and distances for structured data, Machine Learning 57(2004) 205–232.

[8] B. Haasdonk, C. Bahlmann, Learning with distance substitution kernels, In CarlEdward Rasmussen, HeinrichH. Blthoff, Bernhard Schlkopf, and MartinA. Giese, editors, Pattern Recognition, volume 3175 of Lecture Notes in Computer Science, pages 220–227. Springer Berlin Heidelberg, 2004.

[9] R.D. King, M.J.E. Sternberg, A. Srinivasan, Relating chemical activity to structure: An examination of ilp successes, New Generation Computing, 13(3-4)(1995) 411–433.

[10] N. Lavrac, S. Dzeroski, Inductive Logic Programming: Techniques and Applications, Ellis Horwood, 1994.

[11] S. Muggleton, L. de Raedt, Inductive logic programming: Theory and methods, The Journal of Logic Programming (1994) 629–679.

[12] S.H. Nienhuys-Cheng, Distance between herbrand interpretations: A measure for approximations to a target concept, In Proceedings of the 7th International Workshop on Inductive Logic Programming, Lecture Notes in Artificial Intelligence (1997) 213–22.

[13] L. De Raedt, J. Ramon, Deriving distance metrics from generality relations, Pattern Recognition Letters 30(2009) 187–191.

[14] J. Ramon, M. Bruynooghe, A polynomial time computable metric between point sets, Acta Informatica 37(10)(2001) 765–780.

[15] J. Ramon, M. Bruynooghe, W. Van Laer, Distance measures between atoms. In CompulogNet Area Meeting on Computational Logic and Machine Learning (1998) 35–41.

[16] A. Tobudic, G. Widmer, Relational ibl in classical music, Machine Learning (2006) 64:5–24.

[17] G. Wu, E.Y. Chang, Z. Zhang, An analysis of transformation on non-positive semidefinite similarity matrix for kernel machines, In Proceedings of the 22nd International Conference on Machine Learning (2005).