Distance-based Evaluation Function for First-order Rule Construction

Nirattaya Khamsemanan[1], Cholwich Nattee[1],
and Masayuki Numao[2]

[1]Sirindhorn International Institute of Technology, Thammasat University
[2]The Institute of Scientific and Industrial Research, Osaka University

ILP 2016

First-Order Logic objects

- An object can be represented as a set of predicates.
- A predicate describes the object's properties, components, relationships among components, etc.
- Names or IDs are typically used for the identification.

```
nitro(d1,d1_19,d1_24,d1_25,d1_26).
bond(d1,d1_1,d1_2,7).
bond(d1,d1_2,d1_3,7).
bond(d1,d1_3,d1_4,7).
bond(d1,d1_4,d1_5,7).
atm(d1_1,c,22,-0.117).
atm(d1_2,c,22,-0.117).
atm(d1_3,c,22,-0.117).
atm(d1_4,c,195,-0.087).
atm(d1_5,c,195,0.013).
```

Setting for FOL Objects

In our setting, every predicate is written in form of

$$r(ID, x_1, x_2, \ldots, x_n)$$

where

- r is a predicate symbol,
- ID is an object described by the predicate,
- x_i is a property value.

Distance between FOL objects

d1 ← *distance* → e25

```
nitro(d1,d1_19,d1_24,
      d1_25,d1_26).
bond(d1,d1_1,d1_2,7).
bond(d1,d1_2,d1_3,7).
bond(d1,d1_3,d1_4,7).
bond(d1,d1_4,d1_5,7).
atm(d1_1,c,22,-0.117).
atm(d1_2,c,22,-0.117).
atm(d1_3,c,22,-0.117).
atm(d1_4,c,195,-0.087).
atm(d1_5,c,195,0.013).
```

```
nitro(e25,e25_7,e25_21,
      e25_22,e25_23).
bond(e25,e25_1,e25_2,1).
bond(e25,e25_2,e25_3,1).
bond(e25,e25_3,e25_4,7).
bond(e25,e25_4,e25_5,1).
atm(e25_1,c,10,-0.083).
atm(e25_2,c,10,0.017).
atm(e25_3,c,22,-0.113).
atm(e25_4,c,22,0.017).
atm(e25_5,c,10,-0.083).
```

Distance Functions for FOL objects

A number of distance functions have been proposed to measure dissimilarity between two FOL objects:

- FOL Similarity (Bisson, 1992)
- RIBL (Emde and Wettschereck, 1996)
- RB distance (Ramon and Bruynooghe, 2001)
- Kernels and distances for structured data (Gärtner et al., 2004)
- DISTALL (Tobudic and Widmer, 2006)

Four-layer Distance Function

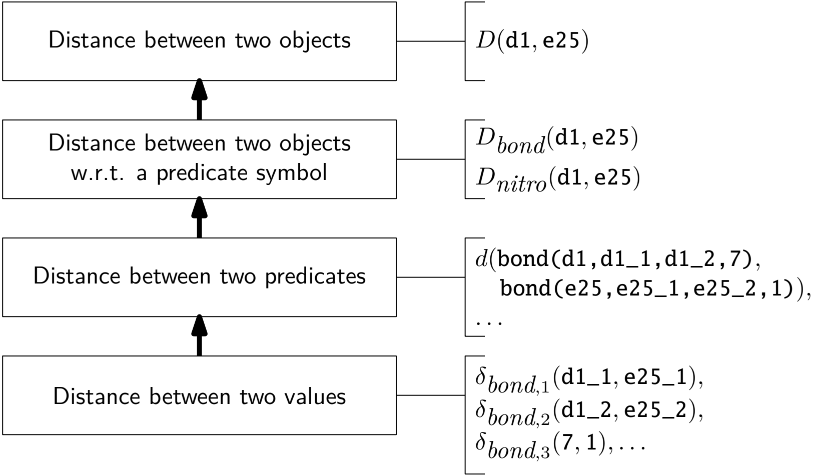
We have proposed a distance function for FOL objects, called four-layer distance function.

This function satisfies the metric properties:

- $d(x, y) = 0$ if and only if $x = y$ (coincidence axiom)
- $d(x, y) = d(y, x)$ (symmetry)
- $d(x, z) + d(z, y) \geq d(x, y), \forall z \in X$ (triangular inequality)

The properties that the function preserves the closeness of the objects.

Four-layer Distance Function



Distance between two objects

$D(\texttt{d1}, \texttt{e25})$

Distance between two objects w.r.t. a predicate symbol

$D_{bond}(\texttt{d1}, \texttt{e25})$

$D_{nitro}(\texttt{d1}, \texttt{e25})$

Distance between two predicates

$d(\texttt{bond(d1,d1\_1,d1\_2,7)},$
$\texttt{bond(e25,e25\_1,e25\_2,1)}),$
$\ldots$

Distance between two values

$\delta_{bond,1}(\texttt{d1\_1}, \texttt{e25\_1}),$
$\delta_{bond,2}(\texttt{d1\_2}, \texttt{e25\_2}),$
$\delta_{bond,3}(7, 1), \ldots$

https://bitbucket.org/fol_dist/fol4l_distance

Four-layer Distance Function

Distance between two objects

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

Distance between two objects w.r.t. a predicate symbol

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

Four-layer Distance Function

Distance between two predicates

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

where,

$$\delta_{r,i}(x_i, y_i) = \begin{cases} 0 & \text{if } x_i = y_i, \\ \Delta_{r,i}(x_i, y_i) & \text{if at most one of } x_i, y_i \text{ is an ID,} \\ D(x_i, y_i) & \text{if both } x_i, y_i \text{ are ID's.} \end{cases}$$

Four-layer Distance Function

Distance between two values

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

Distance Based Evaluation Function

We propose a modification of an evaluation function of an ILP system called ‘Aleph’. This ILP system applies the concept of inverse entailment [**?**] to construct bottom clauses, and searches for the most appropriate clauses according to a search strategy and an evaluation function, called here a coverage function. It evaluates each rule r by

$$\mathrm{coverage}(r) = p - n \tag{1}$$

where p is the number of positive examples covered by r, and n is the number of negative examples covered by r.

Distance Based Evaluation Function

It can be seen from the coverage function that

- Only the numbers of covered examples are used to justify the quality of a rule.
- Two different rules may cover different set of positive and negative examples of the same numbers.
- Distances between the covered examples can be used to differentiate between good and bad rules.

Distance Based Evaluation Function

Our idea is that a rule covering a larger area in the space of the examples performs better or yields a higher accuracy than a rule with smaller coverage area. To show a rough estimation, we find distances between all pairs of covered positive examples, and calculate the average distance. We first calculate $C(r)$ that finds the set of all possible pairs of positive examples by a rule r.

$$C(r) = \{(x, y) \mid x \in P(r),\ y \in P(r),\ x \neq y\}, \tag{2}$$

where $P(r)$ is the set of positive examples covered by r. Then, we calculate the average distance from the set $C(r)$:

$$\overline{d}(r) = \frac{1}{|C(r)|} \sum_{(x,y) \in Cr} d(x, y), \tag{3}$$

where $d(x, y)$ is a distance between two examples x and y. In this paper, we apply the four-layer distance function defined in the previous section.

Distance Based Evaluation Function

We propose two modifications of the coverage function, i.e.

1. We weight the coverage value by the value of $\overline{d}(r)$. Thus, a rule covering the same numbers of positive examples with higher average distance will be evaluated higher.

$$\text{coverage}_{mul}(r) = \overline{d}(r)(p - n) \tag{4}$$

2. We adjust the coverage value by adding $\overline{d}(r)$. This yields the similar effect as the previous modification.

$$\text{coverage}_{add}(r) = p - n + \overline{d}(r) \tag{5}$$

Distance Based Evaluation Function

We also propose a new way to rank seeds to select positive examples for bottom clause construction, whereas Aleph selects its seeds based on their order in the input file. By utilizing our distance function, we rank positive examples based on their distances from the center of positive set and the center of negative set. We employ the eccentricity function to approximate the location of each positive example. The eccentricity function measures the distance from an example to the center of the dataset. It is defined as follows:

$$E_p(x) = \left(\frac{1}{n} \sum_{y \in X} d(x,y)^p \right)^{\frac{1}{p}}, \tag{6}$$

where $1 \leq p < +\infty$, X is the dataset, and n is the cardinality of X.

Experimental Setting

We evaluate our proposed evaluation functions and positive example ordering techniques on real-world datasets, i.e. Mutagenesis dataset, and the Alzheimer dataset. The experiments are conducted using 10-fold cross validation technique.

Experimental Setting

We vary the following setting for the evaluation function and the order of positive examples:

1. The evaluation function is set to be either $\mathrm{coverage}_{mul}$ or $\mathrm{coverage}_{add}$,
2. The eccentricity values used to order the positive examples are computed against either the positive set or the negative set,
3. The eccentricity values are ordered in either ascending or descending order. The experimental results are also compared with the results from Aleph without modification and $\kappa\mathrm{FOIL}$.

Result

Table: Prediction accuracies on real-world datasets using 10-fold cross validation method (figures in boldface font indicate the best accuracy in each dataset; $E^{\oplus}$ indicates the eccentricity against the positive set, and $E^{\ominus}$ is the eccentricity against the negative set; ▲ and ▼ indicate the ascending and descending order, respectively)

Method	Mutagenesis	Alzheimer amine
Aleph	73.4 ± 11.8	70.2 ± 7.3
κFOIL	77.0 ± 4.5	89.8 ± 5.7
$\text{coverage}_{add}, E^{\oplus\blacktriangle}$	81.4 ± 7.5	75.8 ± 6.0
$\text{coverage}_{add}, E^{\oplus\blacktriangledown}$	81.9 ± 7.1	76.7 ± 6.3
$\text{coverage}_{add}, E^{\ominus\blacktriangle}$	81.4 ± 7.5	77.3 ± 7.1
$\text{coverage}_{add}, E^{\ominus\blacktriangledown}$	81.9 ± 7.1	77.4 ± 6.5
$\text{coverage}_{mul}, E^{\oplus\blacktriangle}$	78.2 ± 9.6	78.1 ± 7.3
$\text{coverage}_{mul}, E^{\oplus\blacktriangledown}$	80.3 ± 5.5	77.6 ± 6.0
$\text{coverage}_{mul}, E^{\ominus\blacktriangle}$	78.2 ± 9.6	77.7 ± 7.1
$\text{coverage}_{mul}, E^{\ominus\blacktriangledown}$	80.3 ± 5.5	77.4 ± 6.3

Result

Fig 1 shows the rules with the highest positive example coverage constructed by the proposed technique and the conventional Aleph. This can be seen that the proposed technique can obtain the rules with higher coverage.

The Proposed Technique

```
[Rule 6] [Pos cover = 59 Neg cover = 5]
active(A) :- atm(A,B,c,27,C), lteq(C,0.005).
[Rule 1] [Pos cover = 55 Neg cover = 2]
active(A) :- atm(A,B,c,27,C), lteq(C,-0.076).
[Rule 5] [Pos cover = 41 Neg cover = 5]
active(A) :- atm(A,B,c,29,C), gteq(C,0.004).
```

Aleph

```
[Rule 3] [Pos cover = 56 Neg cover = 5]
active(A) :- atm(A,B,c,27,C), lteq(C,-0.054).
[Rule 4] [Pos cover = 32 Neg cover = 5]
active(A) :- atm(A,B,c,22,C), atm(A,D,c,10,E), bond(A,B,D,1).
[Rule 1] [Pos cover = 28 Neg cover = 0]
active(A) :- atm(A,B,c,27,C), lteq(C,-0.087).
```

Figure: Top Rules obtained from the proposed technique and Aleph

Discussions

- Compared to the conventional Aleph, all settings of the proposed technique can obtain the higher accuracies.
- The proposed technique performs better than κFOIL on the Mutagenesis dataset, but it still performs worse than κFOIL on the Alzheimer dataset.
- The coverage_{add} function performs better than coverage_{mul} on the Mutagenesis dataset, but both functions performs similarly on the Alzheimer dataset.
- The eccentricity against the positive and negative sets and the order of positive examples do not affect the prediction accuracies. These issues need further investigation.

Conclusions

- We propose new coverage functions for ILP systems based on distances between examples in the dataset.
- We also present a technique to order the positive examples based on their eccentricity values, distances from the center of positive or negative sets, in order to obtain the rules with higher performance.
- This work is an attempt to investigate a link between generalization and distance between FOL objects.

Thank you very much
for your time.