Proximidad entre cláusulas en Programación Lógica Inductiva

M.A. Gutiérrez Naranjo J.A. Alonso Jiménez * J. Borrego Díaz *
Dpto. Ciencias de la Computación e Inteligencia Artificial – Universidad de Sevilla
E–mail: {magutier,jalonso,jborrego}@cica.es
WWW: http://www-cs.us.es/{~naranjo, ~jalonso, ~jborrego}

Abstract

En este artículo estudiamos la idea de proximidad en el conjunto de cláusulas de un lenguaje donde dos cláusulas equivalentes por subsunción se consideran la misma. La formalización de proximidad que presentamos está basada en una quasi–métrica (una métrica en la que no consideramos la condición de simetría) $dc : \mathbb{C}_R/\sim \times \mathbb{C}_R/\sim \rightarrow [0, +\infty]$ donde $\mathbb{C}/\sim$ es el espacio cociente obtenido a partir del conjunto de cláusulas $\mathbb{C}$ por la relación de equivalencia basada en subsunción.

Palabras clave: Programación Lógica Inductiva, Quasi-métrica

1 Introducción

En Aprendizaje Automático hay una creciente necesidad de formalizar el concepto de proximidad en espacios cada vez más abstractos. En Programación Lógica Inductiva (ILP), el problema de cuantificar la proximidad entre cláusulas ya ha sido estudiado con anterioridad por A. Hutchinson [4] y S.–H.Nienhuys–Cheng [6], ofreciendo distintas alternativas de solución al problema. En ambos casos se define primero una distancia entre literales y luego se usa la métrica de Hausdorff para obtener a partir de esta una distancia entre cláusulas. Esto tiene dos desventajas. Por un lado la métrica de Hausdorff depende exclusivamente de los puntos extremos (ver [1]) y por otro, estos literales se consideran aislados y en ningún momento se consideran las posibles relaciones entre los literales de la misma cláusula.

En este artículo, proponemos una solución al problema de cuantificar la proximidad entre cláusulas, considerándolas como elementos de un entramado de relaciones vía subsunción que nos va a permitir acceder de una cláusula a otra, en cierto sentido, por el camino más corto. Esta aproximación representa una importante diferencia con [4] y [6], que consideran los literales como elementos aislados. Para ello, proponemos

*Parcialmente financiados por DGES, proyectos PB96–0098-C04–04 y PB96–1345

1. Que dos cláusulas equivalentes bajo subsunción se consideren idénticas. Este planteamiento es mucho más fuerte que la equivalencia módulo renombramiento y nos va a permitir definir nuestra función sobre clases de equivalencia.
2. Siguiendo la intuición geométrica, la distancia entre dos puntos será la longitud del camino más corto entre ellas, considerando que dos cláusulas están a distancia infinita si no existe un camino que las una.

2 Cláusulas

A continuación recordamos algunas definiciones sobre cláusulas que usaremos más adelante. Una visión general puede obtenerse en [7].

En nuestra construcción consideraremos un lenguaje $\mathcal{L}$ de primer orden. Var y $Term$ son, respectivamente, los conjuntos de variables y términos de $\mathcal{L}$. Una *cláusula* es un conjunto finito de literales y $\mathbb{C}$ es el conjunto de las cláusulas del lenguaje.

Sea $S \subseteq Var$ un conjunto finito de variables. Una *sustitución* es una aplicación $\theta : S \to Term$ tal que $(\forall x \in S)[x \neq x\theta]$. Un *renombramiento* es una sustitución inyectiva θ tal que $(\forall x \in S)\,[x\theta \in Var]$. Si C es una cláusula y θ es un renombramiento, C y $C\theta = \{L\theta \,|\, L \in C\}$ son *variantes.*

Sean C y D dos cláusulas. C *subsume* a D, $C \succeq D$, si existe una sustitución θ tal que $C\theta \subseteq D$. Si $C \succeq D$ y $D \succeq C$ entonces C y D son equivalentes por subsunción y lo escribiremos $C \sim D$. Si $C \succeq D$ y $D \not\succeq C$ escribiremos $C \succ D$. Una *cláusula reducida* es una cláusula C tal que no tiene ningún subconjunto propio D tal que $D \sim C$. Plotkin [9] probó que dos cláusulas reducidas equivalentes son variantes.

Puesto que $\sim$ es una relación de equivalencia, denotaremos por $\mathbb{C}/\!\sim$ el espacio cociente y, si $C \in \mathbb{C}$, $[C] = \{D \in \mathbb{C} \,|\, C \sim D\}$. Definimos el orden parcial $\succeq^*$ sobre $\mathbb{C}/\sim$ como $(\forall [C], [D] \in \mathbb{C}/\!\sim)\,([C] \succeq^* [D] \Leftrightarrow C \succeq D)$ El orden $\succeq^*$ está bien definido y no causará confusión si usamos $\succeq$ en lugar de $\succeq^*$.

3 ILP

La Programación Lógica Inductiva (ILP) puede definirse como el área de investigación en la intersección del Aprendizaje Automático y la Lógica Computacional cuyo principal objetivo es el desarrollo de teorías y algoritmos prácticos para el aprendizaje inductivo de programas lógicos (N. Lavrač y L. De Raedt, 1995).

Del Aprendizaje Automático toma sus objetivos, esto es, la síntesis de conocimientos a partir de la experiencia. En este contexto, el aprendizaje de conceptos intenta obtener *definiciones* de conceptos a partir de instancias positivas (ejemplos que verifican la propiedad que queremos definir) e instancias negativas (ejemplos que no la verifican) con la intención de obtener una clasificación de las instancias observadas así como de predecir la posible clasificación de instancias no observadas.

De la Lógica Computacional, la ILP toma su representación formal, su orientación semántica y sus técnicas. La definición de un concepto se representa mediante un programa lógico, que no es más que un conjunto finito ordenado de cláusulas definidas, que puede ser visto como el conjunto de axiomas de una teoría. Si nuestra *definición* (i.e. nuestro programa lógico) es demasiado general, esto es, engloba ejemplos que no deseamos, debemos *especializarlo*. Si por el contrario es demasiado específico, esto es, deja fuera instancias que debería considerar, entonces debe ser generalizado. Estas *generalizaciones* y *especializaciones* se realizan aplicando a alguna cláusula del programa un operador adecuado. De este modo se espera que tras la sucesiva aplicación de operadores la sucesión de programas converja a uno que cubra todos los ejemplos positivos y ninguno de los negativos. En [8] Nienhuys–Cheng y Van der Laag definieron un operador de refinamiento ρ_r que tomaba como dato de entrada una cláusula reducida C y devolvía un conjunto de cláusulas reducidas $\rho_r(C)$:

Definición 1 (Adaptada de [8]) *Sea C una cláusula reducida. Entonces $D \in \rho_r(C)$ si D es reducida y se verifica una de las siguientes condiciones:*

- ρ_r^1: *$C \succ D$ y existen dos cláusulas $C' \in [C]$ y $D' \in [D]$ tales que $C'\theta = D'$, donde θ es la sustitución $\theta = \{x/f(y_1, \ldots, y_n)\}$, f es un símbolo de función*[1]*, x ocurre en C', y las variables $y_1, \ldots, y_n$ son variables distintas que no ocurren en C'.*
- ρ_r^2: *$C \succ D$ y existen dos cláusulas $C' \in [C]$ y $D' \in [D]$ tales que $C'\theta = D'$, donde $\theta = \{x/y\}$ y además las variables x e y ocurren en C'.*
- ρ_r^3: *$D = C \cup \{L\}$ donde L sólo tiene variables distintas que no ocurren en C y para todo literal $M \in C$, L difiere de M en el símbolo de predicado o en el signo.*

Una ρ_r–cadena de longitud n de C a D es una sucesión finita $C = C_0, C_1, \ldots, C_n = D$ tal que para todo $i \in \{1, \ldots, n\}$, $C_i \in \rho(C_{i-1})$.

Por ejemplo, consideremos la cláusula

$$encendida(x) \leftarrow bombilla(x), cable(x, y), corriente(y)$$

que expresada como conjunto de literales es $C = \{encendida(x), \neg bombilla(x), \neg cable(x, y), \neg corriente(y)\}$. Si esta cláusula fuera demasido general podemos hacerla más específica aplicando alguno de los operadores de refinamiento.

Por el operador ρ_r^1, si aplicamos la sustitución $\theta_1 = \{y/bateria\}$, donde $bateria$ es un símbolo de función de aridad cero, tendríamos la cláusula $C = \{encendida(x), \neg bombilla(x), \neg cable(x, bateria), \neg corriente(bateria)\}$.

Por el operador ρ_r^2, si aplicamos la sustitución $\theta_2 = \{y/x\}$, tendríamos la cláusula $C = \{encendida(x), \neg bombilla(x), \neg cable(x, x), \neg corriente(x)\}$.
El operador ρ_r^3 es más técnico. Nos permite añadir literales nuevos como $no\_fundida(z)$ y conseguir cláusulas como $C_1 = \{encendida(x), \neg bombilla(x), \neg cable(x, y), \neg corriente(y), no\_fundida(z)\}$, para después aplicar otros operadores, por ejemplo ρ_r^3 mediante la sustitución la sustitución $\theta_3 = \{z/x\}$ y obtener $C_1\theta_3 = \{encendida(x), \neg bombilla(x), \neg cable(x, y), \neg corriente(y), no\_fu$

4 Quasi–métricas

Las funciones de distancia no simétricas ya fueron consideradas por Hausdorff [3] a principios de siglo. Wilson [12] introdujo el término *quasi–metrics* para estas funciones en 1931. A

[1] Consideramos las constantes como símbolos de función de aridad cero.

lo largo del siglo diversos investigadores han contribuido al desarrollo de las distancias no simétricas, recibiendo recientemente un nuevo empuje con los trabajos en computación teórica de Lawson [5] o Smyth [11] entre otros.

Definición 2 ([11]) *Una quasi–métrica sobre un conjunto X es una aplicación de $X \times X$ en los reales no negativos, incluyendo posiblemente $+\infty$ tal que*

- $(\forall x \in X)\,[d(x,x)=0]$
- $(\forall x,y \in X)\,[d(x,y)=d(y,x)=0 \Rightarrow x=y]$
- $(\forall x,y,z \in X)\,[d(x,z)\leq d(x,y)+d(y,z)]$

Nótese que una quasi–métrica verifica las condiciones de métrica de Fréchet [2], excepto la condición de simetría. Veamos algunos ejemplos (otros ejemplos más sofisticados pueden encontrarse en [11]).

Ejemplo 1: Dado cualquier conjunto parcialmente ordenado $\langle P, \leq\rangle$, la *quasi–métrica discreta* se define como

$$d(x,y)=\begin{cases} 0 & \text{si } x\leq y \\ 1 & \text{e.o.c.}\end{cases}$$

Ejemplo 2: En el intervalo unidad $[0,1]$ podemos definir la quasi–métrica siguiente, cuya métrica asociada es la distancia euclídea en el conjunto.

$$d(x,y)=\begin{cases} 0 & \text{si } x\leq y \\ x-y & \text{si } y<x\end{cases}$$

5 Una quasi–métrica sobre las clases de equivalencia

En [8] Nienhuys-Cheng y y Van der Laag probaron que si C y D son cláusulas reducidas y $C \succ D$ entonces existe una ρ_r*–cadena* de C a D. Vamos a usar esas cadenas para formalizar la proximidad entre cláusulas. En nuestra definición, la distancia entre dos cláusulas vendrá determinada por la longitud del *camino* más corto entre ellas, considerando como camino la sucesión de clases de equivalencia asociada a una ρ_r–cadena.

Definición 3 *Diremos que la sucesión $\mathcal{C}=\langle[C_0],\ldots,[C_n]\rangle$, con $[C_i]\in\mathbb{C}/\sim$ para todo $i\in\{0,\ldots,n\}$ es una L–cadena de $[C_0]$ a $[C_n]$ si podemos elegir como representantes de dichas clases las cláusulas reducidas $C_0,\ldots,C_n$ y dichas cláusulas forman una ρ_r–cadena. En ese caso diremos que la L–cadena $\mathcal{C}$ tiene longitud n y lo denotaremos por $|\mathcal{C}|=n$. Denotaremos como $\mathbf{L}([C],[D])$ el conjunto de todas las L–cadenas de $[C]$ a $[D]$. El único elemento de $\mathbf{L}([C],[C])$ es la sucesión de longitud cero $\langle[C]\rangle$.*

Es trivial comprobar que si $\mathcal{C}_1=\langle[C_0],\ldots,[C_n]\rangle$ es una L–cadena de $[C_0]$ a $[C_n]$ y $\mathcal{C}_2=\langle[D_0],\ldots,[D_m]\rangle$ es una L–cadena de $[D_0]$ a $[D_m]$ con $[C_n]=[D_0]$, entonces

$$\mathcal{C}_3=\langle[C_0],\ldots,[C_n],[D_1],\ldots,[D_m]\rangle$$

es una L–cadena de $[C_0]$ a $[D_m]$ de longitud $n+m$ que llamaremos la *concatenación* de $\mathcal{C}_1$ y $\mathcal{C}_2$. Sabemos que si C y D son cláusulas reducidas y $C\succ D$, entonces existe una ρ_r–cadena de C a D. Como consecuencia inmediata tenemos el siguiente teorema:

Teorema 4 *Consideremos* $[C],[D] \in \mathbb{C}/\sim$ *tales que* $[C] \succ [D]$. *Entonces existe una L–cadena de* $[C]$ *a* $[D]$.

Demostración: Sean $[C],[D] \in \mathbb{C}/\sim$ tales clases de equivalencia y sean C' y D' dos cláusulas reducidas tales que $C' \in [C]$ y $D' \in [D]$. Puesto que $C' \succ D'$, se tiene que existe una ρ_r–cadena de C' a D'. Las clases de equivalencia asociadas a los elementos de la ρ_r–cadena forman una L–cadena de $[C]$ a $[D]$. ■

A continuación definimos nuestra quasi–métrica. Si $[C] \succeq [D]$ entonces existe al menos una L–cadena de $[C]$ a $[D]$, (el conjunto $\mathbf{L}([C],[D])$ no es vacío) y tiene sentido considerar el mínimo del conjunto de longitudes de caminos en $\mathbf{L}([C],[D])$.

Siguiendo la intuición geométrica, si consideramos esas L–cadenas como *caminos* de $[C]$ a $[D]$, podemos definir nuestra quasi–métrica como la longitud del camino más corto de $[C]$ a $[D]$. Si no existe ningún camino, pensamos que $[D]$ no puede ser alcanzado desde $[C]$, así que están separados por una distancia infinita.

Definición 5 *Definimos la aplicación* $dc : \mathbb{C}/\sim \times \mathbb{C}/\sim \to [0,+\infty]$ *de la siguiente manera*

$$dc([C],[D]) = \begin{cases} min\{|\mathcal{C}| \, : \, \mathcal{C} \in \mathbf{L}([C],[D])\} & si\ [C] \succeq [D] \\ +\infty & e.o.c. \end{cases}$$

Teorema 6 *dc es una quasi–métrica*

Demostración: (1) Puesto que $[C] \succeq [C]$ para todo $[C] \in \mathbb{C}/\sim$, se tiene que la L–cadena $\langle [C] \rangle \in \mathbf{L}([C],[C])$. Además $|\langle [C] \rangle| = 0$, luego $dc([C],[C]) = 0$.
(2) Si $dc([C],[D]) = dc([D],[C]) = 0$, entonces $[C] \sim [D]$ y por tanto $[C] = [D]$.
(3) Tenemos que probar que $dc([C_1],[C_3]) \leq dc([C_1],[C_2]) + dc([C_2],[C_3])$. Si $[C_1] \not\succeq [C_2]$ o $[C_2] \not\succeq [C_3]$ el resultado se tiene trivialmente, luego supongamos $[C_1] \succeq [C_2]$ y $[C_2] \succeq [C_3]$. Sean $\mathcal{C}_1 = \langle [D_0], \ldots, [D_n] \rangle$ una L–cadena de $[C_1]$ a $[C_2]$ (esto es, $[D_0] = [C_1]$ y $[D_n] = [C_2]$) tal que $n = |\mathcal{C}_1| = dc([C_1],[C_2])$ y $\mathcal{C}_2 = \langle [D'_0], \ldots, [D'_m] \rangle$ una L–cadena de $[C_2]$ a $[C_3]$ (i.e., $[D'_0] = [C_2]$ y $[D_m] = [C_3]$) tal que $m = |\mathcal{C}_2| = dc([C_2],[C_3])$ Si concatenamos $\mathcal{C}_1$ y $\mathcal{C}_2$ obtenemos $\mathcal{C}_{12} = \langle [D_0], \ldots, [D_n], [D'_1], \ldots, [D'_m] \rangle$ que es una L–cadena de $[C_1]$ a $[C_3]$ de longitud $n+m$, luego

$$\begin{aligned} d([C_1],[C_3]) &\leq |\mathcal{C}_{12}| \\ &= n + m \\ &= d([C_1],[C_2]) + d([C_2],[C_3]) \end{aligned}$$

■

Por tanto es una quasi–métrica. Si ahora volvemos a considerar las cláusulas aisladas y no las clases de equivalencia tendremos una función

$$\begin{array}{llll} \widehat{dc} : & \mathbb{C} \times \mathbb{C} & \longrightarrow & [0,+\infty] \\ & \langle C, D \rangle & \mapsto & \widehat{dc}(C,D) = dc([C],[D]) \end{array}$$

en la cual dos cláusulas equivalentes están a distancia cero (técnicamente una pseudo–quasi–distancia) en la cual mantenemos las siguientes propiedades:

1. $\widehat{dc}(C,D)=0 \Leftrightarrow C \sim D$
2. $\widehat{dc}(C,D)=+\infty \Leftrightarrow C \not\succeq D$
3. $(\forall C_1, C_2, C_3 \in \mathbb{C})\,[\widehat{dc}(C_1,C_3) \leq \widehat{dc}(C_1,C_2)+\widehat{dc}(C_2,C_3)]$

Pensamos que de esta manera, $\widehat{dc}(C,D)$ codifica de manera numérica suficiente información sobre la relación de subsunción entre C y D y permite un tratamiento algebraico de la relación de proximidad.

6 Trabajos relacionados

Como apuntábamos en la introducción, en la literatura puede encontrarse diversas aproximaciones al problema de cuantificar la relación de proximidad entre cláusulas. Nuestra propuesta se suma al esfuerzo de arrojar luz sobre el problema.

6.1 Nienhuys–Cheng [6] y Ramon y Bruynooghe [10]

En [6], Nienhuys-Cheng define una distancia para átomos cerrados

- $d_{nc,g}(e,e)=0$
- $p/n \neq q/m \Rightarrow d_{nc,g}(p(s_1,\dots,s_n),q(t_1,\dots,t_m))=1$
- $d_{nc,g}(p(s_1,\dots,s_n),p(t_1,\dots,t_n))=\frac{1}{2n}\sum_{i=1}^{n} d_{nc,g}(s_i,t_i)$

y luego considera la métrica de Hausdorff para trasladar esa distancia a conjuntos de átomos.

$$d_h(A,B)=\max\{\max_{a\in A}\{\min\{d_{nc,g}(a,b)\mid b\in B\}\},\max_{b\in B}\{\min\{d_{nc,g}(a,b)\mid a\in A\}\}\}$$

El objetivo de esta distancia era definir una métrica entre interpretaciones de Herbrand, así que $d_{nc,g}$ estaba sólo definida sobre átomos cerrados. En [10], Ramon y Bruynooghe extendieron esta distancia a una función sobre expresiones cerradas y no cerradas:

- $d_{nc}(e_1,e_2)=d_{nc,g}(e_1,e_2)$ si e_1,e_2 son expresiones cerradas.
- $d_{nc}(p(s_1,\dots,s_n),X)=d_{nc}(X,p(s_1,\dots,s_n))=1$ con X una variable.
- $d_{nc}(X,Y)=1$ y $d_{nc}(X,X)=0$ para todo $X \neq Y$ con X e Y variables.

Aplicando a d_{nc} la métrica de Hausdorff tenemos una distancia sobre cláusulas, como muestra el siguiente ejemplo

$$\begin{array}{llll} C_1 = & \{p(f(U),X,f(a))\} & C_2 = & \{p(f(a),X,f(a))\} \\ C_3 = & \{p(f(a),X,f(a)),p(Z,X,Z)\} & C_4 = & \{p(f(a),X,f(a)),p(f(a),V,f(a))\} \end{array}$$

con $d_h(C_1, C_2) = \frac{1}{12}$, $d_h(C_1, C_3) = \frac{1}{3}$, $d_h(C_1, C_4) = \frac{1}{4}$. Se observa que los tres valores son muy distintos a pesar de que C_2, C_3 y C_4 son equivalentes bajo subsunción. Con nuestra función se tiene

$$\widehat{dc}(C_1, C_2) = \widehat{dc}(C_1, C_3) = \widehat{dc}(C_1, C_4) = 1$$

puesto que $[C_2] = [C_3] = [C_4]$ con $[C_1] \neq [C_2]$ y $C_1\theta = C_2$ con $\theta = \{U/a\}$.

6.2 Hutchinson [4]

En [4], Hutchinson da una pseudo-métrica sobre el conjunto de términos y la extiende al conjunto de literales. Entonces, considera la métrica de Hausdorff sobre el conjunto de cláusulas usando su pseudo–métrica sobre literales.

En su definición de distancia sobre términos, usa una función del conjunto de sustituciones sobre $\mathbb{R}$ llamada *size*. Da las condiciones que tiene que satisfacer una función para ser una *size* y da una función concreta con esas características

$$S(\theta) = \sum \{w_{f/n} \mid (\exists x)\, (x \in Var \text{ y } f/n \text{ ocurre en } x\theta)\}$$

donde $w_{f/n}$ es un peso positivo para el símbolo de función f/n. Con la métrica de Hausdorff basada en esa pseudo–métrica tenemos que para las cláusulas

$$C_1 = \{p(X, X, Y, Y)\} \qquad C_2 = \{p(U, V, U, V)\}$$

obtenemos los valores $d_h(C_1, C_1) = 0$ y $d_h(C_1, C_2) = 0$ a pesar de que C_1 y C_2 no son ni siquiera comparables bajo subsunción.

Con nuestra función $\widehat{dc}$, al no existir ningún camino de C_1 a C_2, esa relación de inaccesibilidad se codifica con el símbolo $+\infty$.

$$\widehat{dc}(C_1, C_2) = \widehat{dc}(C_2, C_1) = +\infty$$

7 Conclusiones

Este es un trabajo preliminar sobre cómo cuantificar la relación de proximidad entre cláusulas y se suma a otras aproximaciones en un intento de arrojar luz sobre el problema. La idea de nuestra aproximación es aprovechar la relación preexistente entre las cláusulas para definir una función de manera natural. Al ser esta relación de subsunción no simétrica, esta formalización de la proximidad tampoco tiene por qué serlo.

Consideramos que definir una distancia entre cláusulas aplicando la métrica de Hausdorff sobre una distancia entre literales quizá no sea lo más acertado ya que depende exclusivamente de valores extremos.

Por otro lado, quizá la formalización de distancia de Fréchet [2] sea demasiado estricta para espacios donde la principal relación es la de orden parcial. En este sentido, pensamos que este artículo abre una puerta a esa nueva concepción en la formalización de proximidad.

Nuestra investigación se centra en establecer criterios de proximidad en el conjunto de clásulas y estudiar sus propiedades. Para ello debemos dotar al conjunto de cláusulas de una topología apropiada y considerar los operadores de generalización (y especialización) como funciones del conjunto de cláusulas en sí mismo.

Esta formalización se fundamenta en un estudio topológico de operadores entre programas lógicos (o subconjuntos de ellos) y permitirá una mejor comprensión del concepto de proximidad más allá de los espacios métricos y esperamos que ayude a mejorar los algoritmos de búsqueda de soluciones en ILP.

Referencias

[1] T. Eiter and H. Mannila: *Distance Measures for Point Sets and Their Computation.* Acta Informatica 34, 2, pp.: 109–133, 1997.

[2] M. Fréchet. *Sur quelques points du calcul fonctionnel.* Reudicont del Circulo Matematico di Palermo, vol 22, 1906.

[3] F. Hausdorff. *Grundzüge der Mengenlehre.* Leipzig, 1914.

[4] A. Hutchinson. *Metrics on Terms and Clauses.* Proc. ECML–97 Prague April 1997 (Springer). `ftp.//ftp.dcs.kcl.ac.uk/pub/ tech-reports/ tr96-11.ps.gz`

[5] J.D. Lawson. *Order and strongly sober compactifications.* In. G.M. Reed, A.W. Roscoe and R.F. Wachter (Eds.), Topology and Category Theory in Computer Science, Oxford University Press, pp. 179–205, 1991.

[6] S-H. Nienhuys-Cheng. *Distance between Herbrand interpretations. a measure for approximations to a target concept.* Technical Report EUR–FEW–CS–97–05. Department of Computer Science, Erasmus University, the Netherlands, 1997. `www.few.eur.nl/few /research/pubs/cs/1997/eur-few-cs-97-05.pdf`

[7] S-H. Nienhuys-Cheng and R. de Wolf. *Foundations of Inductive Logic Programming.* LNCS 1228. Springer, 1997

[8] P.R.J. van der Laag, S.-H. Nienhuys-Cheng. *Completeness and properness of refinement operators in Inductive Logic Programming.* Journal of Logic Programming, Vol 34, n.3, pp.. 201–225, March 1998

[9] G.D. Plotkin. *A Note on Inductive Generalization.* In Machine Intelligence 5, pp.. 153–163. Edinburgh University Press, Edinburgh, 1970.

[10] J. Ramon and M. Bruynooghe. *A framework for defining distances between first–order logic–objects.* Report CW 263, Department of Computer Science, Katholieke Universiteit Leuven, May 1998. `http.//www.cs.kuleuven.ac.be/publicaties/rapporten /cw/CW263.ps.gz`

[11] M.B. Smyth. *Totally bounded spaces and compact ordered spaces as domains of computation.* In. G.M. Reed, A.W. Roscoe and R.F. Wachter (Eds.), Topology and Category Theory in Computer Science, Oxford University Press, pp. 207-229, 1991.

[12] W.A. Wilson. *On quasi–metric spaces.* Amer. J. Math. 53, pp. 675–684, 1931.