Fondamenti di Teoria delle Basi di Dati

Riccardo Torlone

Parte 3: Calcolo su domini

Calcolo relazionale

- Una famiglia di linguaggi dichiarativi, basati sulla logica del primo ordine
- Diverse versioni:
  - calcolo relazionale su domini
  - calcolo su tuple con dichiarazioni di range

Richiami di logica

- *Logic* is the science of correct reasoning, i.e., reasoning based on correct (sound) arguments.
- Logical formalisms are applied in many areas of science as a basis for clarifying and formalizing reasoning.
- A logic is defined by the (family of) language(s) it uses and by its underlying reasoning machinery.
- *A logical language* is a formal language, reflecting natural language phenomena and allowing one to formulate sentences (called *formulas*) about a particular domain of interest.

Elements of logical language

- individual constants (constants, for short), representing particular objects,
  - 0, 1, John
- variables, representing a range of objects, sets of objects, sets of sets of objects, etc.
  - x, y, m, n
- function symbols, representing functions,
  - +, $*$, father()
- relation symbols, representing relations,
  - =,≤,,Older, Smaller
- logical constants:
  - True, False, sometimes also other such as Unknown, Inconsistent
- connectives and operators, allowing one to form more complex formulas from simpler formulas,
  - connectives: “and”, “or”, “implies”,
  - operators: “for all”, “exists”, “is necessary”,
- auxiliary symbols,
  - “(”, “)”, “[”, “]”.

Terms and formulas

- Expressions formed using individual constants, variables and function symbols, are called terms.
  - Intuitively, a term represents a value of the underlying domain.
  - A *ground term* is a term without variables
  - *mother*(*father*(*John*))
- Expressions formed using constants, variables, terms and relation symbols, are called formulas.
  - Intuitively, a formula represents an expression resulting in a logical value.
  - A ground formula is formula built from ground terms only
  - A closed formula involves quantified (bounded) variables
  - An open formula involves (free) variables

Classi di linguaggi

- In propositional logics (zero-order logics) the building blocks are logical constants, connectives and operators.
- First-order logics contain, in addition, variables ranging over domain elements and allow quantifiers binding such variables.
- Second-order logics contain, in addition, variables ranging over sets of domain elements (called also second-order variables) and allow quantifiers binding such variables. Alternatively, second order variables can range over relations or functions
- Third-order logics contain, in addition, variables ranging over sets of sets of domain elements (called also third-order variables) and allow quantifiers binding such variables.
- ...

Semantica: approccio model theoretic

- In the model theoretic approach we attach meaning ("real entities") to symbols:
  - objects to constants
  - range of objects to variables
  - functions to function symbols
  - relations to relation symbols.
  - The meaning of connectives, operators and auxiliary symbols is fixed by a given logic.
- This meaning allows us to check whether
  - a closed formula is true or false
  - for which values an open formula is true or false

Semantica: approccio proof theoretic

- In the proof theoretic *approach* we fix axioms and proof rules (rules, for short).
  - Axioms are facts “obviously true” in a given reality.
  - Rules allow us to infer new facts on the basis of known facts (axioms, facts derived from axioms, etc.).
  - Axioms together with proof rules are called proof systems
- A proof system allows us to check whether a formula can be implied by others (in particular, by the axioms)

Calcolo relazionale

- Approccio model theoretic
- Differenze rispetto al logica del primo ordine:
  - le costanti non vengono interpretate
  - simboli di predicato
    - relazioni nella base di dati
    - predicati "standard" predefiniti (=, >, ...)
  - non ci sono simboli di funzione
  - utilizziamo notazione non posizionale
- Si usano:
  - formule aperte per specificare query
  - formule chiuse per specificare vincoli

Calcolo su domini, sintassi e semantica

- Le espressioni hanno la forma:

$$\{ A_1: x_1, \ldots, A_k: x_k \mid f \}$$

  - f è una formula (con connettivi booleani e quantificatori)
  - $A_1: x_1, \ldots, A_k: x_k$ è la target list
    - $A_1, \ldots, A_k$ attributi distinti (anche non nella base di dati)
    - $x_1, \ldots, x_k$ variabili distinte
- Semantica intuitiva: il risultato è una relazione su $A_1, \ldots, A_k$ che contiene tuple di valori per $x_1, \ldots, x_k$ che rendono vera la formula f

Sintassi più rigorosa

- Siano D un insieme di costanti e V un insieme di variabili
- Formule su **R**={$R_1(X_1),\ldots,R_n(X_n)$} possono essere:
  - Atomi del tipo
    - $R(A_1{:}x_1,\ldots, A_k{:}x_k)$, dove $R(A_1,\ldots, A_k)\in$ **R** e $x_1,\ldots,x_k$ sono variabili in V dette *libere*
    - $x_1 \Theta x_2$ oppure $x \Theta c$, dove x_1 e x_2 sono variabili libere in V**,** *a* è una costante in D e Θ è un simbolo di confronto
  - Se *f* è una formula su **R** allora
    - $\exists x(f)$ e $\forall x(f)$ sono formule su **R,** le occorrenze di x in f sono dette *legate*, le occorrenze delle altre variabili sono libere (legate) se sono libere (legate) in *f*
  - Se f_1 ed f_2 sono formule su **R** allora
    - $\neg f_1$, $f_1 \wedge f_2$, $f_1 \vee f_2$ sono formule su **R,** le occorrenze delle variabili sono libere (legate) se sono libere (legate) in f_1 o f_2

Semantica più rigorosa

- Sia s una funzione totale $s : V \rightarrow D$ ed $\mathbf{r} = \{r_1,\ldots,r_n\}$ una istanza di $\mathbf{R} = \{R_1(X_1),\ldots,R_n(X_n)\}$
  - $R(A_1:x_1,\ldots, A_k:x_k)$ è vera per *s* se esiste una tupla *t* in *r* tale che $t[A_1] = s(x_1)$ per $1 \leq i \leq k$
  - $x_1 \Theta x_2$ $(x \Theta c)$ è vera per *s* se $s(x_1) \Theta s(x_2)$ $(s(x) \Theta c)$
  - $\exists x(f)$ è vera per *s* se *f* è vera per almeno una sostituzione *s′* diversa da *s* al più su *x*
  - $\forall x(f)$ è vera per *s* se *f* è vera per ogni sostituzione *s′* diversa da *s* al più su *x*
  - $\neg f_1$, $f_1 \wedge f_2$, $f_1 \vee f_2$ sono vere su *s* se valgono le regole dei connettivi logici su f_1 su *s* ed f_2 su *s*

Semantica di una espressione del calcolo

- Il valore di un'espressione:

$$\{ A_1: x_1, \ldots, A_k: x_k \mid f \}$$

- È l'insieme di tuple:

{ t | esiste una sostituzione s tale che f è vera su s

e $t[A_1] = s(x_1)$ per $1 \leq i \leq k$ }

Base di dati per gli esempi

Impiegati(Matricola,Nome, Età, Stipendio)

Supervisione(Capo, Impiegato)

Impiegati

Matricola	Nome	Età	Stipendio
7309	Rossi	26	55
5998	Neri	34	64
9553	Milano	47	44
5698	Neri	52	74

Supervisione

Capo	Impiegato
5998	7309
5698	9553

Esempio 1

- Trovare matricola, nome ed età degli impiegati che guadagnano più di 60 mila

$$\pi_{\text{Matricola, Nome, Età}}(\sigma_{\text{Stipendio}>60}(\text{Impiegati}))$$

$$\{ \text{Matricola: m, Nome: n, Età: e} \mid \exists t(\text{Impiegati(Matricola: m, Nome: n, Età: e, Stipendio: t)} \wedge t > 60) \}$$

Interpretazione della formula

{ Matricola: m, Nome: n, Età: e |
∃t(Impiegati(Matricola: m, Nome: n, Età: e,
Stipendio: t) ∧ t > 60) }

Impiegati

Matricola	Nome	Età	Stipendio
7309	Rossi	26	55
5998	Neri	34	64
9553	Milano	47	44
5698	Neri	52	64

- Sostituzioni che rendono vera la formula
  - $s_1(m)= 5998$, $s_1(n)=$ Neri, $s_1(e)= 34$, $s'_1(t)= 64$
  - $s_2(m)= 5698$, $s_2(n)=$ Neri, $s_2(e)= 52$, $s'_2(t)= 64$

Variante dell'esempio 1

- Trovare matricola, nome ed età degli impiegati che guadagnano più di 60 mila

$$\{ \text{Matricola: m, Nome: n, Età: e} \mid \exists t(\text{Impiegati(Matricola: m, Nome: n, Età: e, Stipendio: t)} \wedge t > 60) \}$$

$$\{ \text{Matricola: m, Nome: n, Età: e} \mid \text{Impiegati(Matricola: m, Nome: n, Età: e, Stipendio: t)} \wedge t > 60 \}$$

Variabili libere e target list

- **Lemma** Per ogni espressione del CRD esiste una espressione equivalente le cui uniche variabili libere sono quelle che compaiono nella target list
- **Proof** Il valore di una formula con quantificatore $\exists x(f)$ o $\forall x(f)$ su una sostituzione s non dipende dal valore di s sulla variabile x. Per induzione si può quindi dimostrare che il valore di una formula dipende solo dai valori di s sulle variabili libere della formula

Esempio 2

- Trovare le matricole dei capi degli impiegati che guadagnano più di 40 mila

$$\pi_{Capo}\,(Supervisione\ JOIN_{Impiegato=Matricola}\ (\sigma_{Stipendio>60}(Impiegati)))$$

$$\{\ Capo: c \mid Supervisione(Capo:c, Impiegato:m) \wedge Impiegati(Matricola: m, Nome: n, Età: e, Stipendio: s) \wedge s > 60\ \}$$

Esercizi

- Con riferimento alla base di dati di esempio, formulare nel calcolo sui domini le seguenti interrogazioni:
  - Trovare il cognome del capo di Rossi
  - Trovare il cognome dei capi con età minore di 45
  - Trovare nome e stipendio dei capi degli impiegati che guadagnano più di 50 mila
  - Trovare matricola e nome dei capi i cui impiegati guadagnano tutti più di 50 mila