Le forme normali

Università di Pisa

2021

Contenti

Introduzione

Obbiettivi

- valutare la qualità della progettazione degli schemi relazionali
- conservazione dell'informazione
- minimizzazione della ridondanza

Approccio top-down: raggruppamenti di attributi analizzati e successivamente decomposti.

Linee guida

1. **Semplice è bello**
   Non raggruppare attributi da più tipi di entità/relazione in un'unica relazione.
2. **No alle anomalie**
   Certificare che i programmi di inserimento, cancellazione o modifica funzionano sempre.
3. **Evitare valori nulli**
   Se inevitabili, assicursi che sono pochi rispetto al numero di tuple di una relazione.

Esempio: LG1

- Per mantenere semplicità semanticare, far corrispondere:
  un schema di relazione $\rightarrow$ un solo tipo di entità

Esempio: LG2

Anagrafe(CF, NomeP, Indirizzo, NomeC, NumAb)

Semantica attributi:

- CF determina NomeP, Indirizzo e NomeC
- NomeC determina NumAb

Considerazioni:

- NumAb ripetuto per tuple con lo stesso NomeC
  $\Longrightarrow$ da mantenere consistenza
- evitare problema trasformando lo schema in due schemi:
  - Persona(CF, NomeP, Indirizzo, NomeC)
  - Residenza(NomeC, NumAb)

Formalizzazione

Dipendenza funzionale: FD

Definizione

Esprime un legame semantico tra due gruppi di attributi di uno schema R.

- è una proprietà di R, non di un particolare stato valido r di R
- non può essere dedotta a partire da uno stato valido r
- deve essere definita esplicitamente da qualcuno che conosce la semantica degli attributi di R

Forme normali

- esistono diversi tipi, ciascuno:
  - garantisce l'assenza di determinati difetti in R
  - quindi definisce un determinato livello di qualità di R
- possibile eseguire una serie di test per certificare che R soddisfa una data forma normale

Normalizzazione

Definizione

La **procedura** che permette di portare uno schema relazionale in una determinata forma normale si dice normalizzazione.

Osservazione

La normalizzazione può essere utilizzata come **tecnica di verifica** dei risultati della progettazione, non costituisce una metodologia di progetto.

Esempio (1/2)

R (<u>Impiegato</u>, Stipendio, <u>Progetto</u>, Bilancio, Funzione)

Considerazioni:

- Ogni impiegato può partecipare a più progetti, sempre con lo stesso stipendio, e con una sola funzione per progetto.
- Ogni progetto ha un bilancio indipendentemente da quanti dipendenti ci lavorano

Esempio (2/2)

Possibili anomalie:

- **di aggiornamento**
  - se lo stipendio di un impiegato varia $\rightarrow$ variano altre tuple (quali?)
  - se bilancio di progetto varia $\rightarrow$ variano altre tuple (quali?)
- **di cancellazione**
  - se un impiegato si licenzia, dobbiamo cancellarlo in diverse ennuple

Causa dei problemi

Abbiamo ripetizione dello stipendio di un impiegato e del bilancio di un progetto.

Errore di progetto

Usare un'unica relazione per rappresentare gruppi di informazione eterogenee.

Dipendenze funzionali

Ma quindi cosa sono le FD??

Siano:

- **r** relazione su $\mathbf{R}(\mathbf{X})$
- due sottoinsiemi non vuoti di $\mathbf{X} \supseteq \mathbf{Y}, \mathbf{Z}$

Definizione

Esiste in **r** una dipendenza funzionale da **Y** a **Z** se

$$\forall t_1,\ t_2 \text{ tuple in } \mathbf{r},\ t_1[\mathbf{Y}] = t_2[\mathbf{Y}] \implies t_1[\mathbf{Z}] = t_2[\mathbf{Z}]$$

$$\mathbf{Y} \rightarrow \mathbf{Z}$$

Nota bene!

$$\mathbf{Y} \to \mathbf{Z} \text{ non implica } \mathbf{Z} \to \mathbf{Y}!$$

FD complete

Una dipendenza funzionale si dice completa se

$$\mathbf{Y} \to \mathbf{Z}, \text{ e } \forall \mathbf{W} \subseteq \mathbf{Y}, \text{ non vale } \mathbf{W} \to \mathbf{Y}$$

- Se $\mathbf{Y}$ superchiave di $\mathbf{R}(\mathbf{X})$, allora $\mathbf{Y}$ determina ogni altro attributo della relazione $\implies \mathbf{Y} \to \mathbf{X}$
- Se $\mathbf{Y}$ chiave, allora $\mathbf{Y} \to \mathbf{X}$ è una DF completa

Esempio

R (<u>Impiegato</u>, Stipendio, <u>Progetto</u>, Bilancio, Funzione)

Caratterizziamo le dipendenze funzionali:

- Impiegato $\rightarrow$ Stipendio
- Progetto $\rightarrow$ Bilancio
- Impiegato Progetto $\rightarrow$ Funzione

Definizione

Una DF $\mathbf{Y} \rightarrow \mathbf{Z}$ si dice banale se $\forall$ Z_i attributo di $\mathbf{Z}$, vale $\mathbf{Y} \rightarrow Z_i$

- Impiegato Progetto $\rightarrow$ Progetto (DF banale, sempre soddisfatta.)

Legame fra DF e anomalie

Notare che:

- Impiegato $\rightarrow$ Stipendio, ci sono ripetizioni
- Progetto $\rightarrow$ Bilancio, ci sono ripetizioni
- Impiegato Progetto $\rightarrow$ Funzione, **non** ci sono ripetizioni

Le informazioni legate ad attributi non chiave causano problemi.

Implicazione

Siano F un insieme di dipendenze funzionali su $\mathbf{R}(\mathbf{Z})$ e $\mathbf{X} \to \mathbf{Y}$
Allora:

Definizione

Se $\forall$ $\mathbf{r}$ istanza di $\mathbf{R}$ che verifica tutte le dipendenze in F, risulta verificata anche $\mathbf{X} \to \mathbf{Y}$, allora si dice che F implica $\mathbf{X} \to \mathbf{Y}$.

$$\mathrm{F} \implies \mathbf{X} \to \mathbf{Y}$$

Chiusura

Siano F un insieme di dipendenze funzionali su $\mathbf{R}(\mathbf{Z})$ e $\mathbf{X} \to \mathbf{Y}$, allora la chiusura di F è l'insieme di tutte le dipendenze funzionali implicate da F:

Definizione

$$F^+ = \{\ \mathbf{X} \to \mathbf{Y} \mid F \implies \mathbf{X} \to \mathbf{Y}\ \}$$

Osservazione

Se un'istanza **r** dello schema **R** soddisfa F, allora soddisfa anche F^+.

Superchiave

Siano F un insieme di dipendenze funzionali su $\mathbf{R}(\mathbf{Z})$, e $\mathbf{K} \subseteq \mathbf{Z}$. Allora:

Definizione

$\mathbf{K}$ si dice superchiave di R se la dipendenza funzionale $\mathbf{K} \to \mathbf{Z}$ è logicamente implicata da F, ovvero se $\mathbf{K} \to \mathbf{Z} \in F^+$.

Ricordiamo che se nessun insieme proprio di **K** è superchiave di **R**, allora **K** si dice chiave di **R**. Gli attributi di una chiave non si possono ottenere da nessuna DF, si deve partire da loro per ottenere gli altri.

Costruzione della chiusura

Problema

Trovare tutte le chiavi di una relazione — al peggio esponenziale.
Rinuncia alla "chiusura totale"

Algoritmo:

- attributi SOLO a sx delle DF stanno in tutte le chiavi
- chiamo l'insieme di quei attributi N
- calcola N^+
- aggiungi ad N un attributo alla volta, poi una coppia, etc

Calcolo di F^+

La definizione di implicazione prevede un quantificatore universale.

- non utilizzabile in pratica
- servono regole di inferenza per derivare costruttivamente le DF

1974, Armstrong "Dependency structures of data base relationships"
(Spiegati bene anche su Wikipedia)

Regole di inferenza di Armstrong

1. **Riflessività**: se $\mathbf{Y} \subseteq \mathbf{X}$, allora $\mathbf{X} \rightarrow \mathbf{Y}$
2. **Additività/Arricchimento**: se $\mathbf{X} \rightarrow \mathbf{Y}$, allora $\mathbf{XZ} \rightarrow \mathbf{YZ}\ \forall \mathbf{Z}$
3. **Transitività**: se $\mathbf{X} \rightarrow \mathbf{Y}$ e $\mathbf{Y} \rightarrow \mathbf{Z}$, allora $\mathbf{X} \rightarrow \mathbf{Z}$

Proprietà

Teorema correttezza

Applicandole ad un insieme F di dipendenze funzionali, si ottengono solo dipendenze logicamente implicate da F.

Teorema completezza

Applicandole ad un insieme F di dipendenze funzionali, si ottengono tutte le dipendenze logicamente implicate da F.

Teorema minimalità

Ignorando anche una sola regola, l'insieme di regole che rimangono non è più completo.

Dimmostrazione addittività: per assurdo

Supponiamo che $\exists$ **r** istanza di **R** | vale $\mathbf{X} \rightarrow \mathbf{Y}$, ma non $\mathbf{XZ} \rightarrow \mathbf{YZ}$
Allora $\exists\ t_1, t_2$ di **r** tali che:

(1) $t_1[\mathbf{X}] = t_2[\mathbf{X}]$

(2) $t_1[\mathbf{Y}] = t_2[\mathbf{Y}]$

(3) $t_1[\mathbf{XZ}] = t_2[\mathbf{XZ}]$

(4) $t_1[\mathbf{YZ}] \neq t_2[\mathbf{YZ}]$

Da (1) e (3) si deduce: (5) $t_1[\mathbf{Z}] = t_2[\mathbf{Z}]$
Mentre da (2) e (5) abbiamo: (6) $t_1[\mathbf{YZ}] = t_2[\mathbf{YZ}]$,
in contradizione con (4), assurdo.

$\blacksquare$

Dimmostrazione transitività: sempre per assurdo

Fai finta che $\exists$ **r** istanza di **R** $|$ valgono $\mathbf{X} \to \mathbf{Y}$ e $\mathbf{Y} \to \mathbf{Z}$, ma non $\mathbf{X} \to \mathbf{Z}$
Allora $\exists\ t_1, t_2$ di **r** tali che:

(1) $t_1[\mathbf{X}] = t_2[\mathbf{X}]$

(2) $t_1[\mathbf{Y}] = t_2[\mathbf{Y}]$

(3) $t_1[\mathbf{Z}] = t_2[\mathbf{Z}]$, ma anche

(4) $t_1[\mathbf{Z}] \neq t_2[\mathbf{Z}]$

in contradizione con (3), assurdo.

$\blacksquare$

Dimmostrazione riflessività: fatta io, aperta a critiche

Supponiamo che $\exists$ $\mathbf{Y} \subseteq \mathbf{X}$ | non vale $\mathbf{X} \rightarrow \mathbf{Y}$.
Comunque deve valere $\mathbf{X} \rightarrow \mathbf{X}$, DF banale. Allora $\exists$ t_1, t_2 di $\mathbf{r}$ tali che:

(1) $t_1[\mathbf{X}] = t_2[\mathbf{X}]$.
Ma per ipotesi $\mathbf{Y} \subseteq \mathbf{X}$, quindi per forza di cose vale:

(2) $t_1[\mathbf{X}] = t_2[\mathbf{Y}]$,

in contradizione con l'ipotesi che non vale $\mathbf{X} \rightarrow \mathbf{Y}$, assurdo.

$\blacksquare$

Algoritmo per trovare F^+

```
F+ = F
ripeti
    per ogni DF f in F+
        aggiungi riflessivita(f) ad F+
        aggiungi arricchimento(f) ad F+
    fine
    per ogni DF f1, f2 in F+
        // f1 = X->Y, f2 = Z->W
        se Y == Z
            aggiungi transitivita(f1, f2) ad F+
    fine
fine
```

- complessità esponenziale: il per ogni nasconde il problema di trovare tutti i sottoinsiemi (backtracking), che va come 2^n.

Esempio

Sia lo schema **R** (A, B, C, G, H, I), con le dipendenze funzionali $F = \{ A \to B, A \to C, CG \to H, CG \to I, B \to H \}$ Alcuni membri di F^+ sono:

- $A \to H$, ottenuto da Transitività($A \to B$, $B \to H$);
- $AG \to I$, ottenuto da Arricchimento($A \to C$, G) = $AG \to CG$
  e Transitività($AG \to CG$, $CG \to I$);
- $CG \to HI$, ottenuto da Arricchimento($CG \to I$, CG) = $CG \to CGI$
  e Arricchimento($CG \to H$, I) = $CGI \to HI$
  e Transitività($CG \to CGI$, $CGI \to HI$);

Regole derivate di Armstrong

1. **Unione**: $\{\mathbf{X} \to \mathbf{Y}, \mathbf{X} \to \mathbf{Z}\} \implies \mathbf{X} \to \mathbf{YZ}$
2. **Pseudotransitività**: $\{\mathbf{X} \to \mathbf{Y}, \mathbf{WY} \to \mathbf{Z}\} \implies \mathbf{WX} \to \mathbf{Z}$
3. **Decomposizione**: se $\mathbf{Z} \subseteq \mathbf{Y}$, $\{\mathbf{X} \to \mathbf{Y}\} \implies \mathbf{X} \to \mathbf{Z}$

Dimostrazione della Regola dell'Unione

Le regole derivate si dimostrando a partire dalle regole di braccio forte. Infatti, per ipotesi valgono:

(1) $\mathbf{X} \rightarrow \mathbf{Y}$, che per addittività diventa (3) $\mathbf{XZ} \rightarrow \mathbf{YZ}$

(2) $\mathbf{Y} \rightarrow \mathbf{Z}$, che per addittività diventa (4) $\mathbf{XX} \rightarrow \mathbf{XZ} = \mathbf{X} \rightarrow \mathbf{XZ}$

Applicando la transitività a (4) e (3), segue la tesi.

$\blacksquare$

Esempio (LG2)

R (Impiegato, Stipendio, Progetto, Bilancio, Funzione)

- Ogni impiegato può partecipare a più progetti, sempre con lo stesso stipendio, e con una sola funzione per progetto.
- Ogni progetto ha un bilancio indipendentemente da quanti dipendenti ci lavorano

---

(a) Impiegato → Stipendio + arricchimento
= Impiegato Proggetto → Stipendio Proggetto (d)

(b) Progetto → Bilancio + arricchimento
= Proggetto Impiegato → Stipendio Impiegato (e)

(c) Impiegato Progetto → Funzione

Applicandocenter la Regola dell'Unione alle {(c), (d), (e)} si trova:

Impiegato Proggetto → Stipendio Proggetto Impiegato Funzione,

e quindi Impiegato, Proggetto è chiave.

Equivalenza fra insiemi di DF

Problema

Dati F, G insiemi di DF, torna utile sapere se sono equivalenti. Ovvero:

Definizione

F, G si dicono equivalenti se $F^+ = G^+$, quindi

- $\forall$ $\mathbf{X} \rightarrow \mathbf{Y} \in F$ risulta $\mathbf{X} \rightarrow \mathbf{Y} \in G^+$, e viceversa
- $\forall$ $\mathbf{X} \rightarrow \mathbf{Y} \in G$ risulta $\mathbf{X} \rightarrow \mathbf{Y} \in F^\cdot$

- In pratica, per verificare l'equivalenza, prendi tutte le DF in F, e prova a dimostrarle usando le DF in G, e viceversa. Se tutto va bene, congratulazioni, F equivalente a G!

Chiusura transitiva di un insieme di attributi

Il calcolo di F^+ è molto costoso, ma tutte le DF banali manco servono! Spesso ci interessa se F^+ contiene qualche specifica dipendenza. Che fare?

Definizione

La chiusura di un insieme di attributi A è l'insieme di tutti gli attributi che dipendono da A, relativamente a un dato F.

Boom!

Teorema

$$\mathbf{X} \to \mathbf{Y} \in F^+ \iff \mathbf{Y} \subseteq \mathbf{X}^+$$

Quindi è sufficiente fare la chiusura di **X** per capire se la dipendenza vale

Algoritmo per trovare $\mathbf{X}^+$

```
InsiemeAttributi X+ = X
InsiemeAttributi oldX+ = NULL
InsiemeDF F

finche (X+ != oldX+)
ripeti
    oldX+ = X+
    per ogni DF Vi -> Wi in F
        se (Vi incluso in X+) e (Wi non incluso in X+)
            X+ = X+ reunito con Wi
        fine
    fine
fine
```

Esempio: calcola se A è superchiave

Sia $F = \{A \to B, BC \to D, B \to E, E \to C\}$. Calcoliamo A^+

- $A^+ = A$
- $A^+ = AB$ poiché $A \to B$ e $A \subseteq A^+$
- $A^+ = ABE$ poiché $B \to E$ e $B \subseteq A^+$
- $A^+ = ABEC$ poiché $E \to C$ e $E \subseteq A^+$
- $A^+ = ABCED$ poiché $BC \to D$ e $BC \subseteq A^+$

Quindi da A dipendono tutti gli altri attributi, ovvero A è superchiave (e anche chiave)!

Esempio: calcola se F,G equivalenti

Siano: $F = \{A \to C,\ AC \to D,\ E \to AD,\ E \to H\}$,
$G = \{A \to CD,\ E \to AH\}$

Invece di verificare se $\forall\ \mathbf{X} \to \mathbf{Y} \in F$ anche in G^+, vedo se $\mathbf{Y} \subseteq (\mathbf{X})^{+G}$ (notazione fancy per chiusura di $\mathbf{X}$ rispetto a G) e viceversa.

- per $A \to C$ si ha $(A)^{+G} = ACD$, $C \subseteq (A)^{+G}$ ✓
- per $AC \to D$ si ha $(AC)^{+G} = ACD$, $D \subseteq (AC)^{+G}$ ✓
- per $E \to AD$ si ha $(E)^{+G} = EACDH$, $AD \subseteq (E)^{+G}$ ✓
- per $E \to H$ si ha $H \subseteq (AC)^{+G}$ ✓

The viceversa is left as an exercise for the reader.

QUINDI...

Equivalenza con chiusura di attributi

F, G equivalenti sse...

- $\forall\, \mathbf{X} \rightarrow \mathbf{Y} \in F,\ \mathbf{Y} \in (\mathbf{X})^{+G}$
- $\forall\, \mathbf{X} \rightarrow \mathbf{Y} \in F,\ \mathbf{Y} \in (\mathbf{X})^{+G}$

Conclusioni sulla chiusura di attributi

Sia $\mathbf{R}(\mathbf{Z})$ con le sue dipendenze in F.
Allora, la chiusura di $\mathbf{X} \subseteq \mathbf{Z}$ utile per verificare se:

- una dipendenza funzionale è logicamente implicata da F (vedi Teorema)
- un insieme di attributi è superchiave o chiave
  - $\mathbf{X}$ è superchiave di $\mathbf{R}$ sse $\mathbf{X} \to \mathbf{Z} \in F^+$, ovvero sse $\mathbf{Z} \subseteq (\mathbf{X})^+$
  - $\mathbf{X}$ è chiave di $\mathbf{R}$ sse $\mathbf{X} \to \mathbf{Z} \in F^+$ e $\nexists$ $\mathbf{Y}$ $\subset$ $\mathbf{X}$ tale che $\mathbf{Z} \subseteq (\mathbf{Y})^+$

Ridondanza di un insieme di DF

Overview

Si vuole usare il concetto di equivalenza tra insiemi di DF per partire da una F più semplice possibile.

- a destra: $\{A \to BC\}$ equivalente a $\{A \to B, A \to C\}$ "FD semplici"
- a sinistra: $\{A \to B, AB \to C\}$ equivalente a $\{A \to B, A \to C\}$ "senza attributi estranei"

- possono esserci DF ridondanti, aka ottenibili da altre DF: $A \to C$ ridondante $\{A \to B, B \to C, A \to C\}$

La riduzione della complessità può riguardare il numero di attributi che si usano in una dipendenza, o il numero di dipendenze nell'insieme.

FD semplici

Possiamo portare un insieme di FD F in forma "standard", quella in cui sulla destra c'è un singolo attributo.
Se $F = \{AB \rightarrow CD, AC \rightarrow DE\}$ lo si scompone in:

$$\{AB \rightarrow C, AB \rightarrow D, AC \rightarrow D, AC \rightarrow E\}$$

Attributi estranei

Definizione

Gli attributi a sinistra di una FD che sono inutili entro un dato F si dicono estranei.

Sia $F = \{AB \rightarrow C, A \rightarrow B\}$, e calcoliamo A^+, B^+:

- $A^+ = A$
- coming soon in a cinema near you

FD ridondanti

coming soon in a cinema near you

Copertura minimale: definizione

coming soon in a cinema near you

Copertura minimale: algoritmo

coming soon in a cinema near you