Università degli Studi di Salerno

Dipartimento di Informatica

Tesi di Laurea Triennale in

Informatica

Abstract

UN ALGORITMO PER L'INFERENZA DI DIPENDENZE FUNZIONALI RILASSATE: VERIFICA DI AMMISSIBILITA' DI PATTERN

Relatore

Chiar.mo Prof. Vincenzo Deufemia

Secondo Relatore

Dott.sa Loredana Caruccio

Candidato

Luigi Durso

Matr. 0512101919

Anno Accademico 2016-2017

Abstract

Nella progettazione di una base di dati ci sono aspetti essenziali da prendere in considerazione per assicurare un servizio quanto più efficiente possibile. Considerato il netto aumento del flusso di dati degli ultimi anni, la *data quality* è divenuta una materia estremamente interessante vista la cospicua presenza di dati "sporchi" nelle basi di dati. Per ridurre anomalie ed inconsistenze ci vengono incontro le *Dipendenze funzionali*, utilizzate ampiamente per stabilire vincoli di integrità tra i dati. La grande mole di dati, però, ha reso necessario un riadattamento delle dipendenze funzionali rendendole in grado di catturare inconsistenze più ampie nei dati. Le *Dipendenze funzionali rilassate o approssimate* **(RFD)** sono da considerarsi come una naturale evoluzione o generalizzazione delle *dipendenze funzionali canoniche*. Infatti, il concetto più importante introdotto dalle RFD è quello della *similarità*. Mentre nelle dipendenze funzionali classiche esiste soltanto il vincolo di uguaglianza tra dati, nelle RFD questo vincolo viene esteso introducendo il confronto approssimato tramite funzioni di similarità, questo ci permetterà di coprire una quantità di dati maggiore. Tuttavia le RFD possono fornire vantaggi solo se possono essere scoperte automaticamente dai dati. Il lavoro di tesi sifocalizza su quest'ultimo aspetto di recuperare le RFD attraverso una procedura automa-

tizzata. L'algoritmo proposto è in grado di scoprire le RFD presenti in un dataset fornito come input attraverso tre differenti fasi. La prima fase, chiamata *Feasibility*, si occupa di estrarre un insiemi di tuple dal set rilevanti per l'identificazione delle RFD. La seconda fase, chiamata *Minimality*, si occupa di minimizzare il numero di pattern ottenuti dalla prima fase. Infine, l'ultima fase, chiamata *Generation*, estrae le RFD attraverso la verifica di opportune regole. Per questo lavoro di tesi mostreremo l'idea dell'algoritmo generale ed entreremo nel dettaglio della prima fase di sviluppo(Feasibility), mostrando, infine, i risultati della sperimentazione.