UNIVERSITÀ DEGLI STUDI DI BARI ALDO MORO

DIPARTIMENTO DI INFORMATICA

CORSO DI LAUREA IN
INFORMATICA

Progettazione di algoritmi per la risoluzione di problemi di deduplica dei dati strutturati

Relatori:
Chiar.mo prof. **Michele SCALERA**
Dott. **Giovanni Bruno**

Laureando: **Filippo Antonio SARCINELLI**

SERLAB
SOFTWARE ENGINEERING RESEARCH LABORATORY
c/o DIPARTIMENTO DI INFORMATICA
via Orabona, 4 - 70126 - Bari
Tel.: + 39.080.5443279 Fax: + 39.080.5442536

INDICE

- Data Quality
- Obiettivo
- Ricerca
- Esperimento
- Sperimentazione
- Risultati
- Sviluppi futuri

Data Quality

Con il termine *Data Quality* si fa riferimento a un insieme di caratteristiche che il dato dovrebbe possedere come:

- ⇨ Accuratezza
- ⇨ Correttezza
- ⇨ Aggiornamento
- ⇨ Ecc.

La qualità dei dati aiuta a prendere decisioni aziendali incisive per il raggiungimento degli obiettivi.

2 standard ISO per la qualità dei dati:

- ISO/IEC 25012:
  - Definisce un modello di qualità in cui vengono stabilite quali sono le caratteristiche che devono essere prese in considerazione, quando vengono analizzati i dati.
- ISO/IEC 25024:
  - Definisce le misure di qualità per quantificare la qualità dei dati in termini di caratteristiche definite nella ISO 25012.

Obiettivo

- Sviluppare un sistema in grado di rilevare tuple duplicate in un database

- Il sistema deve permettere:
  - Effettuare confronti tra singoli attributi;
  - Filtrare l'intera lista di tuple;
  - Effettuare confronti tra le tuple filtrate;
  - Consentire l'analisi di dettaglio tramite generazione di report;
  - Permettere la gestione delle eccezioni.

Ricerca

Una tecnica utilizzata per trovare duplicati è la suddivisione in blocchi.

- L'obiettivo degli algoritmi di blocco è filtrare l'intero set di dati, da confrontare tra loro, in blocchi più piccoli contenenti tuple con alcune caratteristiche in comune.

Dall'analisi della letteratura esistente, è stato possibile riscontrare quattro diversi algoritmi di blocco:

- Standard Blocking;
- Sorted Neighborhood;
- Bigram Indexing;
- Canopy Clustering.

Algoritmi di blocco

- Hanno lo scopo di alleggerire il numero di confronti da effettuare.
  - Filtrano l'intero set di dati selezionando solo tuple potenzialmente simili e ignorando le restanti, considerate dissimili.

- Due funzioni pricipali:
  - Corrispondenza:

    Ha il compito di identificare record duplicati o che fanno riferimento alla stessa entità.
  - Unione:

    Ha il compito di accorpare le tuple simili riscontrate dalla prima.

La creazione di blocchi, detta fase di indicizzazione, richiede l'ausilio di uno o più attributi di blocco, detti chiave di blocco.

⇨ Chiave di blocco

- È una stringa formata:
  - Da un singolo attributo di blocco;
  - Dalla congiunzione tra i vari attributi di blocco;
  - Da una sottostringa dell'attributo di blocco.

Sequenza *blocking methods:*

- ⇨ Fase di indicizzazione:
  - Viene creata e associata una chiave di blocco per ogni tupla presente nel database;
  - Se due o più tuple condividono la stessa *blocking key* vengono raggruppate, formando un blocco.
- ⇨ Confronto tra le tuple di ogni blocco;
- ⇨ Classificazione dei risultati in:
  - Corrispondenti;
  - Probabili corrispondenze:
    - Le tuple richiedono successiva revisione umana per essere classificate come corrispondenti o meno;
  - Non corrispondenti.

Standard Blocking

- Creazione chiave di blocco;
- Indicizzazione database;
- Confronto tra tutte le tuple contenute in ogni singolo blocco;
- Classificazione.

Sorted Neighborhood

- Creazione chiave di blocco;
- Indicizzazione database;
  - Viene creata una finestra di dimensione limitata per evitare blocchi troppo grandi;
  - Ogni qualvolta viene aggiunta una tupla, viene confrontata con le precedenti;
  - Se la finestra è piena viene eliminato il primo elemento e l'ultimo viene accodato. FIFO (First In First Out).
- Classificazione.

Bigram Indexing

- Determinare la soglia, intervallo [0,1];
- Creazione chiave di blocco;
- Indicizzazione database;
  - Creazione iniziale dei blocchi.
- Suddivisione in bigram della chiave di blocco;
  - Creata una sotto-lista ottenuta da ogni coppia di lettere.

- Vengono determinati dei sinonimi della chiave di blocco ottenuti:
  - Lunghezza sinonimi = numero di bigram ottenuti x soglia (ls = b x s);
  - Create tutte le possibili permutazioni ottenute utilizzando ls bigram, considerate come sinonimi della chiave di blocco;
- I blocchi vengono arricchiti con le successive ricerche effettuate con i sinonimi.
- Confronto tuple di ogni blocco;
- Classificazione.

Bigram esempio…

- Chiave di blocco = baxter;
- Soglia = 0,8;
- Sottolista bigram = (‘ba’, ‘ax’, ‘xt’, ‘te’, ‘er’);
- ls = 5 x 0,8 = 4;

(‘ax’, ‘xt’, ‘te’, ‘er’) -> axxtteer
(‘ba’, ‘xt’, ‘te’, ‘er’) -> baxtteer
(‘ba’, ‘ax’, ‘te’, ‘er’) -> baaxteer
(‘ba’, ‘ax’, ‘xt’, ‘er’) -> baaxxter
(‘ba’, ‘ax’, ‘xt’, ‘te’) -> baaxxtte

Canopy Clustering

- Creazione chiave di blocco;
- Indicizzazione database;
- Confronto tra le tuple contenute in ogni singolo blocco.
  - Confronto viene effettuato tramite la metrica TF – IDF;
  - Risultato tra 0 = dissimili e 1 = perfettamente uguali.
- Classificazione.

Esperimento

- Il sistema, sviluppato interamente in java, permette l'analisi completa dei duplicati di un singolo database.

- Permette sia di visualizzare i risultai ottenuti a video, sia di consultare i risultati tramite report.

Sperimentazione

Si è optato per una sperimentazione in vitro.

- ⇨ È stato realizzato un database di anagrafica di 100 tuple;
- ⇨ Contenente in totale 55 duplicati di qualunque tipo;

In ogni fase sperimentale, sono state analizzate tutte le coppie di tuple associate allo stesso blocco, con l'obiettivo di individuare tutti quei record aventi attributi scritti con diverse formattazioni.

Risultati

	SQL	Standard Blocking	Sorted Neighborhood			Bigram Indexing			Canopy Clustering
	Totale	Totale	Finestra = 3	Finestra = 5	Finestra = 7	Soglia = 0,3	Soglia = 0,5	Soglia = 0,7	Totale
			Totale	Totale	Totale	Totale	Totale	Totale	
Nome	63	98	65	86	92	98	99	99	98
Cognome	69	140	71	98	114	0	0	0	140
Data di nascita	79	187	92	127	149	187	187	187	187
Sesso	99	2707	188	368	540	0	0	0	2707
Nome e cognome	36	41	30	38	41	41	41	41	41
Nome, cognome e data di nascita	32	39	28	36	39	39	39	39	39

Risultati

I risultati ottenuti garantiscono la superiorità di Deduplication Tool rispetto alla semplice query SQL.

Nelle varie iterazioni eseguite, il Tool ha dimostrato di essere in grado di ritrovare da un minimo di 19 a un massimo di 47 duplicati su 55. Quindi con una precisione che va dal 34,5% nel caso peggiore all'85,4% di duplicati ritrovati nel migliore dei casi.

Sviluppi futuri

- Il progetto sviluppato rappresenta il secondo step di una linea di ricerca del laboratorio SERLAB.
- Esso si presta ad innumerevoli evoluzioni e all'aggiunta di nuovi servizi. Come dei controlli basati sulla correttezza delle parole tramite algoritmi di string-matching.