Datenbank Spektrum
DOI 10.1007/s13222-011-0043-9

KURZ ERKLÄRT

Kurz erklärt: Datenfusion

Jens Bleiholder · Felix Naumann

1 Heterogenität und Datenreinigung

Moderne Werkzeuge zur Informationsintegration ermöglichen es auf zunehmend einfache Weise, die Daten weltweit verteilter, heterogener Datenquellen zusammenzuführen. Dazu zählt die Überwindung technischer Heterogenität (Formate, Protokolle, ...), die Überwindung struktureller Heterogenität (Datenmodell, Schema, ...) sowie die Überwindung der semantischen Heterogenität, die sich also auf den konkreten Inhalt der Quellen bezieht. Im letzteren, oft vernachlässigten Gebiet ist die Datenfusion angesiedelt.

Datenfusion bezeichnet die Zusammenführung mehrerer verschiedener Datensätze, die alle dasselbe Objekt in der realen Welt beschreiben. Ein typisches Beispiel ist ein mehrfach erfasster Kunde, der mit möglicherweise leicht unterschiedlichen Daten in den zu integrierenden Datenbanken vertreten ist. In einem ersten Schritt werden Struktur und Schema der einzelnen Datenbanken in Einklang gebracht, so dass alle Datensätze dem gleichen Schema unterliegen und vergleichbar sind (*schema mapping*). In einem zweiten Schritt werden die so genannten Dubletten oder Duplikate mittels spezialisierter Duplikaterkennungsverfahren (*duplicate detection*) entdeckt [6]. Schließlich werden die entdeckten Duplikate mittels der Datenfusion kombiniert. Abbildung 1 stellt diesen Prozess dar. Insbesondere die letzten beiden Schritte werden oft als Datenreinigung (*data cleansing*) bezeichnet. Das Ergebnis der Integration und Reinigung ist idealerweise ein einziger, konsistenter und sauberer Datensatz für jede relevante Entität der Domäne. Datenfusion ist somit für die Aufgaben der Stammdatenverwaltung (*master data management*) besonders wichtig: Referenzdaten entstehen oft als Ergebnis der Fusion mehrerer Datensätze.

Im Beispiel soll für jeden realen Kunden höchstens ein Datensatz verbleiben. Dieser Datensatz kann durchaus aus Daten verschiedener Ursprungsdatensätze bestehen – also eine Kombination der Datensätze sein. In der Praxis werden auch die in den resultierenden Datensatz einfließenden Daten aufbewahrt, um Datenherkunft (*provenance/lineage*) nachweisen zu können und Konsistenz zu älteren Transaktionen zu gewährleisten.

Zusammenfassend hat die Datenfusion also die Aufgabe, in einem Datenbestand mit einer möglicherweise sehr großen Anzahl an Duplikaten sämtliche Konflikte innerhalb der Duplikate zu lösen. Dabei können Konflikte verschiedenen Typs sein und mittels verschiedener Verfahren gelöst werden, wie in den folgenden Abschnitten ausgeführt wird.

2 Datenkonflikte

Ein Datenkonflikt besteht zwischen zwei Datensätzen, falls sie ein Duplikat sind und sie für eine gleiche Eigenschaft (Attribut) unterschiedliche Werte speichern. Mindestens einer der in Konflikt stehenden Werte ist also falsch. Falsch erfasste Werte sind eine typische Ursache für Duplikate. Wird eine Entität ein zweites Mal erfasst, dieses Mal mit richtigen Werten oder mit anderen Fehlern, geht das Informationssystem von einer zweiten Entität aus. Die Ursachen für Datenkonflikte und somit auch für Duplikate sind mannigfaltig: Tippfehler und Zahlendreher, veraltete oder unbekannte Werte, fehlerhaft oder ungenau gemessene Werte, usw.

J. Bleiholder
Opitz Consulting Berlin GmbH, Berlin, Deutschland
e-mail: jens.bleiholder@opitz-consulting.com

F. Naumann (✉)
Hasso Plattner Institut, Potsdam, Deutschland
e-mail: naumann@hpi.uni-potsdam.de

Published online: 10 February 2011

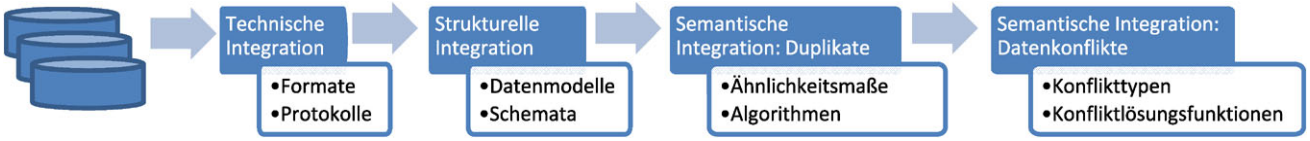

Abb. 1 Datenintegration in drei Schritten

Rahm und Do führen eine gelungene Klassifikation der Fehlerursachen ein [8].

Im folgenden Beispiel kann mittels der ISBN das Duplikat erkannt werden. Alle anderen Attribute jedoch konfligieren.

ISBN	Titel	Autor	Jahr	Preis
3423124008	Faust	Goethe	1997	6,99
3423124008	Fuast	von Goethe	⊥	7,50

Im Titel-Attribut befindet sich ein Tippfehler, so dass die Buchtitel unterschiedlich erscheinen. Es ist zunächst nicht leicht, einen solchen Fehler automatisiert zu korrigieren. Das Autor-Feld des zweiten Datensatzes enthält vollständigere Information – eine gängige Konfliktlösung würde diesen längeren Wert übernehmen. Das Attribut Jahr desselben Datensatzes hingegen enthält einen Nullwert. Es bietet sich hier an den vorhandenen Wert des ersten Datensatzes zu übernehmen. In einem solchen Fall spricht man oft auch von *Unsicherheit* und nicht von einem *Konflikt* oder *Widerspruch*. Und schließlich unterscheiden sich die Daten im Preis-Attribut. Es liegt in diesem Fall nahe, den kleineren Preis zu wählen. An diesem Beispiel erkennt man die Wichtigkeit der Herkunftsinformationen – ohne sie ginge die Information verloren, bei welchem Händler der niedrigere Preis zu erzielen ist.

3 Konfliktlösung

Konflikte können mittels verschiedener Konfliktlösungsstrategien gelöst werden. Eine wurde bereits im letzten Kapitel genannt: die Strategie TAKE THE INFORMATION kann auf das Attribut Jahr angewandt werden und dient insbesondere der Lösung von Unsicherheiten. Dabei wird die vorhandene Information dem Nullwert vorgezogen. Eine weitere gängige Strategie ist die TRUST YOUR FRIENDS Strategie: Es wird eine Reihenfolge der Quellen nach Vertrauenswürdigkeit angelegt und der Wert aus der vertrauenswürdigeren Quelle genommen (bspw. Amazon.de vor libri.de). Stehen mehr als zwei Quellen oder Attributwerte zur Auswahl, empfiehlt sich auch die CRY WITH THE WOLVES-Strategie: Hierbei wird der am häufigsten auftretende Wert gewählt und so ein Mehrheitsentscheid herbeigeführt. Natürlich sind nicht alle Strategien perfekt und die Möglichkeit eines Fehlers ist insbesondere bei automatischer Konfliktlösung, z. B. mittels eines solchen Mehrheitsentscheides nicht auszuschliessen. Eine sorgfältige Auswahl und auf den jeweiligen Anwendungszweck abgestimmte Strategie und Anwendung von Operatoren und Techniken sind daher unerlässlich. So kann z. B. eine Abwandlung des Mehrheitsentscheides verwendet werden, um das Faust–Fuast-Problem zu lösen, oder generell Vor- und Nachnamen zu unterscheiden und so z. B. auch komplizierte Fälle zu korrigieren, in denen Datenkonflikte auch aufgrund vertauschter Vor- und Nachnamensfelder auftreten (siehe folgendes Beispiel).

ISBN	Titel	Autor	Jahr	Preis
3423124008	Faust	Goethe	1997	6,99
3423124008	von Goethe	Fuast	⊥	6,99

Mit Hilfe eines Referenzdatenbestandes von Vor- bzw. Nachnamen (Titel und Autorennamen im Beispiel) kann so z. B. bestimmt werden, welcher Attributwert mit höherer Wahrscheinlichkeit als Vor- bzw. Nachname auftritt und der Konflikt entsprechend gelöst werden (Titel und Autor im Beispiel).

Viele Konfliktlösungsstrategien und ihre Umsetzung mit Hilfe relationaler Techniken und in Integrationssystemen sind in [1] beschrieben. Im Folgenden wird eine Auswahl kurz vorgestellt.

3.1 Relationale Operatoren

Operatoren der relationalen Algebra erlauben es, bestimmte Konfliktlösungsstrategien für relationale Daten anzuwenden. Jedoch haben alle Standardoperatoren (*join*, *union*, etc.) ihre jeweils eigenen Besonderheiten und erzeugen nicht in allen Fällen eindeutige, konfliktfreie Datensätze. Auch die Frage, ob Datenkonflikte nur innerhalb einer Tabelle oder zwischen Tabellen auftreten können und gelöst werden müssen, beeinflusst die Operatorwahl. Mehr Erfolg versprechen komplett neue Operatoren oder die Kombinationen von Standardoperatoren. So lösen z. B. die Operatoren *minimum union* und *complement union* viele Fälle von Unsicherheiten, sowohl bei der Konfliktlösung innerhalb einer Tabelle als auch Tabellen-übergreifend [2]. *Full disjunction* ist eine Erweiterung des *outer join*, der auch einige Fälle von Unsicherheiten abdeckt; auch der *merge*-Operator löst Unsicherheiten auf. Alle diese Operatoren, im Überblick in [1], sind

in Fällen sehr erfolgreich, in denen nur Unsicherheiten, aber keine Widersprüche auftreten, wie im folgenden Beispiel.

ISBN	Titel	Autor	Jahr	Preis
3423124008	Faust	⊥	1997	6,99
3423124008	⊥	von Goethe	⊥	6,99

Zur Lösung echter Widersprüche bieten sich Techniken an, die auf Gruppierung (aller Tupel anhand einer eindeutigen Kennung, im Beispiel die ISBN) und Aggregation beruhen, z. B. in Form des *fuse by*-Operators. Unterschiedliche Konfliktlösungstrategien werden durch eine geschickte Auswahl an Konfliktlösungsfunktionen implementiert, die innerhalb einer Duplikatgruppe (gleiche ID) auf einzelne Attribute angewendet werden. So kann die im vorigen Abschnitt beschriebene Konfliktlösung (Goethe vs. von Goethe), den längsten Autorennamen zu übernehmen, durch Gruppierung nach ISBN und Aggregation des Autorenfeldes mittels der LONGEST-Funktion realisiert werden.

3.2 Fusionssysteme

Viele einzelne Konfliktlösungsstrategien, -techniken und relationale Operatoren zur Konfliktlösung wurden in der Vergangenheit oft als Teil eines Integrationssystems eingesetzt. Die Geschichte der Integrationssysteme ist dabei fast so alt wie die Geschichte der (relationalen) Datenbanksysteme selbst und die Anzahl der verschiedenen Systeme entsprechend groß. Mit Beginn der 80er Jahre tauchten die ersten Systeme auf, die Daten aus mehreren Quellen integrieren und sich vor allem mit Techniken der technischen und strukturellen Integration befassen. Systeme, die sich um semantische Integration bemühen (sogenannte Fusionssysteme), sind dagegen seltener zu finden. Doch bereits MULTIBASE [3] stellt als eines der ersten Integrationssysteme das Problem der Datenkonflikte in Attributen dar und schlägt erste Lösungsansätze basierend auf Gruppierung und Aggregation vor. Als weiterer bekannter Vertreter verwendet TSIMMIS [7] als erstes System die TRUST YOUR FRIENDS-Strategie der Quellenauswahl. In jüngerer Vergangenheit wurde mit dem CONQUER-System [5] eines von vielen Systemen zu einer weiteren Konfliktlösungsstrategie vorgestellt: CONQUER (abkürzend für *consistent query answering*) gibt mittels eines SQL-Rewriting nur konsistente (widerspruchsfreie) Ergebnisse zurück und verwirft die restlichen.

4 Ausblick

Datenfusion beschäftigt sich mit den unterschiedlichen Möglichkeiten, die bei der Integration von Daten auftretenden Datenkonflikte zu beheben und somit die semantische Heterogenität zu überwinden. Es existieren viele unterschiedliche Verfahren, die überwiegende Mehrheit stammt aus dem Umfeld relationaler Operatoren. Mit deren Hilfe können die oftmals vorhandenen Unsicherheiten (in Form von Nullwerten) in den Daten mittlerweile recht zuverlässig behandelt werden. Die Behandlung echter Widersprüche hingegen erfordert eine sorgfältige Kenntnis der unterschiedlichen Eigenschaften und eine auf den Einsatzzweck sorgfältig abgestimmte Auswahl der verschiedenen Operatoren.

Die Einbeziehung relationaler Fusionsoperatoren in die Anfrageoptimierung steckt allerdings noch genauso in den Anfängen wie die (Teil-)Automatisierung von Datenfusionsprozessen oder die (teil-)automatische Ableitung von Fusionsregeln. Beide Bereiche sind Gegenstand aktiver Forschung, z. B. in [4].

Danksagung Die hier vorgestellte Arbeit wurde zum Teil durch die Deutsche Forschungsgemeinschaft unterstützt (DFG NA 432).

Literatur

1. Bleiholder J, Naumann F (2008) Data fusion. ACM Computing Surveys 41(1)
2. Bleiholder J, Szott S, Herschel M, Kaufer F, Naumann F (2010) Subsumption and complementation as data fusion operators. In: Proc of EDBT, pp 513–524
3. Dayal U (1983) Processing queries over generalization hierarchies in a multidatabase system. In: Proc of VLDB. Morgan Kaufmann, Florence, pp 342–353
4. Dong XL, Berti-Equille L, Srivastava D (2009) Truth discovery and copying detection in a dynamic world. PVLDB 2(1):562–573
5. Fuxman A, Fazli E, Miller RJ (2005) Conquer: efficient management of inconsistent databases. In: Proc of SIGMOD. ACM Press, New York, pp 155–166
6. Naumann F, Herschel M (2010) An Introduction to Duplicate Detection. Morgan Kaufmann, San Mateo
7. Papakonstantinou Y, Abiteboul S, Garcia-Molina H (1996) Object fusion in mediator systems. In: Proc of VLDB. Morgan Kaufmann, San Mateo, pp 413–424
8. Rahm E, Do HH (2000) Data cleaning: Problems and current approaches. IEEE Data Eng Bull 23(4):3–13