

Klaus Koch, Torsten Kaup, Felix Michels, Daniel Elsner, Bernhard Kurpicz, Dirk Malzahn

Schema Matching – Automatische Erkennung semantisch gleicher Attribute

Frankfurt am Main, 22. Mai 2007



Gliederung

- Vorstellung des Teams
- Hintergrund der Thematik Schema-Matching
- Genereller Ablauf des Lösungsansatzes
- Unsere Software: Hot Match!
- Vorstellung der Matching-Algorithmen
- Ausblick
- Live Präsentation



Das Team DaQuMa Westfalen

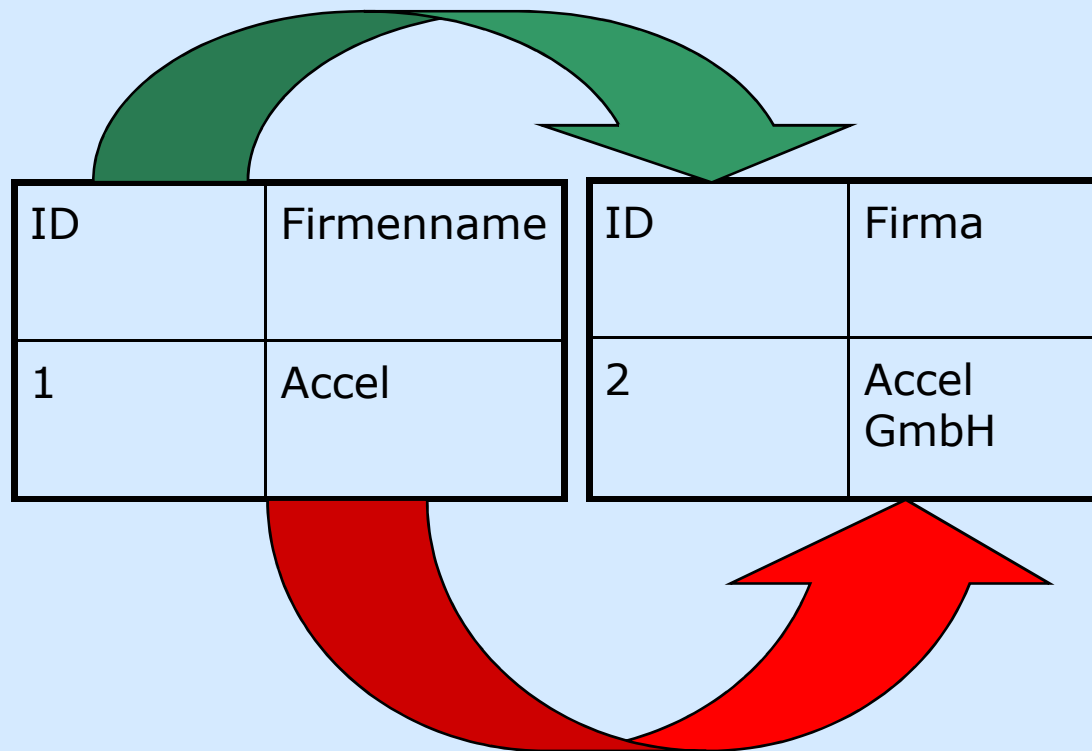
- gegründet zum 3rd IQ-Contest 2006
- Klaus Koch
- Torsten Kaup
- Felix Michels
- Daniel Elsner
- Dirk Malzahn
- Bernhard Kurpicz

Hintergrund der Thematik Schema-Matching

Worum geht es?



1. Vergleich der Spalten und Daten zweier Tabellenschemata
2. Finden von möglichen Matchings innerhalb der Tabellen



Hintergrund der Thematik Schema-Matching

Was ist zu tun?



- Definition von Ähnlichkeitsmerkmalen
- Auswertung dieser Merkmale durch Algorithmen
- automatisierte Analyse der Tabellenspalten
- Aufbereitung des Analyse-Ergebnisses zur weiteren Verarbeitung

Genereller Ablauf des Schema-Matchings

1. Einlesen der Schema-Informationen
2. Erstellung einer Ähnlichkeitsmatrix
3. Verbesserung der Ähnlichkeitsmatrix durch mehrstufige Anwendung von verschiedensten Suchalgorithmen
4. Auswertung anhand eines Kombinationsalgorithmus zu einem Schema-Matching

Hot Match!

Allgemeine Vorgehensweise

- automatischer Import von Schemainformationen
 - Format: Microsoft Excel
- Auswahl der gewünschten Suchalgorithmen
 - modularer Aufbau, dadurch leichte Erweiterbarkeit
- Generierung der Ähnlichkeitsmatrix durch Suchalgorithmen
- Kombinationsalgorithmus
 - finden optimaler Werte in der Matrix



Hot Match!

Implementierte Suchalgorithmen



- Syntaktischer Vergleich der Spaltennamen
- Dictionary zur Analyse nicht-gleichsprachlicher Schemata
- Einheitensuche
- Kategoriensuche

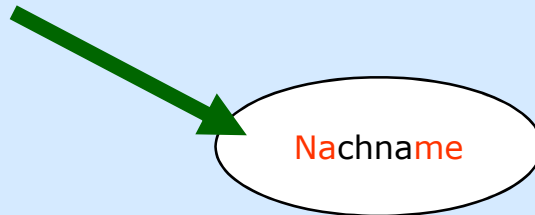


Syntaktischer Vergleich der Spaltennamen I

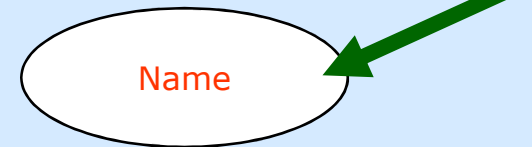
Berechnung der mittleren Übereinstimmung über die Wortlänge

- Zeichenfolgen können sein:
 - Spaltenbezeichnungen
 - Feldinhalte

Anzahl der Buchstaben: 8



Anzahl der Buchstaben: 4



Anzahl identischer Buchstaben: 4

Mittlere Wortlänge: $(8 + 4) / 2 = 6$ Buchstaben

Syntaxkoeffizient: $4 / 6 = \underline{0,67}$

$$M_{SY(AB)} = \frac{ID_{AB}}{\left(\frac{Length_A + Length_B}{2} \right)}$$

Syntaktischer Vergleich der Spaltennamen II

Beispiel für hinterlegte Methodik

- Ansatz: Ermittlung der syntaktischen Überdeckung auf Basis der Zeichen zu vergleichender Feldnamen.

Beispiel 1 – syntaktische Abhängigkeiten

A = {Strasse} B={Str} C={Name}

A und B enthalten identische Buchstaben. Daraus folgt: $ID_{AB} = 3$.

$$MSY_{AB} = ID_{AB} = \frac{3}{\frac{\#A + \#B}{2}} = \frac{3}{\frac{7+3}{2}} = \frac{3}{5} = 0,6$$

$$MSY_{AC} = \frac{2}{\frac{7+4}{2}} = \frac{2}{6} = 0,33$$

Die größere Gemeinsamkeit hat demnach auch rechnerisch A und B (Wert 0,6) und nicht A und C (Wert 0,33). Ein Mapping würde also für $MSY_{AB} = 0,6$ durchgeführt werden.

Ergebnis Syntaktischer Vergleich

Ähnlichkeitsmatrix

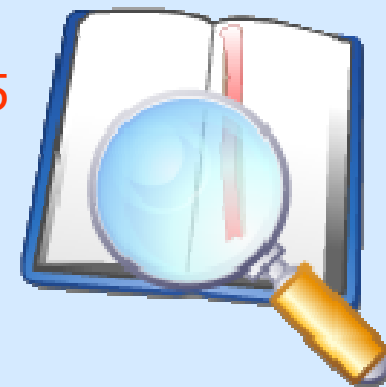


	ID	FIRSTNAME	LASTNAME	STREET_ADDRESS	CITY	ZIP_CODE	COUNTRY	PHONE_NUMBER	COMPANY	AUTO_MODEL
PID	0,80	0,17	0,00	0,12	0,29	0,55	0,00	0,13	0,20	0,00
VNAME	0,00	0,57	0,62	0,21	0,00	0,15	0,17	0,35	0,33	0,27
NNAME	0,00	0,57	0,62	0,22	0,00	0,15	0,17	0,47	0,50	0,40
TELEFON	0,00	0,50	0,53	0,30	0,00	0,27	0,43	0,42	0,29	0,35
GEBURTSORT	0,00	0,32	0,22	0,35	0,14	0,22	0,47	0,45	0,12	0,40
JOB	0,00	0,00	0,00	0,00	0,00	0,18	0,20	0,27	0,20	0,15
VEREIN	0,25	0,53	0,43	0,30	0,20	0,29	0,31	0,44	0,15	0,13
POSITION	0,20	0,47	0,38	0,18	0,33	0,38	0,40	0,30	0,20	0,33

Die Algorithmen von Hot Match!

Dictionary

- Identifiziert mehrsprachige Spaltennamen anhand einer hinterlegten Übersetzungsdatenbank
- Ausgewähltes Dictionary: Deutsch / Englisch
- Vergleich: Vorname zu Firstname
- 1. Syntaktikvergleich: Koeffizient: $4/8 = 0,5$
- 2. Syntaktikvergleich nach Anwendung des Dictionarys: Koeffizient: 1,0

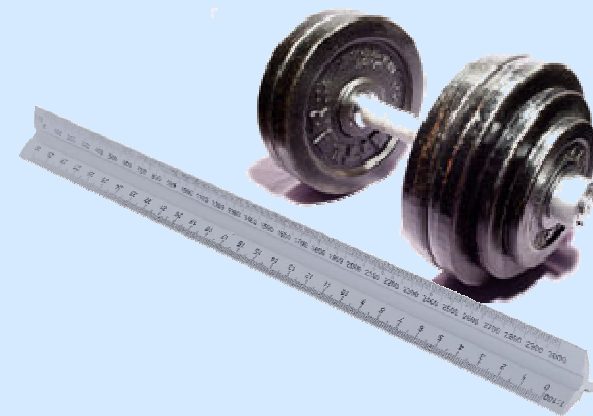


Die Algorithmen von Hot Match!

Einheitensuche



- bestimmt Matches über Einheitenkonvertierung von Werten
- durchsucht Spaltennamen und Daten nach definierten Einheitenkürzeln
 - Erweiterbarkeit des Algorithmus über datenbankbasierende Kürzeldefinitionen
- findet Einheitenkonvertierung durch bekannte Konvertierungsfunktionen
- einfaches Beispiel: $1000\text{g} = 1\text{kg}$





Die Algorithmen von Hot Match!

Kategoriesuche

- findet Matchings anhand von Ähnlichkeiten der Daten
- Analyse der Daten jeder Spalte auf Kategorieeigenschaften
 - nur eine geringe Anzahl von unterschiedlichen Dateneinträgen vorhanden
- sucht Spalten die sich in ihrer Kategorieeigenschaft ähneln
- erstellt ein Matching dieser Spalten
- Beispiel: boolesche Felder, Dropdownliste

Hot Match! Ausblick

Datumssuche



- identifiziert Datumseinträge (Tag, Monat, Jahr, Woche...)
- verfügt über eine Menge von Beziehungen zwischen Datumseinträgen
 - Zusammenfügung
 - Verallgemeinerung
 - Spezialisierung
 - Untermengenbildung
- bildet Matching aus gefundenen Beziehungen
 - Spalte Geburtsdatum → identifiziert tag, monat, jahr
 - findet Beziehung → Datum = concat(tag, monat, jahr)
 - Matching: Geburtsdatum = concat(Geburtstag, Geburtsmonat, Geburtsjahr)





Hot Match! Ausblick

Numerische Suche

- Findet Ähnlichkeiten bei Wertebereichen von Spalten
- Findet Ähnlichkeiten bei Wertebereichen von berechneten Spalten
- Matcht Spalten die in ihrer Wahrscheinlichkeitsfunktion Ähnlichkeiten aufweisen

Live Präsentation





Vielen Dank für Ihre Aufmerksamkeit!

Fazit

- unsere Software liegt irgendwo zwischen Genie und Wahnsinn



Die Algorithmen von Hot Match!

Schema-Mismatch-Suche

- Vergleicht die Daten eines Schemas mit dem Schema eines anderen
- identifiziert zuerst binäre Attribute eines Schemas
- durchsucht dann die Daten des Vergleichsschemas nach der Spaltenbezeichnung des binären Attributs
- Matching, wenn in einer Spalte Datenstamm eine bestimmte Anzahl Übereinstimmungen gefunden wurden
- identifiziere diese Spalte als Kategorie
- Matche die Datensätze die dem binären Attribut entsprechen