Informationsintegration

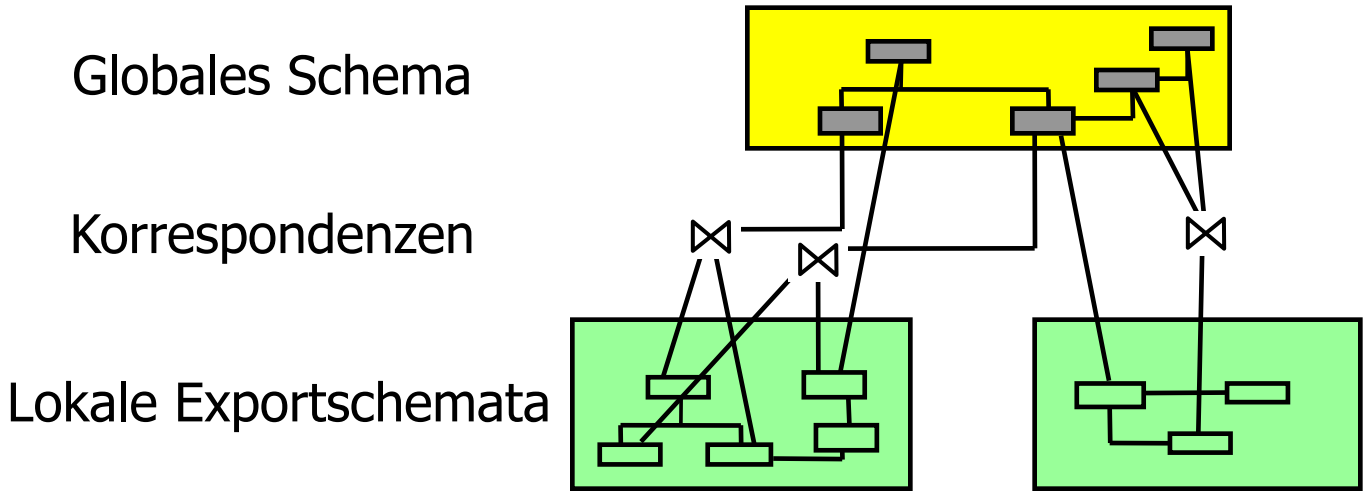
Duplikaterkennung

Ulf Leser

Inhalt dieser Vorlesung

- Data Cleansing
- Duplikaterkennung
- Sorted-Neighborhood Algorithmus
- Duplikaterkennung auf Bäumen und Graphen

Anfragebearbeitung in Mediator-basierten Syst.



- Aufgabe der Anfragebearbeitung
  - Gegeben eine Anfrage q gegen das globale Schema
  - Gegeben eine Menge von Korrespondenzen zwischen globalem und lokalen Schemata
  - Finde alle Antworten auf q

Ergebnisintegration

- Schritt 1: Anfrageplanung
- Schritt 2: Anfrageübersetzung
- Schritt 3: Anfrageoptimierung
- Schritt 4: Anfrageausführung
- Schritt 5: Ergebnisintegration

Ergebnis-integration	**Eingabe**: Pro Ausführungsplan eine Menge von Ergebnistupeln **Aufgabe**: Die Ergebnisse der einzelnen Ausführungspläne sind oftmals redundant oder uneinheitlich. Während der Ergebnisintegration werden die einzelnen Ergebnisse zu einem Gesamtergebnis integriert, was insbesondere Duplikaterkennung und Auflösen von Widersprüchen in den Daten erfordert. **Ergebnis**: Das Ergebnis der Benutzeranfrage

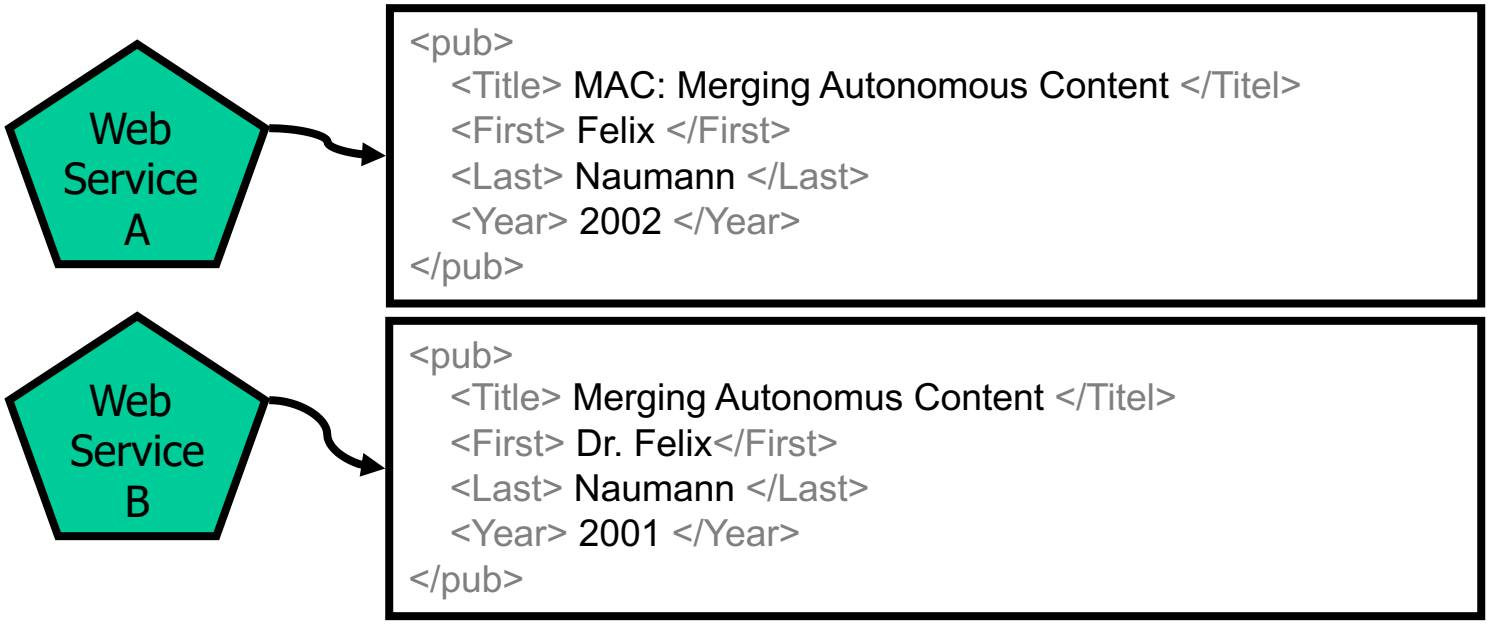
Datenreinigung

- Jeder Plan erzeugt Ergebnistupel
- Ziel: Einheitliche, nicht redundante Darstellung der integrierten Informationen
- Probleme: Duplikate, fehlende Werte, Widersprüche etc.
- Allgemein: Geringe Qualität der Daten
- Abhilfe: Data Cleansing
  - Duplikaterkennung
  - Datenfusion

Data Cleansing in integrierten Systemen

- Föderierte Systeme
  - Online (während Anfragebearbeitung)
  - Passiert im Mediator
  - Erhält Ergebnisse aller Pläne und bereitet diese auf
  - Zugriff meist nur auf die aktuellen Anfrageergebnisse
- Data Warehouses
  - Offline (eventuell schon bei den Quellen)
  - Passiert während des ETL-Prozesses
  - Erzeugt persistente und qualitativ hochwertige Daten
  - Zugriff auf gesamten Datenbestand möglich
  - Muss ggf. bei jeder Neu-Integration wiederholt werden

Beispiel bei virtueller Integration



Title	First	Last	Year
MAC: Merging Autonomous Content	Felix	Naumann	2002
Merging Autonomus Content	Dr. Felix	Naumann	2001

Nachvollziehbarkeit

- Data Cleansing verändert Daten
- (Neue) Daten müssen erklärbar sein
  - Anwender eines DWH: „Da fehlt ein Produkt im Report“
  - Analysewerkzeug fehlerhaft? Report falsch? Data Mart Definition falsch? Basisdatenbank unvollständig? ETL Prozeduren fehlerhaft?
- Data Cleansing Aktionen müssen nachvollziehbar sein
  - Protokollieren der Aktionen durch alle Prozessschritte
  - Wiederholbarkeit aller Aktionen bei neuen Daten / Anfragen
    - Schwierig: Unsystematische ad-hoc Aktionen
    - DC programmieren, keine manuellen Änderungen
  - Mühsam, aber notwendig zur Unterstützung kontinuierlicher Prozesse

Duplikate und Fusion

- Viele Fehler findet man erst durch Vergleich
  - Quelle 1: Hans Meyer, Frankfurter Allee 200
  - Quelle 2: Hans Meier, Frankfurter Alee 202
- Das ist nur sinnvoll, wenn Duplikate vorliegen

- Duplikaterkennung: Identifikation verschiedener Repräsentationen desselben Objekts
- Datenfusion: Verschmelzen von Duplikate zu einem Objekt
  - Nicht-redundant: Jedes Objekt ist nur einmal im Ergebnis
  - Homogen: Jedes Attribut hat nur einen (richtigen) Wert

Inhalt dieser Vorlesung

- Data Cleansing
- Duplikaterkennung
- Sorted-Neighborhood Algorithmus
- Duplikaterkennung auf Bäumen und Graphen

Fehlerquote auf No-Fly Listen

- Nach dem 11.9. 2001 hat die Transportation Security Administration (TSA), die nach der Gründung des Heimatschutzministeriums ein Teil dessen wurde, begonnen, eine No-Fly-Liste und eine "Selectee"-Liste über Menschen anzulegen, die eine Bedrohung darstellen können. Im Oktober 2002 wurde nach einem Antrag nach Einsicht gemäß dem Informationsfreiheitsgesetz erstmals belegt, dass solche, bislang heimlich geführten Listen existieren, nach denen Personen am Fliegen gehindert oder einer strengeren Kontrolle unterzogen werden. Zehntausende von Namen, vermutlich über 40.000, soll die No-Fly-Liste mittlerweile enthalten, in der Terrorist Identities Datamart Environment-Datenbank (TIDE), der Stammliste, aus der die andere Listen abgeleitet werden, sind über 300.000 Personen gespeichert (Die Mutter aller Terror-Datenbanken quillt über).
- Das Problem der nach geheim gehaltenen Kriterien gesammelten Einträge auf den Listen ist neben einer gewissen Willkürlichkeit vor allem die hohe Fehlerquote. Sie soll bis zu 50 Prozent betragen, wie aus einem Bericht des Government Accountability Office hervorging. Fehler treten vorwiegend beim Vergleich der Namen auf. Ähnlichkeiten der Namen von Reisenden mit denen auf den Terrorlisten führen zu Verwechslungen, die unangenehme Folgen haben können, zumal es äußerst schwierig ist, einen Namen wieder von der Liste streichen zu lassen, und es undurchsichtig ist, warum ein Name oder eine Person auf diese geraten ist. Man darf gespannt sein, welche Folgen etwa die Antiterrorliste in Deutschland haben wird.
- Ein entscheidendes Problem in den USA stellt das Programm zum Abgleich der Namen dar. Mit dem Soundex-Programm werden die Namen von Reisenden mit denen der No-Fly-Liste verglichen. Die auf der englischen Sprache basierende Suche stößt natürlich in den Reservierungsdatenbanken auf Probleme, wenn dort etwa arabische Namen in das römische Alphabet übersetzt werden, was bekanntlich in zahlreichen Versionen gemacht werden kann.
  - An sich hatte sich bereits ein anderes Programm in den Startlöchern befunden, um Verdächtige auszusortieren, nämlich CAPPS II (Computer Assisted Passenger Pre-screening System II). Hier wären neben den Namen noch weitere persönliche Daten wie Adresse, Telefonnummer und Geburtstag zur Identifizierung eingeflossen. Die Informationen wären dann mit weiteren persönlichen Daten wie Kreditkarteninformationen, Finanztransaktionen und vielen anderen, von privaten Datenunternehmen gesammelten Daten abgeglichen und ergänzt worden, um eine Risikobewertung durchzuführen.
  - Obgleich vom Kongress die Gelder für CAPPS II 2004 aus Datenschutzgründen gesperrt wurde, ist Ende des letzten Jahres bekannt geworden, dass die TSA weiterhin mit dem Automated Targeting System (ATS) eine Risikobewertung von international Reisenden durchführt, wobei die Flugpassagierdaten mit weiteren Daten aus unterschiedlichen Quellen verbunden werden (US-Regierung bewertet das Risikopotenzial aller Ein- und Ausreisenden). Ob ATS Vorläufer oder Teil des Secure Flight-Programms ist, ist angesichts der verwirrenden Vielzahl von Projekten und Plänen zur Überwachung nicht eindeutig. Wie beim Secure Flight-Programm sollen jedoch bei ATS auch keine Daten von privaten Datenhändlern wie ChoicePoint verwendet werden. Das soll allerdings beim Registered Flight-Programm gemacht werden, an dem sich Reisende freiwillig beteiligen können, um nach einem Hintergrundcheck, der von beauftragten Firmen durchgeführt wird, schneller durch die Kontrollen zu kommen.
  - Nachdem Zehntausende von Beschwerden aufgrund von Verwechslungen aufgekommen sind und die Listen unter schwere Kritik geraten sind, hat nun das Heimatschutzministerium eine Website zur Abhilfe eingerichtet. Auf DHS TRIP (Traveler Redress Inquiry Program) können Reisende, die meinen, fälschlicherweise auf der Liste zu stehen, ihre Beschwerde einreichen. Sie müssen dazu zusätzliche Informationen eingeben, die überprüft und auch an andere Behörden weiter gegeben werden können. Mit TRIP lasse sich, so Minister Chertoff, die Liste säubern, die angeblich aber schon gründlich überprüft worden sei, um endlich das lange angekündigte Programm Secure Flight nächstes Jahr einführen zu können.
- Beim Secure Flight-Programm werden PNR-Daten von Reisenden vor allem mit den Informationen aus der Terrorist Screening Database (TSDB) abgeglichen, um "verdächtige und bekannte Terroristen" zu erkennen und am Besteigen von Flugzeugen zu verhindern. Dazu will man offenbar die Terror-Liste bereinigen, auch wenn man mit den PNR-Daten mehr Informationen zur Überprüfung besitzt als nur die Namen. Das Problem mit Soundex ist damit allerdings noch nicht ganz gelöst. Angeblich hofft man im Heimatschutzministerium mit biometrischen Datenbanken die bislang dank Soundex hohen falschen Trefferquoten reduzieren zu können. Der Plan, nicht mehr nur zwei Fingerabdrücke, sondern die digitalen Fingerabdrücke aller 10 Finger zu erhalten, könnte damit zusammenhängen.
- Quelle: http://www.heise.de/tp/r4/artikel/25/25058/1.html

Ein Siebtel der Einträge in der britischen Gendatenbank ist falsch

- Die britische Polizei hat die weltweit größte Gendatenbank und wahrscheinlich die fehlerhafteste, was die Glaubwürdigkeit der Genprofile untergräbt
  - Die britische Polizei hat die größte Gendatenbank der Welt, die in den letzten Jahren massiv ausgebaut wurde. Mittlerweile enthält sie die Genprofile von 4 Millionen Menschen, mehr als 5 Prozent der Gesamtbevölkerung. In jedem Jahr wächst die Datenbank um eine halbe Million weiterer Genprofile. Dabei handelt es sich nicht nur um die Gendaten von verurteilten Straftätern, sondern auch von denen, die einer Tat verdächtigt und festgenommen wurden, gegen die aber dann keine Anklage erhoben wurde.
  - Auch bei teils trivialen Verstößen, beispielsweise im Sinne der ständig erweiterten Liste des "antisozialen Verhaltens", können Genprofile gemacht und gespeichert werden. Auch die Gendaten von 150.000 Kindern unter 16 Jahren finden sich in der Datenbank, auch hier handelt es sich keineswegs immer um Straftäter.
  - Die Begehrlichkeiten der Strafverfolger sind hoch. Das Innenministerium beabsichtigt, die Abnahme des genetischen Fingerabdrucks noch einmal massiv zu erweitern. So würde man auch gerne durchsetzen, dass DNA-Proben von Verdächtigen, die nur kleiner Vergehen beschuldigt werden, neben Fotografien und Finger- und Fußabdrücken schon auf der Straße genommen werden können. Wer nicht angeschnallt mit dem Auto fährt oder Abfall auf die Straße wirft, könnte dann schon für immer in die Datenbank geraten. Der Zweck sei, so heißt es in einem Papier des Innenministeriums, damit besser Straftäter identifizieren und Datenbanken durchsuchen zu können.
- Allerdings könnten die Sammelwut und das Vertrauen in die Korrektheit der Genprofile und der darüber bewerkstelligten Nachweise mit den neuesten Informationen einen Dämpfer erleiden. Um die 550.000 Namen in der Datenbank, so stellte sich nun heraus, sind falsch, falsch geschrieben oder fehlerhaft. Das macht immerhin Siebtel der gesamten Genprofile aus. Dabei handelt es sich nur um Kopien von DNA-Proben.
- Der Grund ist offensichtlich auch die große Sammelwut, die die Polizisten, getrieben vom Innenministerium, betreiben. So werden DNA-Proben bei Festnahmen genommen, während die Namen der Festgenommen nicht überprüft oder in falscher Schreibweise eingegeben werden. Manchmal stimmen Namen und Genprofile nicht überein oder sind Genprofile unter dem Namen von Menschen registriert, die es gar nicht gibt.
- Die Regierung musste nun sogar einräumen, nicht genau zu wissen, wie viele Einträge wirklich falsch sind, da von der Zählung nur die vorhandenen Kopien erfasst wurden. Es könnten also weitaus mehr sein. So sagte Meg Hillier, Staatssekretärin des Innenministeriums, dass die Zahl der Personen, deren Genprofil in der Datenbank ist, etwa um 13,7 Prozent niedriger sei als die Zahl gesamten Einträge. Man kann offensichtlich ohne Schuld mit seinem Genprofil auf Dauer in die Datenbanken gelangen, aber auch womöglich unschuldig darüber einer Straftat bezichtigt werden, weil Namen vertauscht oder Genprofile falsch zugeordnet wurden.
- Das Innenministerium versichert, dass man nun hart daran arbeite, die Ungenauigkeiten zu beseitigen und die Genprofile mit den Fingerabdrücken abzugleichen, um die Identität der Personen zu überprüfen.
- Quelle: http://www.heise.de/tp/r4/artikel/26/26061/1.html

Duplikate

- Relationale Welt: Duplikat = Zwei Tupel einer Relation die identische Werte in allen Attributen besitzen
- Allgemein: Duplikat = Paar von Tupeln, die demselben Realweltobjekt entsprechen
  - Also „synonyme" Objekte
- Typische Beispiele: Personen, Rechnungen, Lieferungen, Bestellungen, …
- Viele kommerzielle Tools, vor allem im CRM
- Viele Synonyme: Duplicate Detection, Record Linkage, Object Identification, Deduplication, Entity Resolution, …

Auswirkungen

- Überflüssiger Verbrauch von Plattenplatz und Rechenleistung
- Einfachste Auswertungen sind falsch
  - Anzahl Kunden != SELECT COUNT(*) FROM customer
  - Umsatz != SELECT SUM(revenue) FROM order
- Man sieht nur noch Teile und nicht das Ganze
  - Überschreiten des Kreditlimits wird nicht erkannt
  - Kein Ausnutzen von Mengenrabatten
  - Fehleinschätzung von Kunden
  - Mehrfachzusendung von Katalogen
  - …

Das formale Problem

- Gegeben: Tupel $T=\{t_1,..,t_m\}$ mit Attributen $A_1,...,A_k$
  - Komplexere Modelle (Bäume, Graphen) später
- Tupel entsprechen Objekten $O=\{o_1,...,o_n\}$
  - Mit $n \leq m$
- Manche der Tupel sind Duplikate
- Gesucht: eine Funktion dup: $TxT \rightarrow bool$ die true zurückgibt, wenn Tupel t_1 und t_2 demselben Objekt o entsprechen
  - Normalisierung verlangt auch die Zurückgabe von o – schwer!
  - Fusion konstruiert eine Repräsentation von o

Der gängige Ansatz (Wiederholung)

- Man nimmt an, dass Duplikate ähnlich sind
- Ähnlichkeit misst man durch eine geeignete Ähnlichkeits-funktion sim: TxT $\rightarrow$ [0,1]
- Verwend. eines Schwellwerts t:
  $dup_{sim}(t_1,t_2)$=true gdw. $sim(t_1,t_2)>t$

- Daher muss man über Qualität sprechen

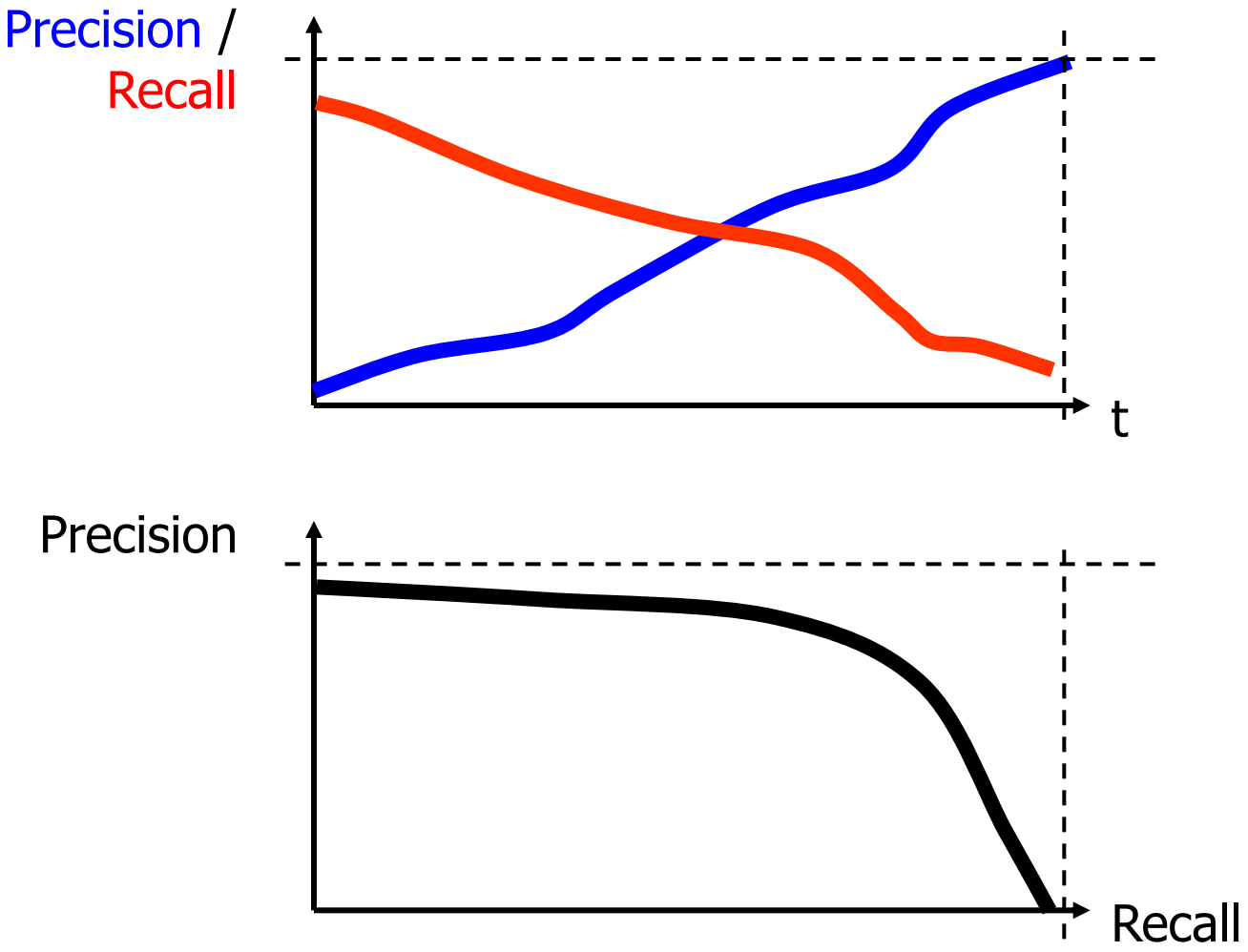
Precision und Recall

Realität

Methode	Duplikat	Kein Duplikat
Duplikat	true-positive	false-positive
Kein Duplikat	false-negative	true-negative

- Precision = TP/(TP+FP)
  - Wie viele der vorhergesagten Duplikate sind wirklich welche?
- Recall = TP/(TP+FN)
  - Wie viele der echten Duplikate haben wir gefunden?

P/R bei verschiedenen Schwellwerten t



Transitivität

- Im strengen Sinne ist die Duplikateigenschaft transitiv:
  $dup(t_1,t_2)=true \wedge dup(t_2,t_3)=true \rightarrow dup(t_1,t_3)=true$
  - Das sind ja alles dieselben Realweltobjekte
- Aber: Ähnlichkeit+Schwellwert ist mitnichten transitiv
  - Meier, Meyer, Mayer, Bayer, Bayes, Bades, …
  - Meier, Meir, Mer, Er, R, …
- Gut überlegen, bevor man Transitivität ausnutzt
  - Hängt ab vom Vertrauen in sim / t
  - Kann Fehlerraten erhöhen oder verringern

Äquivalenzrelation und Clustering

- Annahme: Transitivität
- Dann ist dup eine Äquivalenzrelation
  - dup partitioniert T in Cluster
  - Jeder Cluster entspricht einem Realweltobjekt
  - Jedes Tupel in einem Cluster repräsentiert sein Realweltobjekt
- Duplikaterkennung ist im Grunde ein Clusteringproblem
  - Gängige Clusterverfahren z.B. k-Means oder hierarchisch
  - Diese können auch mit Verstößen gegen Transitivität umgehen

Prinzipielles Vorgehen

- Wir betrachten das ursprüngliche Problem: Finde Paare von Tupeln, die Duplikate sind

- Annahme: Verwendung einer „guten" sim
- Um alle Duplikate zu finden, muss man alle Paare miteinander vergleichen
- Beim Vergleich geht man attributweise vor
  - Schema Matching muss also schon vorher erfolgt sein
- Ähnlichkeit zweier Tupel setzt sich aus der Ähnlichkeit ihrer Attributwerte zusammen, ggf mit Gewichtung der Attribute

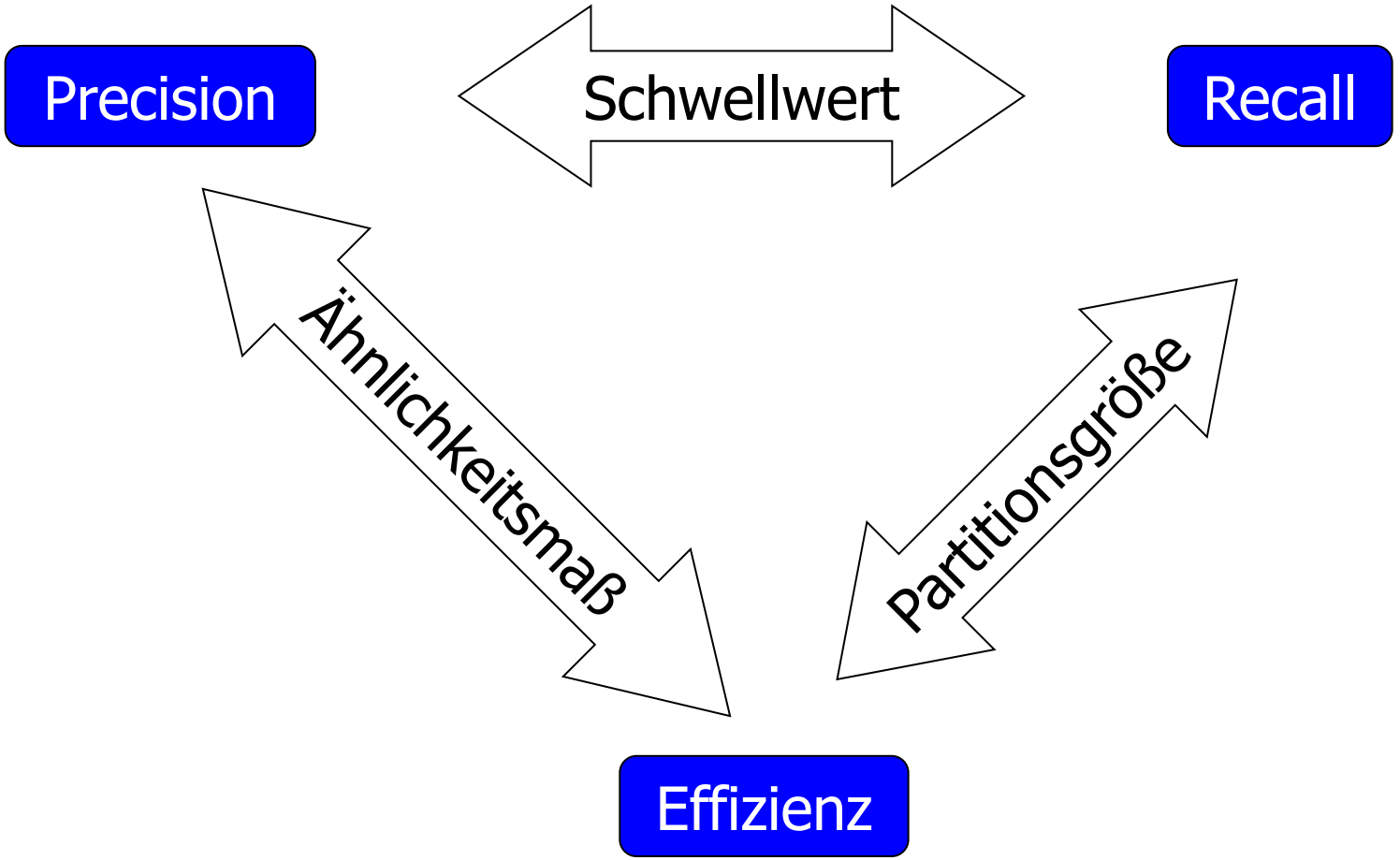
Probleme

- Geeignete Ähnlichkeitsmaße?
- Beitrag von Attributen zur Tupelähnlichkeit?
- Skalierbarkeit / Laufzeit
  - Der Vergleich zweier Tupel ist nicht billig – oftmals komplexe Vergleichsfunktionen
  - Es gibt $O(n^2)$ viele Tupelpaare
  - Nur sehr wenige sind tatsächlich Duplikate
    - Beispiel: 1000 Tupel, 50 Duplikate - ~500.000 Vergleiche, von denen 0,0001% TRUE ergeben
- Wir müssen weg von der quadratischen Komplexität
  - Im Unterschied zu einem Join kann man hier nicht sortieren

Partitionierung zur Beschleunigung

- Statt alle Paare zu vergleichen, nimmt man eine Vor-Partitionierung der Daten vor
  - Tupel dann nur innerhalb der Partitionen vergleichen
- Kleine Partitionen: Weniger Vergleiche, aber auch schlechterer Recall, wenn Duplikate in verschiedenen Partitionen liegen
- Große Partitionen: Viele Vergleiche, schlechtere Effizienz, aber besserer Recall

Trade-Offs


Precision
Schwellwert
Recall
Ähnlichkeitsmaß
Partitionsgröße
Effizienz

Inhalt dieser Vorlesung

- Data Cleansing
- Duplikaterkennung
- Sorted-Neighborhood Algorithmus
- Duplikaterkennung auf Bäumen und Graphen

Sorted Neighborhood [HS98]

- Sorted Neighborhood Algorithmus
  - Sort-Phase: Sortiere Tupel so, dass Duplikate (hoffentlich) nahe beieinander sind
  - Merge-Phase: Fenster der Größe w über sortierte Liste schieben und alle Tupel innerhalb eines Fensters miteinander vergleichen
    - Überlappende Partitionen
  - Purge-Phase: Duplikate verschmelzen (Fusion)
- Komplexität für n Tupel (Schlüsselerzeugung sei linear)
  - Sortieren ist O(n*log(n))
  - Anzahl Vergleiche ist O(n*w)

Schlüsselerzeugung

- Schlüsselerzeugung zentral für Effektivität des Verfahrens
  - Schlechte Reihenfolge: Schlechter Recall
- Leider ist überhaupt nicht klar, was ein guter Schlüssel zur Duplikaterkennung ist
  - Implizite Priorisierung der Attribute durch Schlüsselstruktur
  - Wichtig: Domänenwissen

Vorname	Nachname	Adresse	ID
Sal	Stolpho	123 First St.	456780
Mauricio	Hernandez	321 Second Ave	123456
Felix	Naumann	Hauptstr. 11	987654
Sal	Stolfo	123 First Street	456789

Schlüssel
STOSAL123FRST456
HERMAU321SCND123
NAUFEL11HPTSTR987
STOSAL123FRST456

Merge (Fenstergröße 2)

Vorname	Nachname	Adresse	ID	Schlüssel
Mauricio	Hernandez	321 Second Ave	123456	HERMAU321SCND123
Felix	Naumann	Hauptstr. 11	987654	NAUFEL11HPTSTR987
Sal	Stolpho	123 First St.	456780	STOSAL123FRST456
Sal	Stolfo	123 First Street	456789	STOSAL123FRST456

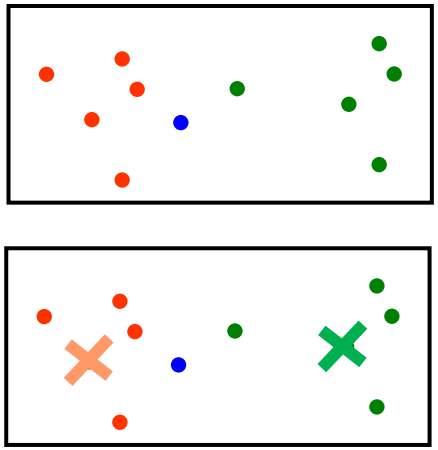
- Regelbasierte Entscheidung auf Duplikat, z.B.
  - **IF last_name(r_1) = last_name(r_2)**
    AND edit_distance(first_name(r_1), firstname(r_2)) < 5,
    AND address(r_1) = address(r_2)
    THEN dup(r_1,r_2)=true
  - **IF (ID(r_1) = ID(r_2) OR last_name(r_1) = last_name(r_2))**
    AND address(r_1) = address(r_2)
    AND city(r_1) = city(r_2)
    AND (state(r_1) = state(r_2) OR zip(r_1) = zip(r_2))
    THEN dup(r_1,r_2)=true

Aufwand

- Falls die Daten nicht in Hauptspeicher passen – viel IO
  - Mindestens drei vollständige Scans notwendig
    - Schlüssel erzeugen und speichern
    - Sortieren
    - Fenster schieben
- Praktisch sollte sich dann w an der Blockgröße und der Größe des Hauptspeichers orientieren
  - Partitionen, die kleiner als Disk-Blöcke sind, sparen kein IO

Ausnutzung von Transitivität

- Kann zur Reduktion von Arbeit benutzt werden
  - Pro Fenster merkt man sich k Äquivalenzklassen statt w Tupel (k<w)
  - Pro Äquivalenzklasse ein Vertreter
    - Schwierige Wahl, z.B. Medoid
- Bei vielen Duplikaten: k << w
- Sonst lohnt sich der Aufwand kaum
- Preis: Erhöhung der Fehlerrate



Alternative zu Mediode-Methode

- Medoide berechnen (und Updaten) ist sehr teuer
  - Immer Vergleich aller Paare im Cluster notwendig
- Alternative: Heuristik mit Ausnutzung der Sortierung
  - Jeder Cluster wird als Priority Queue organisiert
    - Sortiert nach Reihenfolge des Einfügens von Tupeln
  - Neues Tupel t
    - Benutze nur das höchste Element t‘ pro Cluster zum Vergleich
      - Eventuell: Top-k Tupel
    - t‘ liegt bzgl. Sortierung am nächsten zu t
    - Höchste Chance auf Duplikateigenschaft
- Heuristik zur Beschleunigung mit Auswirkungen auf P/R
  - Siehe Trade-Off: Schneller, aber vermutlich schlechtere Qualität

Multipass Verfahren

- Problematischer Schritt: Schlüsselwahl
- Lösung 1: Vergrößerung des Fensters
  - Effizienz leidet, Effektivität steigt idR nur wenig
  - Die „Präfix-Verzerrung" der Sortierung kriegt man nicht weg
- Lösung 2: Multipass
  - Algorithmus mehrmals mit verschiedenen Schlüsseln laufen lassen
  - Pro Lauf kleine w und einfache Schlüssel verwenden
  - Duplikatbeziehungen aller Läufe kombinieren
  - Weniger effizientes, aber deutliches effektiveres Verfahren als Single-Pass

Inhalt dieser Vorlesung

- Data Cleansing
- Duplikaterkennung
- Sorted-Neighborhood Algorithmus
- Duplikaterkennung auf Bäumen und Graphen