Analyse, Design, Entwicklung und Evaluation eines skalierbaren, Echtzeit Entity Resolution Streaming Framework

---

Kevin Sapper

19.10.16
Version: 0.1

Hochschule RheinMain

DCSM - Design Informatik Medien
Informatik (M.Sc.)

Masterarbeit

Analyse, Design, Entwicklung und Evaluation eines skalierbaren, Echtzeit Entity Resolution Streaming Framework

Kevin Sapper

Referent Prof. Dr. Adrian Ulges
Hochschule RheinMain
DCSM - Design Informatik Medien

Koreferent Prof. Dr. Reinhold Kröger
Hochschule RheinMain
DCSM - Design Informatik Medien

Betreuer Thomas Strauß
Universum Group

19.10.16

Kevin Sapper
Analyse, Design, Entwicklung und Evaluation eines skalierbaren, Echtzeit Entity Resolution Streaming Framework
Masterarbeit, 19.10.16
Referenten: Prof. Dr. Adrian Ulges und Prof. Dr. Reinhold Kröger
Betreuer: Thomas Strauß

Hochschule RheinMain
Informatik (M.Sc.)
DCSM - Design Informatik Medien
Kurt-Schumacher-Ring 18
65197 Wiesbaden

Inhaltsverzeichnis

1 Einleitung

Die Masterarbeit soll in Zusammenarbeit mit der Firma Detim Consulting GmbH geschrieben werden. Dazu wird ein Problemfeld bei dem Kunden Universum Group, der Detim Consulting GmbH, in deren Geschäftsfeld dem Inkasso-, Liquiditäts-, und Risikomanagement, gewählt.

2 Problemfeld Universum Group

Die Universum Group bietet Lösungen für Onlineshops zur Bonitäts- und Adressprüfung, sowie dem Forderungsankauf, der Onlineshop-Kunden. Dabei wird dem Händler bei entsprechender Bonität seines Kunden das Angebot gemacht, die Forderung, nach Ablauf einer Zahlungsperiode, zu 100 % zu übernehmen. Damit eine möglichst zuverlässige Aussage, über die Bonität des Kunden, getroffen werden kann, muss zunächst herausgefunden werden, ob der Kunde bereits bei der Universum Group bekannt ist. Das Problem an dieser Stelle ist, dass der Kunde Online seine Daten selbst erfasst und diese nicht anhand von Personalausweis oder ähnlichen Dokumenten überprüft werden können. Fehler bei der Datenerhebung sind, beispielsweise unterschiedliche Schreibweisen, insbesondere bei Adressen, Tippfehler, welche bei Namen nicht offensichtlich sind, unterschiedliche Konventionen, etwa Str. für Straße, oder akademische Titel und Adelstitel, welche in Onlineformularen nicht standardisiert erfasst werden. Bei der Bonitätsprüfung dient die Personenidentifizierung dazu, Kunden mit positiver oder negativer Zahlungsmoral zu erkennen und anzunehmen bzw. abzulehnen. Je genauer die Personenidentifikation ist, desto aussagekräftiger sind die Bonitätsauskunfte von externen Dienstleister, beispielsweise der Schufa. Beim Inkassomanagement gilt zudem das sog. Schadensminderungsprinzip. Das bedeutet, das alle angekauften Forderungen eines Kunden nur einmalig abgemahnt werden dürfen. Daher müssen hier Personendubletten gefunden und zusammengeführt werden.

Das aktuelle System zur Personenidentifizierung funktioniert nur bei der Bonitätsprüfung und ist durch einen externen Dienstleister realisiert. Dieser bereinigt und prüft Namen und Adressen. Allerdings skaliert das System dabei nur innerhalb eines vorgegebenen monatlichen Kontingents.

3 Duplikatserkennung

Die Methoden zur Duplikatserkennung stammen ursprünglich aus dem Gesundheitsbereich (Felegi & Sunter 1969). Je nach Fachgebiet gibt es unterschiedliche Fachbegriffe. Statistiker und Epidemiologen sprechen von record oder data linkage während Informatiker das Problem unter entity resolution, data oder field matching, duplicate detection, object identification oder merge/purge kennen. Dabei geht es nicht um die reine Personenidentifikation, sondern vielmehr um die Identifikation von Entitäten aller Art, beispielsweise Kunden, Patienten, Produkte oder Orte. Dabei können die Entitäten nicht durch ein einzigartiges Attribut identifiziert werden. Zudem sind die Datensätze oft fehlerhaft, beispielsweise durch Rechtschreibfehler oder unterschiedliche Konventionen. Die Methoden zur Entitätsauflösung arbeiten meist auf Datensatzpaaren und liefern als Ergebnis eine Menge von Übereinstimmungen. Eine Übereinstimmung verknüpft zwei Entitäten. Zusätzlich kann über einen optionalen Ähnlichkeitswert (engl. similarity score), normalerweise zwischen 0 und 1, die Intensität der Übereinstimmung angegeben [1].

Zur Bestimmung der Ähnlichkeit eines Datensatzpaares unterscheiden Elmagarmid et al. [???] zwischen Attributsvergleichs- (engl. field matching) und Datensatzvergleichsmethoden (engl. record matching). Methoden zum Attributsvergleich sind zeichenbasierend (edit distance, affine gap distance, Jaro distance metric oder Q-gram distance), tokenbasierend (atomic strings, Q-grams mit tf.idf), phonetisch (soundex) oder nummerisch. Die Datensatzvergleichsmethoden sind probabilistisch (Naive Bayes), überwachtes bzw. semi-überwachtes Lernen (SVMLight, Markov Chain Monte Carlo), aktives Lernen (ALIAS), distanzebasierend (siehe Attributsvergleich - Datensatz als konkatenierter String) oder regelbasierend (AJAX). Die Ausführung der Vergleichsmethoden ist enorm teuer, da diese das Kreuzprodukt zweier Mengen bilden müssen. Um die Ausführungszeit zu reduzieren wird versucht den Suchraum auf die wahrscheinlichsten Duplikatsvorkommen zu begrenzen. Diese Vorgehen werden als Blocking oder Indexing bezeichnet. Elmagarmid et al. nennen Standard Blocking, Sorted Neighboorhodd Approach, Clustering und Canopies, sowie Set Joins als Vorgehensweisen. (Referenzen zu den Methoden folgen noch!)

Da es keine Methode zur Entity Resolution gibt, welche allen anderen überlegen ist, wurden Ende der 90er Jahre begonnen Frameworks zu entwickeln, welche verschiedene Methoden miteinander kombinieren. Einen Vergleich dieser Frameworks wurde durch Köpcke & Rahm 2010 [???] durchgeführt. Ein Framework besteht aus verschie-

denen Matchern. Ein Matcher ist dabei ein Algorithmus, welcher die Ähnlichkeit zweier Datensätze ermittelt. Ähnlich wie Elmagarmid et al. unterscheiden Köpcke & Rahm zwischen attributs- und kontextbasierenden Matchern. Als Kontext bezeichnen Sie die semantische Beziehung bzw. Hierarchie zwischen den Attributen. Um die Matcher miteinander zu kombinieren nutzen die Frameworks min. eine Matching Strategie. Eine Strategie ist, die Ähnlichkeitswerte verschiedener Matcher nummerisch zu kombinieren, beispielsweise durch eine gewichtete Summe oder einen gewichteten Durchschnitt. Ein anderer Ansatz ist regelbasierend. Eine einfache Regel besteht aus einer logischen Verbindung und einer Match-Kondition, beispielsweise einem Schwellenwert. Die dritte und komplexeste Strategie ist workflow-basierend. Hierbei kann beispielsweise eine Sequenz von Matchern die Ergebnisse iterativ einschränken. Grundsätzlich können Workflows beliebig komplex werden. Einen passenden Workflow zu finden kann selbst Domainexperten vor eine große Herausforderung stellen. Daher gibt es trainingbasierende Ansätze passende Parameter für Matcher oder Kombinationsfunktionen (z.B. Gewicht für Matcher) zu bestimmen. Solche Ansätze sind etwa, Naive Bayes, Logistic Regression, Support Vector Maschine oder Decision Trees. (Referenzen zu den Ansätzen folgen noch!)

Ein Großteil der Forschung in Entity Resolution konzentriert sich auf die Qualität der Vergleichsergebnisse. Die von Köpcke & Rahm verglichenen Frameworks konzentrieren sich alle Samt darauf zwei statische Mengen zu miteinander vergleichen. Bei großen Datenmengen kann dies durchaus mehrere Stunden dauern. Daher gibt es in den letzten Jahre einige Ansätze und Frameworks, welche MapReduce zum Skalieren nutzen [???][???]. Zudem gibt es immer mehr Bedarf, Vergleichsergebnisse in nahe Echtzeit zu liefern. Erste Ergebnisse Entity Resolution skalierbar und in nahe Echtzeit zu erreichen, präsentieren Christen & Gayler in [???] 2008, unter Verwendung von Inverted Indexing Techniken, welche normalerweise bei der Websuche anwendung finden. Dabei betrachten Sie vor allem die Anforderungen eines Anfragestroms (engl. query stream). Ihre Anforderungen sind einen Strom von Anfragedatesätzen, gegen potentielle riesige Datenmengen, im Subsekundenbereich pro Anfrage abzuarbeiten. Dabei sollen die Treffer der Anfrage mit einem Ähnlichkeitswert versehen sein. Zudem muss es möglich sein die Menge an Anfragen zu skalieren. Das Hauptproblem ist hierbei die Skalierung. Um skalieren zu können wird versucht die Abarbeitung des Suchraums zu parallelisieren. Eine Studie von Kwon, Balazinska, Howe, & Rolia [???] in MapReduce Anwendungen zeigt, das selbst geringe Ungleichgewichte bei der Verteilung des Suchraum auf Mapper bzw. Reducer, aufgrund der Komplexität der Matching Algorithmen, zu deutlich längeren Laufzeiten und damit Gesamtlaufzeiten führt. In einem ihrer Beispiele sind bei einer Gesamtzeit von 5 Minuten die meisten Mapper innerhalb von 30 Sekunden fertig. Auch beim Streaming kann diese sog. Datenschiefe (engl. data skew) den Durchsatz eines Clusters signifikant mindern. Einen weiteren Ansatz die Laufzeit für nahe Echtzeit Anwendungen zu optimieren präsentieren Whang et al. [???]. An-

statt eine Ergebnismenge nach Abschluss eines Algorithmus zu liefern, zeigen Sie Möglichkeiten partielle Ergebnisse während der Laufzeit des Algorithmus zu erhalten.

4 Zielsetzung

Im Rahmen der Thesis soll ein Entity Resolution Framework für Datensatzströme entstehen. Als Basis soll ein (Event) Stream Processing Framework genutzt werden. Das Framework soll eine Reihe von Matchern, sowie Kombinationsfunktionen der Matcher unterstützen. Hauptaugenmerk ist jedoch die Skalierbarkeit. Gelöst werden soll das Data Skew Problem bei verschiedenen Blocking Strategien. Eine weitere Schwierigkeit ist, dass die Datenmenge nicht statisch ist, sondern neue Datensätze jederzeit hinzukommen können. Beim Erweitern des Suchraums soll beachtet werden, dass kein Data Skew auftritt. Dadurch soll vermieden werden, dass der Durchsatz innerhalb des Clusters signifikant sinkt. Idealerweise soll der Durchsatz, sowie die Qualität der Suchergebnisse, mit bereits bekannten Veröffentlichungen verglichen werden. Das Framework soll dabei kein Domainwissen eines bestimmten Entitätstypen berücksichtigen.

5 Methoden

Zur Umsetzung der in Abschnitt 4 beschriebenen Ziele muss zunächst eine Wissensbasis durch Literaturarbeit in folgenden Grundlagen geschaffen werden:

- Algorithmen zur Entity Resolution
- Blocking und Indexing Strategien für Entity Resolution
- Data Skew bei verteilten und parallelen Anwendungen
- Entity Resolution Frameworks - traditionell, MapReduce, Streaming
- (Event) Streaming Frameworks

Weitere Methoden sind:

- UML-Entwurf
- Proof of Concept
- Funktionelle Leistungsbewertung anhand von Datensätzen in wissenschaftlichen Publikationen

6 Erwartete Ergebnisse

Die erwarteten Ergebnisse der Masterarbeit sind:

- Analyse von Entity Resolution Algorithmen
- Analyse von Entity Resolution Frameworks
- Analyse von (Event) Stream Processing Frameworks, für gegebenen Anwendungsfall
- Design eines Entity Resolution Streaming Framework
- Prototyp der wesentlichen Funktionen
- Evaluation des Prototypen, gegen öffentliche Datensätze existierender Veröffentlichungen

7 Vorbedingungen

- Datensätze zum Evaluieren und Trainieren des Frameworks bzw. der Algorithmen

Abbildung 7.1: My Logo

Tabelle 7.1: My Table

a	b	c
1	2	3
4	5	6

Auflistung 7.1 Listing caption

```
main :: IO ()
main = putStrLn "Hello World!"
```

Literaturverzeichnis

[1] Altwaijry, Hotham ; Mehrotra, Sharad ; Kalashnikov, Dmitri V.: QuERy: A Framework for Integrating Entity Resolution with Query Processing. In: Proc. VLDB Endow. Bd. 9 (2015), Nr. 3, S. 120–131

Abbildungsverzeichnis

Auflistungsverzeichnis

Tabellenverzeichnis

Erklärung

Erklärung gem. ABPO, Ziff. 6.4.3

Ich versichere, dass ich die Master-Thesis selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe.

Wiesbaden, 19.10.16

Kevin Sapper

Hiermit erkläre ich mein Einverständnis mit den im Folgenden aufgeführten Verbreitungsformen dieser Master-Thesis:

Verbreitungsform	ja	nein
Einstellung der Arbeit in die Bibliothek der Hochschule RheinMain	√	
Veröffentlichung des Titels der Arbeit im Internet	√	
Veröffentlichung der Arbeit im Internet	√	

Wiesbaden, 19.10.16

Kevin Sapper