Maschinelles Lernen expressiver Verknüpfungsregeln zur Duplikaterkennung mittels genetischer Programmierung*

Robert Isele
mail@robertisele.com

Abstract: Die vorliegende Arbeit führt neuartige Algorithmen für ein zentrales Problem der Datenintegration und Datenbereinigung ein, das Finden von Paaren von Entitäten in Datensets, welche das gleiche Realwelt-Objekt beschreiben. Sie deckt dabei den gesamten Entity-Matching-Prozess ab, von der automatischen Erzeugung von effektiven Verknüpfungsregeln mittels genetischen Algorithmen, bis zu deren effizienten Ausführung. Die eingeführten Algorithmen tragen dazu bei den Konfigurationsaufwand des Verknüpfungsprozesses zu reduzieren, die Genauigkeit der generierten Links zu erhöhen, sowie schließlich die Ausführung auf Einzelmaschinen oder Rechenclustern zu beschleuningen.

1 Einführung

Ein zentrales Problem der Datenintegration und Datenbereinigung ist das Finden von Paaren von Entitäten in Datensets, welche das gleiche Realwelt-Objekt beschreiben. Viele bestehende Methoden für das Identifizieren solcher Paare basieren auf domänenspezifischen Verknüpfungsregeln, die festlegen wie zwei Entitäten auf Äquivalenz verglichen werden. Allerdings ist das Schreiben solcher Verknüpfungsregeln von Hand in der Praxis aufwendig und erfordert eine detaillierte Kenntnis der verwendeten Datensets. Ein weiteres wichtiges Problem stellt das effiziente Ausführen der generierten Verknüpfungsregeln dar. Abbildung 1 zeigt die wichtigsten Schritte eines regelbasierten Prozesses zur Duplikaterkennung. Im Folgenden gehen wir für jeden Schritt näher ins Detail.

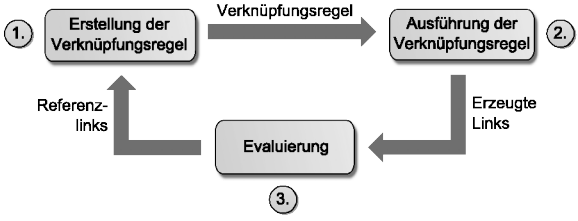

Abbildung 1: Duplikaterkennung mit Verknüpfungsregeln.

*Englischer Titel der Dissertation: "Learning Expressive Linkage Rules for Entity Matching using Genetic Programming"

1.1 Erstellung der Verküpfungsregel

Bevor zwei Datenquellen verlinkt werden können, muss eine Verknüpfungsregel definiert werden, welche die Voraussetzungen angibt, die gelten müssen damit zwei Entitäten verlinkt werden. Verknüpfungsregeln können entweder von einem Domänenexperten oder einem überwachten Lernalgorithmus erstellt werden. Das Erstellen einer effektiven Verknüpfungsregel von Hand ist ein nicht-triviales Problem und verlangt vom Autor detaillierte Kenntnisse über die Struktur der zu verlinkenden Datensätze. Wir veranschaulichen dies am einfachen Beispiel zweier Datensätze über Filme. Zunächst ist ein Vergleich der Filme alleinig basierend auf den Filmtiteln in der Regel nicht ausreichend, da Filme mit dem gleichen Titel tatsächlich unterschiedliche Filme bezeichnen können, die in verschiedenen Jahren veröffentlicht wurden. Daher muss die Verknüpfungsregel zumindest die Filmtitel sowie ihre Veröffentlichungstermine kombinieren und dabei eine geeignete Aggregationsfunktion wählen, welche die Ähnlichkeiten beider Eigenschaften vereint. In der Regel muss mit Unregelmässigkeiten in beiden Datensätzen gerechnet werden muss. So können die Veröffentlichungsdaten des gleichen Films in verschiedenen Datenquellen um mehrere Tage abweichen. Deshalb muss der Autor der Verknüpfungsregel auch geeigneten Distanzmaße zusammen mit adäquaten Schwellenwerten auswählen. Die Abdeckung aller in den Datenquellen vorhandener Heterogenitäten durch die Verknüpfungsregel ist oft schwierig und kann nur durch mehrere Iterationen erreicht werden.

1.2 Ausführung der Verknüpfungsregel

Die Ausführung einer Verknüpfungsregel hat das Ziel alle Paare von Entitäten zu identifizieren, für welche die gegebene Regel erfüllt ist. Das Ergebnis ist eine Menge von Links, worin jeder Link zwei Entitäten verbindet, welche laut Verknüpfungsregel ähnlich sind.

Abbildung 2 zeigt einen typischen Ausführungsprozess für zwei Datenquellen. Um zu

Abbildung 2: Ausführung einer Verknüpfungsregel.

Vermeiden, dass die Verknüpfungsregel für jedes mögliche Paar von Entitäten ausgewertet werden muss, kann zunächst eine Indizierung durchgeführt werden. Das Ziel der Indizierung ist es Paare von Entitäten herauszufiltern welche potentiell ähnlich sind um dabei andererseits Paare die definitiv verschieden sind früh verwerfen zu können. Basierend auf dem Index wird die Verknüpfungsregel nur für die potentiell ähnlichen Paare von Entitäten ausgewertet.

Verschiedene Indizierungsverfahren wurden entwickelt um die Effizienz des Prozesses zu

erhöhen [Chr12]. Allerdings können viele dieser Verfahren dazu führen, dass korrekte Links durch das Indizierungsverfahren fälschlicherweise verworfen werden und damit weniger Links generiert werden als durch die Verknüpfungsregel vorgegeben.

Im vorigen Beispiel zweier Filmdatenbanken, könnte eine einfache Indizierungsmethode so aussehen, dass jedem Film basierend auf seinem Erscheinungsjahr ein Index zugewiesen wird. In diesem Fall müssten anschließen lediglich Filme mit dem gleichen Erscheinungsjahr auf Ähnlichkeit mit der Verknüpfungsregel getestet werden. Der Nachteil dieser Methode wäre allerdings der Verlust von Links zwischen Filmen, für welche das falsche Erscheinungsjahr angegeben ist. Die Entwicklung von Indizierungsverfahren, welche die Anzahl der nötigen Vergleiche reduzieren und zugleich den Verlust korrekter Links vermeiden, ist ein wichtiges Forschungsgebiet.

1.3 Evaluierung

Der Zweck des Evaluierungsschritts ist es die Qualität der Verknüpfungsregel zu bestimmen und potentielle Fehler in den generierten Links zu finden. In der Regel erfolgt die Überprüfung basierend auf manuell erstellten Referenzlinks. Eine Menge von Referenzlinks besteht hierbei aus positiven Referenzlinks, welche Paare von Entitäten verknüpfen welche nachgeprüft das gleiche Real-Welt Objekt beschreiben, sowie negativen Referenzlinks, welche Paare von Entitäten verknüpfen, welche verschiedene Objekte beschreiben.

Nach der Ausführung der Verknüpfungsregel, können mit Hilfe der Referenzlinks zwei Arten von fehlerhaften Links identifiziert werden: Falsch positive Links, d.h. für einen negativen Referenzlink wurde basierend auf der aktuellen Verknüpfungsregel ein Link erstellt und falsch negative Links, d.h. positive Referenzlinks für die kein Link generiert wurde. Die gefundenen Fehler können genutzt werden um die Verknüpfungsregel zu verbessern. Auf diese Weise kann die Verknüpfungsregel iterativ verbessert werden.

1.4 Vorliegende Arbeit

Die vorliegende Arbeit führt neuartige Algorithmen ein, die den gesamten Prozess der Duplikaterkennung abdecken, von der automatischen Erzeugung von effektiven Verknüpfungsregeln, bis zu deren effizienten Ausführung. Die folgenden Abschnitte fassen die Hauptbeiträge zum aktuellen Stand der Forschung zusammen:

Abschnitt 2 stellt eine expressive Repräsentation für Verknüpfungsregeln vor, welche Regeln als Operatorbaum darstellt und expressiver als durch bestehende Lernverfahren unterstützte Repräsentationen ist. In *Abschnitt 3* wird zunächst ein genetischer Algorithmus vorgestellt, der Verknüpfungsregeln aus einem Goldstandard lernt, welcher aus einer Menge von Paaren von Entitäten besteht, worin jedes Paar als Duplikat oder Nicht-Duplikat gekennzeichnet ist.

Um auch Anwendungsfälle abzudecken, in denen kein Goldstandard verfügbar ist, wird

anschließend in *Abschnitt 4* ein komplementärer Algorithmus eingeführt, welcher Verknüpfungsregeln interaktiv lernt indem er dem Nutzer eine kleine Anzahl von Beispielpaaren präsentiert, welcher dieser als Duplikat oder Nicht-Duplikat kennzeichnet. Abschließend führt *Abschnitt 5* ein neuartiges Indizierungsverfahren ein, welches die Anzahl der notwendigen Vergleiche signifikant reduziert und die Ausführung der gelernten Verknüpfungsregeln auf verteilten Architekturen ermöglicht.

2 Expressive Verknüpfungsregeln

Die Aufgabe einer *Verknüpfungsregel* ist es, einem gegebenen Paar von Entitäten einen Ähnlichkeitswert zuzuweisen. Um den Ähnlichkeitswert zu bestimmen, verbindet eine Verknüpfungsregel einzelne Vergleiche. Im Laufe der Zeit wurden verschiedene Modelle vorgeschlagen, die das Ziel haben die einzelnen Distanzen zu einem Ähnlichkeitswert zu kombinieren. Die meisten Modelle basieren auf zwei Typen von Klassifikatoren:

- Ein *linearer Klassifikator* kombiniert mehrere Distanzen, indem die gewichtete Summe der einzelnen Distanzen berechnet wird.Die Größe der Gewichte reguliert hierbei den Einfluss eines bestimmten Vergleichs auf den Distanzwert.
- Ein *schwellwertbasierter Klassifikator* kombiniert verschiedene Ähnlichkeitstests mit logischen Operatoren.Ein Ähnlichkeitstest besteht dabei aus einem Distanzmaß und einem Schwellwert.

In der vorliegenden Arbeit wird ein Modell für die Repräsentation von Verknüpfungsregeln vorgeschlagen welches expressiver ist als bisher verwendete Modelle und Datentransformationen enthalten kann, welche Werte vor dem Vergleich normalisieren. Das vorgeschlagene Modell repräsentiert eine Verknüpfungsregel als einen Baum der aus vier Typen von Operatoren aufgebaut ist:

Ein **Eingabeoperator** gibt alle Werte eines bestimmten Attributs, beispielsweise die Adresse einer Person, zurück. Ein **Transformationsoperator** transformiert Werte basierend auf eine vorgegebenen Transformationsfunktion.Mehrere Transformationsoperatoren können verschachtelt werden um Ketten von Transformationen abzubilden. Ein **Vergleichsoperator** vergleicht zwei Werte mit Hilfe eines vorgegebenen Distanzmaßes und einem Distanzschwellwert und gibt einen Ähnlichkeitswert zurück. Ein **Aggregationsoperator** vereint mehrere Ähnlichkeitswerte zu einem Ähnlichkeitswert mit Hilfe einer Aggregationsfunktion, z.B. dem gewichteten Mittelwert.

Abbildung 3 zeigt eine Verknüpfungsregel um geographische Orte zu verlinken.

3 GenLink — Überwachtes Lernen von Verknüpfungsregeln

Im vorhergehenden Abschnitt haben wir bereits gezeigt wie Verknüpfungsregeln verwendet werden können um die genauen Bedingungen anzugeben die erfüllt sein müssen damit

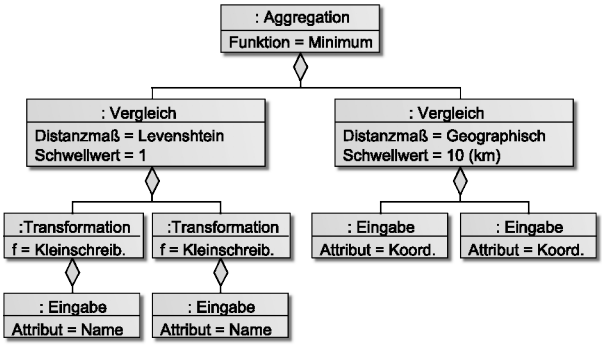

Abbildung 3: Beispiel einer Verknüpfungsregel.

zwei Entitäten verlinkt werden. Allerdings ist das Schreiben und die Feinoptimierung solcher Verknüpfungsregeln von Hand oft schwierig und zeitaufwendig.

Eine Möglichkeit diesen Aufwand zu reduzieren, stellen Lernverfahren dar, welche Verknüpfungsregeln aus bestehenden Referenzlinks generieren. Das Erstellen von Referenzlinks ist viel einfacher als das Schreiben von Verknüpfungsregeln, da es keine Vorkenntnisse über verschiedene Distanzmaße oder das durch das vorliegende System verwendete Modell für die Repräsentation von Verknüpfungsregeln erfordert. Referenzlinks können von Domain-Experten erstellt werden, indem sie die Gleichwertigkeit von einer Menge von Paaren von Entitäten aus den Datensätzen bestätigen oder ablehnen.

3.1 Der Algorithmus

Der GenLink-Algorithmus basiert auf genetischer Programmierung. Genetische Programmierung ist eine Erweiterung des genetischen Algorithmus, welche ursprünglich von Cramer vorgeschlagen wurde. Genetische Programmierung eignet sich für Probleme deren Lösungen als Bäume von Operatoren dargestellt werden können, so wie es bei der eingeführten Repräsentation für Verknüpfungsregeln der Fall ist.

Abbildung 4 veranschaulicht den allgemeinen Ablauf des GenLink-Algorithmus. Zu Be-

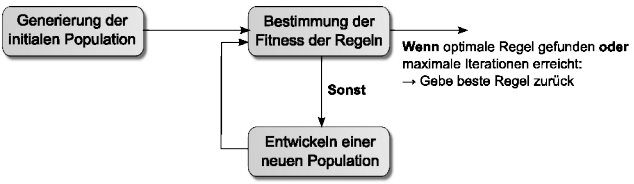

Abbildung 4: GenLink Iterationen.

ginn generiert GenLink eine Population von initialen Verknüpfungsregeln, welche nach einem Zufallsalgorithmus erstellt werden. In jeder Iteration überprüft GenLink die Fitness aller Verknüpfungsregeln in der aktuellen Population indem für jede Regel überprüft wird wie viele der aktuellen Referenzlinks korrekt bestimmt werden. Anschließen erzeugt GenLink neue Population indem basierend auf einer Selektionsstrategie, welche Regeln mit höherer Fitness bevorzugt, Regeln aus der Population ausgewählt werden und zu einer neuen Regel kombiniert werden. Dazu werden zwei Hauptoperationen eingesetzt: 1. Mutation modifiziert eine einzelne Regel zufällig. 2. Crossover rekombiniert zwei Regeln zu einer neuen Regel. Der Algorithmus ist beendet sobald entweder eine optimale Verknüpfungsregel gefunden wurde oder eine vorgegebene maximale Anzahl von Iterationen erreicht wurde.

GenLink erweitert die traditionelle genetische Programmierung in drei wichtigen Punkten:

1. Während bei genetischen Algorithmen die initiale Population in der Regel vollständig zufällig generiert wird, verwendet GenLink ein Verfahren, welches eine initiale Population erstellt, die nur Regeln enthält die Eigenschaften mit ähnlichen Werten vergleichen. Somit werden weniger Iterationen gebraucht um Regeln zu lernen.

2. Die meisten Algorithmen der genetischen Programmierung im Allgemeinen und das einzige GenLink vorrausgehende überwachte Lernverfahren von Carvalho et al. [dCLGdS12] verwenden einen generischen Crossover-Operator der Subtree-Crossover genannt wird. GenLink dagegen setzt eine Menge von spezialisierten Crossover-Operatoren ein, worin jeder Operator nur einen Aspekt der Verknüpfungsregeln kombiniert. Beispielweise ist ein Crossover-Operator enthalten welcher Transformationen beider Regeln kombiniert und damit Transformationsketten aufbauen kann.

3. Ein häufiges Problem von Algorithmen der genetischen Programmierung ist das Aufblähen der gelernten Lösungen, d.h. das Entwickeln von Lösungsbäumen, die zunehmend grösser werden, ohne dass deren Qualität im gleichem Maße zunimmt. GenLink verwendet ein neuartiges Verfahren um ein Aufblähen der Lösungen zu verhindern, ohne dass die Qualität der gelernten Verknüpfungsregeln leidet.

3.2 Ergebnisse

3.2.1 Evaluierung der Erweiterungen

Der vorgestellte GenLink-Algorithmus enthält Erweiterungen des verbreiteten Algorithmus für genetische Programmierung. Der Betrag jeder der eingeführten Erweiterungen wurde experimentell auf sechs Datensets aus drei verschiedenen Gebieten evaluiert:

(1) Die eingesetzte Strategie zur Generierung der initialen Population erhöhte die Qualität der Verknüpfungsregeln auf Datensets mit vielen Attributen. **(2)** Die spezialisierten Crossover-Operatoren erzielten auf allen sechs Datensets eine vergleichbare oder höhere Performance als traditionelles Subtree-Crossover. **(3)** Die Effektivität des eingesetzten Verfahren um ein Aufblähen der Regeln zu verhindern konnte ebenfalls gezeigt werden.

3.2.2 Expressivität der gelernten Verknüpfungsregeln

Die eingesetzte, expressive Repräsentation der Verknüpfungsregeln konnte die Performance der gelernten Verknüpfungsregeln auf allen sechs Datensets erhöhen. Dafür zeigte sich vorwiegend die Einbeziehung von Datentransformationen in die Regeln verantwortlich.

3.2.3 Vergleich mit bestehenden Lernverfahren

Zwei Datensets haben sich dadurch, dass sie bereits für die Evaluierung von mehreren existierenden Lernalgorithmen verwendet wurden, als Benchmarkdatensets etabliert: Das *Cora* Datenset [MNRS00] enthält Zitierungen für Publikationen aus der Cora Computer Science Datenbank für Forschungspublikationen. Das *Restaurant* Datenset [TKM01] enthält Einträge aus dem *Fodor's and Zagat's Restaurant Guide*.

Tabelle 1 vergleicht die Qualität der mit unterschiedlichen Verfahren gelernten Verknüpfungsregeln mit GenLink. GenLink konnte auf beiden Datensets eine höhere Performance erzie-

Ansatz	Cora	Restaurant
Lineare und schwellwertbasierte Klassifikatoren		
Active Atlas [TKM01]	(not available)	99.7% accuracy
MARLIN [BM03]	86.7% F-measure	92.2% F-measure
Genetische Programmierung		
[dCLGdS12]	91.0% F-measure	98.0% F-measure
GenLink	96.6% F-measure	99.3% F-measure
Kollektive Ansätze		
[Dom04]	87.0% F-measure	(not available)
[DHM05]	95.4% F-measure	(not available)
Unüberwachte Ansätze (Top 3)		
RiMOM [WZH+10]	(not available)	81% F-measure
LN2R [SNPR10]	(not available)	75% F-measure
ObjectCoref [HCCQ10]	(not available)	73% F-measure

Tabelle 1: Vergleich verschiedener Ansätze mit GenLink.

len als der bestehende genetische Algorithmus von Carvalho et al. [dCLGdS12]. Darüber hinaus konnte GenLink eine Reihe von bestehenden Lernalgorithmen die unterschiedliche Verfahren benutzen übertreffen.

4 ActiveGenLink — Interaktives Lernen von Verknüpfungsregeln

Der ActiveGenLink-Algorithmus kombiniert aktives Lernen (engl. active learning) mit genetischer Programmierung um Verknüpfungsregeln interaktiv zu lernen. Die für das Lernen der Verknüpfungsregel notwendigen Referenzlinks werden in Interaktion mit einem menschlichen Experten erstellt. Abbildung 5 illustriert die drei Hauptschritte, welche iterativ ausgeführt werden.

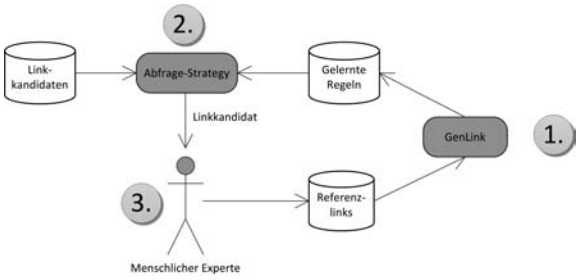

Abbildung 5: Ablauf des ActiveGenLink-Algorithmus.

(1) Die Abfrage-Strategie (engl. query strategy) wählt einen Linkkandidaten aus, für welchen die Verknüpfungsregeln in der gelernten Population unsicher sind. **(2)** Ein menschlicher Experte markiert den ausgewählten Linkkandidaten als korrekt oder inkorrekt woraus ein neuer Referenzlink generiert wird. **(3)** Der GenLink-Algorithmus lernt eine Population von Verknüpfungsregeln aus den aktuellen Referenzlinks.

In der Evaluierung konnten drei Eigenschaften von ActiveGenLink gezeigt werden:

- Auf allen sechs Datenset erreichte ActiveGenLink, nach dem der Nutzer maximal 50 Kandidaten manuell gekennzeichnet hat, eine vergleichbare Genauigkeit als der überwachte GenLink Algorithmus auf dem gesamten Goldstandard, reduzierte also die Anzahl der Linkkandidaten die manuell überprüft werden müssen erheblich.
- Auf einem großen geografischen Datenset konnte die Skalierbarkeit von ActiveGenLink gezeigt werden. Die zwei zu verlinkenden Datensets enthielten 323.257 und 560.123 Einträge und ergeben damit über 180 Milliarden potentielle Linkkandidaten, welche durch die Abfrage-Strategie selektiert werden können. Im Durschnitt dauerte jede Iteration etwa 1,5 Sekunden, wobei jede Iteration das Lernen einer neuen Verknüpfungsregel aus den aktuellen Referenzlinks, sowie das Selektieren eines neuen Linkkandidaten durch die Abfrage-Strategie umfasst.
- Die für ActiveGenLink entwickelte, neuartige Abfrage-Strategie benötigt weniger Referenzlinks als bestehende Strategien.

5 MultiBlock — Effizientes Ausführen von Verknüpfungsregeln

Die Grundidee des MultiBlock-Indizierungsalgorithmus ist das Generieren eines mehrdimensionalen Indexes in welchem die Distanzen der Entitäten zueinander beibehalten werden, d.h. laut Verknüpfungsregel ähnliche Entitäten im Index nahe beieinander liegen. In der Evaluierung konnten drei Haupteigenschaften von MultiBlock gezeigt werden:

Effizienz: Auf einem Datenset mit Personendaten verglichen wir die Leistung von Multiblock mit vier etablierten Indizierungsverfahren. Multiblock war die einzige Me-

thode, die eine Trefferquote von 100% erreichte. Unter den Verfahren mit einer Trefferquote von mindestens 50%, erreichte Multiblock die niedrigste Laufzeit. Auf einem großen geographischen Datenset verglichen wir die Leistung von Multiblock mit dem StringMap-Verfahren, welches ebenfalls einen mehrdimensionalen Index verwendet. Obwohl StringMap eine größere Reduzierung der Anzahl der notwendigen Vergleiche erreichen konnte, dauerte das Generieren des mehrdimensionalen Indexes damit etwa 10-mal länger als der gesamte Prozess mit MultiBlock veranschlagte. Multiblock reduziert die Anzahl der Vergleiche mit einem Faktor von 25.000 und war etwa 800-mal schneller als die vollständige Auswertung aller Paare.

Effektivität: Selbst mit komplexen Verknüpfungsregeln konnte MultiBlock die Anzahl der notwendigen Vergleiche stark reduzieren, ohne korrekte Links auszulassen.

Skalierbarkeit: Es wurde gezeigt, dass MultiBlock mit Hilfe von MapReduce auf einem Rechencluster ausgeführt werden kann. Dazu wurde MultiBlock auf einem Cluster mit 32 Maschinen für ein sehr großes Datenset ausgeführt, wodurch 1 Million Links in 15 Minuten generiert wurden.

6 Fazit

Die vorgestellte Arbeit deckt den vollständigen Prozess der Duplikaterkennung, von der Generierung der Verknüpfungsregel bis zu ihrer Ausführung, ab. Die vorgestellten Verfahren erbringen drei wichtige Beiträge zum aktuellen Stand der Forschung:

Genauigkeit: Der vorgestellte GenLink-Algorithmus verwendet genetische Programmierung zusammen mit einer expressiven Repräsentation für Verknüpfungsregeln um Datenquellen mit *höherer Genauigkeit* zu Verlinken als bisherige Verfahren.

Aufwand: Der vorgestellte ActiveGenLink-Algorithmus *reduziert den Aufwand* der für die Verlinkung zweier Datenquellen notwendig ist, indem er Verknüpfungsregeln interaktiv generiert, während ein menschlicher Experte eine Reihe von Linkkandidaten überprüft.

Effizienz: Das MultiBlock Indizierungsverfahren ermöglicht die *effiziente Ausführung* der gelernten Regeln und schließt somit den Duplikaterkennungsprozess ab. MultiBlock ist effizienter als verglichene Indizierungsverfahren und kann auf einem Rechencluster ausgeführt werden.

Die vorgestellten Algorithmen wurden im *Silk Link Discovery Framework*[1] implementiert, wodurch ein System zur Duplikaterkennung zur Verfügung gestellt wird, welches den vollständigen Duplikaterkennungsprozess unterstützt.

[1] `http://silk.wbsg.de/`

Literatur

[BM03] M. Bilenko und R.J. Mooney. Adaptive Duplicate Detection Using Learnable String Similarity Measures. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seiten 39–48, 2003.

[Chr12] Peter Christen. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Springer, 2012.

[dCLGdS12] Moisés G de Carvalho, Alberto HF Laender, Marcos André Gonçalves und Altigran S da Silva. A Genetic Programming Approach to Record Deduplication. *IEEE Transactions on Knowledge and Data Engineering*, 24(3):399–412, 2012.

[DHM05] Xin Dong, Alon Halevy und Jayant Madhavan. Reference reconciliation in complex information spaces. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Seiten 85–96, 2005.

[Dom04] Pedro Domingos. Multi-relational record linkage. In *In Proceedings of the KDD Workshop on Multi-Relational Data Mining*, Seiten 31–48, 2004.

[HCCQ10] Wei Hu, Jianfeng Chen, Gong Cheng und Yuzhong Qu. ObjectCoref & Falcon-AO: Results for OAEI 2010. In *Proceedings of the Fifth International Workshop on Ontology Matching (OM)*, Seiten 158–165, 2010.

[MNRS00] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie und Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000. First mention of the Cora dataset.

[SNPR10] Fatiha Saıs, Nobal Niraula, Nathalie Pernelle und Marie-Christine Rousset. LN2R–a knowledge based reference reconciliation system: OAEI 2010 Results. In *Proceedings of the Fifth International Workshop on Ontology Matching (OM)*, Seiten 172–180, 2010.

[TKM01] Sheila Tejada, Craig A. Knoblock und Steven Minton. Learning object identification rules for information integration. *Information Systems*, 26(8):607–633, 2001.

[WZH+10] Zhichun Wang, Xiao Zhang, Lei Hou, Yue Zhao, Juanzi Li, Yu Qi und Jie Tang. RiMOM results for OAEI 2010. In *Proceedings of the Fifth International Workshop on Ontology Matching (OM)*, Seiten 194–201, 2010.

Robert Isele, geboren am 27. Juni 1983 in Waldshut-Tiengen, wohnhaft in Berlin, schloss 2008 das Studium der Informatik an der Technischen Universität München mit dem Diplom mit der Gesamtnote 1,2 ab, worauf er zunächst als Softwareentwickler bei der REC GmbH angestellt war.

Zwischen Oktober 2009 und März 2013 arbeitete er als wissenschaftlicher Mitarbeiter, zunächst bei der Freien Universität Berlin und ab September 2012 an der Universität Mannheim, wo er 2013 seine Promotion mit dem Titel Dr. rer. nat. abschloss. Das Thema der Arbeit lautete “Learning Expressive Linkage Rules for Entity Matching using Genetic Programming” und wurde mit der Gesamtnote “summa cum laude” bewertet. Während seiner Promotion absolvierte er unter anderem einen Forschungsaufenthalt am Yahoo! Research Lab in Barcelona. Seit dem April 2013 ist Robert Isele bei der Brox IT-Solutions GmbH als Consultant angestellt.