SABINE MASSMANN

Ontologie-Matching von Produktkatalogen

Sowohl im E-Commerce als auch in der Forschung stellt das Matching von Produktkatalogen ein wichtiges Problem dar. Heterogenitäten, Redundanz und mehrfach zugeordnete Instanzen erschweren dabei das Matchproblem und werden in diesem Beitrag näher betrachtet. Zur Lösung des allgemeinen Matchproblems wurden zahlreiche Techniken entwickelt, die Metadaten, Instanzen und auch Zusatzinformation wie Thesauri verwenden. In diesem Artikel wird eine Auswahl von vier Lösungsstrategien in Bezug auf das Matching von Produktkatalogen untersucht. Zusätzlich werden Ergebnisse des OAEI-Directory-Tests präsentiert, bei dem es galt, verschiedene hierarchische Klassifikationen in Form von Webverzeichnissen zu matchen. Die Ergebnisse geben einen Anhaltspunkt, wie gut gegenwärtig Matchsysteme in der Praxis abschneiden.

1 Einleitung

Ontologien spielen eine immer wichtigere Rolle sowohl in der Geschäftswelt als auch in der Forschung. Objekte dieser Domänen, wie z.B. Produkte oder Jobangebote, können mithilfe von Ontologien semantisch beschrieben und anhand von bestimmten Merkmalen Klassen, z.B. Kategorien, zugeordnet werden.

Produktkataloge sind Ontologien, die in meist hierarchisch angeordneten Kategorien Informationen zu den zugeordneten Produkten enthalten. Produkte, die einer Unterkategorie zugeordnet sind, erfüllen dabei auch die Merkmale bzw. Eigenschaften der übergeordneten Kategorien. Beziehungen zwischen den Kategorien, wie dies zwischen Klassen einer Ontologie möglich ist, gibt es bei Produktkatalogen nicht.

In Abbildung 1 ist ein Ausschnitt der Amazon-Produktontologie für Filme dargestellt. Filme werden in dieser Ontologie z.B. in die Kategorien *Genres* und *Regisseure* eingeteilt. Die Filme lassen sich durch eine Suche über Kategorien, z.B. *Genre=Western*, oder durch »Browsen« innerhalb der Ontologie identifizieren.

Für den Kunden bieten Produktkataloge große Vorteile, da sie z.B. die Suche nach bestimmen Produkten und den Produktvergleich erleichtern. Unternehmen setzen Produktkataloge in unternehmensübergreifenden Prozessen ein, wie z.B. für die Einkaufs- und Verkaufsorganisation, die immer häufiger über internetbasierte Anwendungen abgewickelt werden.

Verschiedene Produktkataloge unterscheiden sich gewöhnlich im Aufbau und in der Verwendung von Bezeichnungen. Dies ist oft selbst dann der Fall, wenn sie für denselben Zweck entworfen wurden. Für den Datentransfer und als ersten Schritt zur Integration werden daher *Mappings* benötigt, die semantisch korrespondierende Konzepte (hier Kategorien) zwischen unterschiedlichen Katalogen verknüpfen [Schulten et al. 2001].

Das (semi)automatische Erstellen von Mappings zwischen Ontologien – Matching genannt – ist ein Problem, das hauptsächlich in der Forschung z.B. von [Rahm & Bernstein 2001] und [Shvaiko & Euzenat 2005] betrachtet wird. Je nachdem, welche Art von Ontologien gematcht werden soll, z.B. Kataloge oder Thesauri, treten unterschiedliche Schwierigkeiten auf.

In diesem Beitrag wird das Matching von Produktkatalogen betrachtet. Dabei werden zuerst in Abschnitt 2 die Herausforderungen beim Matching von Produktkatalogen diskutiert. Eine Auswahl an aktuellen Matchansätzen wird in Abschnitt 3 präsentiert. Ergebnisse aus einer Evaluierungsinitiative werden in Abschnitt 4 vorgestellt. Zum Abschluss wird ein Fazit in Abschnitt 5 gezogen.

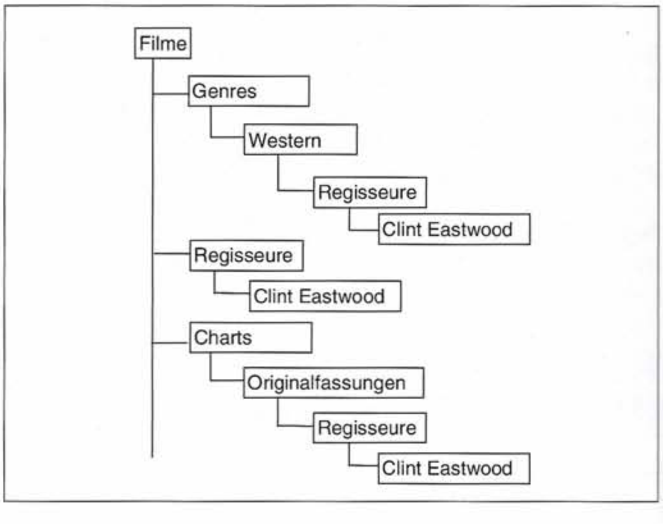
2 Herausforderungen

Zum automatischen Generieren von Mappings werden Ontologie-Matching-Techniken eingesetzt. Diese verwenden entweder Metadaten, z.B. Kategorienamen, oder Instanzen, z.B. Produkte. Beim Matching von Produktkatalogen gilt es, mehrere Herausforderungen zu meistern, die zum Teil durch spezielle Eigenschaften der Kataloge entstehen: Heterogenitäten, Redundanz und Instanzen.

2.1 Heterogenitäten

Die Produktkataloge von Amazon und Softunity in Abbildung 2 machen deutlich, dass Produktkataloge selbst dann große Unterschiede aufweisen können, wenn sie für dieselbe Anwendung entworfen wurden. Dabei lassen sich verschiedene Arten von Heterogenität unterscheiden (Auswahl aus [Euzenat & Shvaiko 2007]).

Abb. 1: Ausschnitt aus der Amazon-Produktontologie



Von *terminologischer Heterogenität* spricht man, wenn gleiche Kategorien verschiedene Namen besitzen (Synonyme, z.B. Autor und Schriftsteller) oder umgekehrt gleiche Bezeichnungen verschiedene Klassen verkörpern (Homonyme, z.B. Bank als Sitzgelegenheit oder als Kreditinstitut). Auch die Verwendung von Abkürzungen (z.B. SciFi steht für Science-Fiction) oder Bezeichnungen in unterschiedlichen Sprachen (Multilingualität) führen zu Unterschieden in der Bezeichnung. Diese terminologischen Unterschiede sind ein häufiges Problem, da Bezeichnungen oft branchen- und unternehmensabhängig sind, z.B. durch die verwendete Sprache, durch Abkürzungen und technische Begriffe.

Neben der terminologischen existiert das Problem der *konzeptuellen Heterogenität*. Konzeptuelle Heterogenität entsteht hauptsächlich durch:

- Unterschiede in der Abdeckung: Kataloge decken meist einen bestimmten Bereich an Instanzen ab. Dieser kann sehr spezifisch sein (z.B. Comic.de nur Comics) oder auch vielfältig (z.B. Amazon.de mit u.a. Literatur, Musik, Filmen und Elektronik).
- Unterschiede in der Granularität: Bei Katalogen kann die Einteilung der Instanzen in unterschiedlich detaillierte Kategorien erfolgen. In der Amazon-Produktontologie der Abbildung 2 wird zwischen *Windows* und *Linux* als Betriebssystem unterschieden. Softunity dagegen hat nur die Kategorie *Betriebssysteme*.
- Unterschiede in der Perspektive: Diese Heterogenität tritt auf, wenn zwei Ontologien denselben Teil der realen Welt mit demselben Detailniveau abbilden, aber von unterschiedlichen Perspektiven aus, z.B. Produkte nach Preisniveau oder nach Verkaufsrang.

2.2 Redundanz

Kataloge ermöglichen die schnelle und einfache Suche nach Instanzen, z.B. nach einem bestimmten Film in einem Produktkatalog. Dies wird durch Redundanz unterstützt: Es gibt mehrere Wege zu einer Instanz bzw. umgekehrt betrachtet, sind einer Instanz mehrere Kategorien zugeordnet. Beispielsweise sind Filme bei der Amazon-Produktontologie (siehe Abb. 1) sowohl nach *Genres*, *Schauspieler* als auch nach *Regisseure* einsortiert.

Sind die Hierarchien vollständig überlappend, kann ein Produktkatalog in mehrere Teilontologien zerlegt werden, die dieselben Produkte nach unterschiedlichen Gesichtspunkten abbilden. Da nun mehrere Ontologien statt nur zwei miteinander gematcht werden müssen, spricht man von einem *Multiontologie-Matchproblem*.

Innerhalb einer Ontologie kann es Redundanzen bezüglich der Kategorien geben, die in unterschiedlichen Kontexten verwendet werden. So werden Kategorien wie beispielsweise *Preishits* oder *Zubehör* an mehreren Stellen in Produktkatalogen verwendet. In Abbildung 1 existiert in der Amazon-Produktontologie die Kategorie *Clint Eastwood* sowohl unter *Genres*, *Regisseure* als auch unter *Charts*.

Redundanzen erleichtern den Anwendern das Auffinden gesuchter Instanzen, erschweren jedoch beim Matching das Auffinden der korrekten Korrespondenzen. Der Kontext von Kategorien spielt daher eine wichtige Rolle.

2.3 Instanzen

Eine weitere Herausforderung liegt in der Zuordnung der Instanzen zu Kategorien und der Heterogenität im Aufbau und bei Beschreibungen von Produkten.

Während bei [Agrawal & Srikant 2001] davon ausgegangen wird, dass Instanzen nur bei den Blättern der Kataloge vorhanden sind, gehen wir davon aus, dass jeder Kategorie Instanzen zugeordnet sein können. Des Weiteren kann eine Instanz nicht nur einer, sondern mehreren Kategorien zugeordnet werden, wie z.B. in Abbildung 2 »SuSe Linux 10.1 (DVD)« zu *Novell* und *Linux*.

Produktinstanzen stellen komplexe Objekte dar und besitzen meist mehrere Attribute, z.B. Name, Hersteller und Preis. Da auch bei Instanzen terminologische Heterogenitäten auftreten

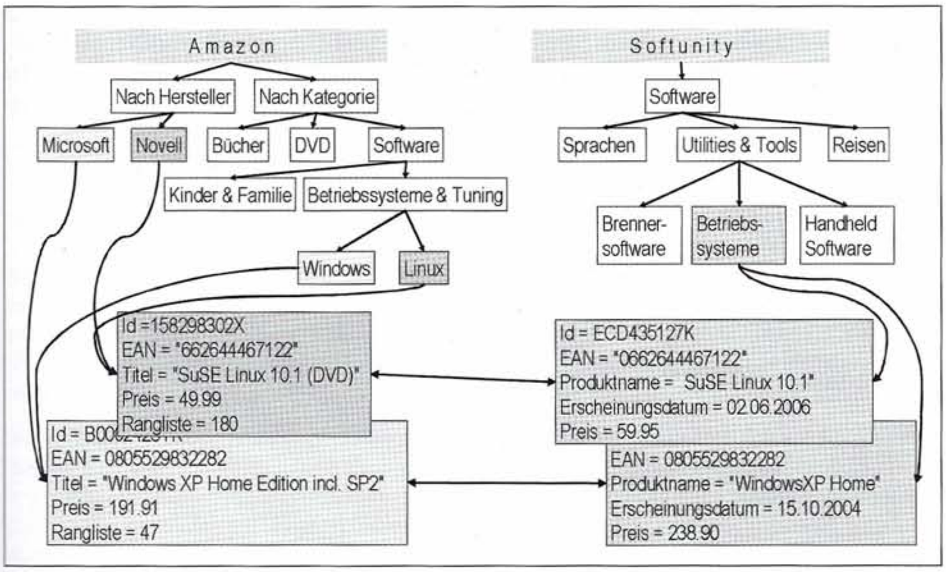


Abb. 2: Ausschnitt aus der Amazon- und der Softunity-Produktontologie mit assoziierten Produkten (basierend auf [Thor et al. 2007])

können, ist das Auffinden von Instanzduplikaten verschiedener Quellen ein eigenes Matchproblem [Elmagarmid et al. 2007]. Vereinfacht wird dies durch das Vorhandensein global eindeutiger Identifizierer wie der European Article Number (EAN), wie in Abbildung 2 gezeigt wird.

2.4 Merkmale und Auswirkungen der Heterogenität

Um die Unterschiedlichkeit der zu matchenden Kataloge einschätzen zu können, lassen sich folgende Merkmale ausnutzen:

- Statistiken der Ontologien, wie z.B. Breite, Verhältnis Breite zu Höhe und die durchschnittliche Anzahl von Unterklassen
- Art der Instanzen: einfach oder komplex
- Der Anteil der in beiden Produktkatalogen vorkommenden Instanzen
- Die Verbindung der Instanzen zu den Klassen, z.B. durchschnittliche Anzahl an Klassen, denen eine Instanz zugeordnet ist, und ob nur Blattkategorien Instanzen besitzen.

Stark voneinander abweichende Werte oder unterschiedliche Merkmale deuten auf heterogene Kataloge hin. Diese Heterogenität erschwert das Auffinden der korrekten Korrespondenzen und hat außerdem drei Konsequenzen für das zu erstellende Mapping.

Zum einen beruht die Beziehung zweier Kategorien nicht nur auf Äquivalenz. Auch weitere Beziehungsarten, z.B. Spezialisierung, sollten unterstützt werden. In [Bouquet et al. 2003] werden fünf Beziehungen vorgeschlagen: weniger allgemein, allgemeiner, äquivalent, kompatibel und inkompatibel. Die Amazon-Kategorien *Windows* und *Linux* in Abbildung 3 sind weniger allgemein als die Softunity-Kategorie *Betriebssysteme*.

Zum anderen werden komplexe Beziehungen benötigt (n:m), um Zusammenhänge von Kategorien darzustellen, die sich in beliebiger Art und Weise überlappen können [Thor et al. 2007]. In Anwendungen können diese komplexeren Korrespondenzen für z.B. sortierte Stichwortanfragen oder Produktempfehlungen verwandter Kategorien anderer Onlineshops genutzt werden.

Darüber hinaus führt die Heterogenität zweier Kataloge häufig dazu, dass ein Mapping nicht alle Kategorien der beiden Kataloge abdeckt. Einfacher ist der Spezialfall, wenn ein kleiner Katalog mit einem großen Katalog, z.B. Mediator, gematcht wird, wie dies in [Agrawal & Srikant 2001] betrachtet wird. Für diesen Beitrag gehen wir von dem allgemeinen Fall aus.

3 Bisherige Ansätze

Es existieren zahlreiche Publikationen zu den Themen Schema-Matching und Ontologie-Matching. Die Ansätze können grob in metadatenbasiert, instanzbasiert und gemischte Formen eingeteilt werden [Rahm & Bernstein 2001]. In [Euzenat & Shvaiko 2007] werden 48 Matchsysteme vorgestellt, die diesen drei Formen zugeordnet werden. Mit der Hälfte der Systeme ist das Matching von Produktkatalogen möglich, da sie Ontologien unterstützen und Mappings (als Alignments bezeichnet) berechnen.

Aus der Fülle der Möglichkeiten werden im Folgenden vier Ansätze vorgestellt und untersucht, inwieweit sie unterschiedliche Beziehungstypen und Matchkardinalitäten unterstützen. Tabelle 1 enthält eine Kurzübersicht über die vier Ansätze: CtxMatch, COMA++, GLUE und [Thor et al. 2007].

Die Auswahl soll einen Überblick über möglichst viele vorhandene Techniken – ohne Anspruch auf Vollständigkeit – geben. Ein Kriterium für die getroffene Auswahl war, dass der Ansatz entweder für das Matching von hierarchischen Klassifikationen entworfen wurde oder anhand von Katalogen evaluiert wurde. Evaluierungsergebnisse werden hier jedoch nicht aufgeführt, da die Matchaufgaben bei allen Ansätzen unterschiedlich waren und ein Vergleich der erzielten Ergebnisse somit nicht sinnvoll ist.

3.1 Metadatenbasierte Ansätze

Mehrere Ansätze verwenden Metadaten wie die Kategorienamen, Kategoriebeschreibungen und strukturelle Kontextinformation, um ein Mapping zwischen Ontologien zu bestimmen.

CtxMatch [Bouquet et al. 2003] bestimmt semantische Relationen von zwei hierarchischen Klassifikationen, wie z.B. den Produktkatalogen. CtxMatch fasst das Problem des Matchens semantischer Strukturen als Problem logischer Erfüllbarkeit auf.

Der Algorithmus besteht aus zwei Phasen. Dabei wird angenommen, dass die Klassifikationselemente durch Worte und Phrasen der natürlichen Sprachen beschrieben werden. In der ersten Phase wird die Beschreibung und implizites Wissen, das im Kontext und in der Struktur vorhanden ist, für jede Klasse in Form einer logischen Formel erstellt. Auch das Wissen über die Domäne wird in einem Satz von Formeln codiert. Hierzu wird WordNet als Quelle von lexikalischen und Hintergrundinformationen benutzt. Bei der Aufstellung der Formeln werden somit drei unterschiedliche Wis-

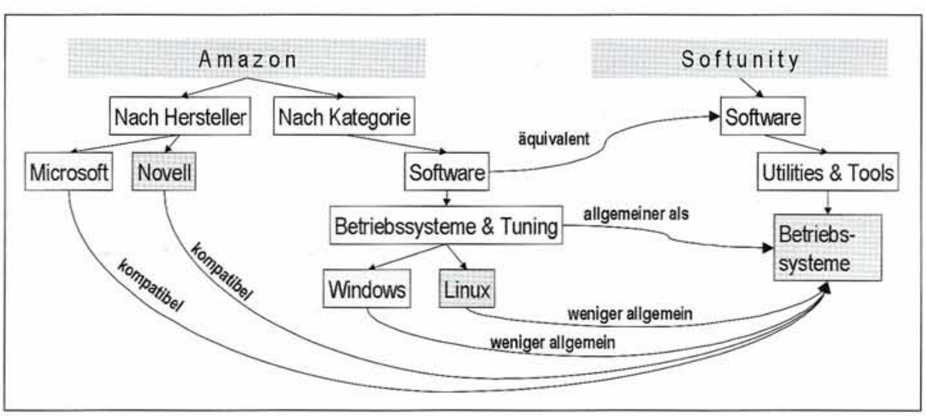


Abb. 3: Beispiele für semantische Beziehungen und Beziehungstypen

Tab. 1: Vergleich der Matchverfahren

	CtxMatch [Bouquet et al. 2003]	COMA++ [Aumueller et al. 2005]	GLUE [Doan et al. 2002]	[Thor et al. 2007]
verwendete Daten	Metadaten	hauptsächlich Metadaten, Instanzdaten	hauptsächlich Instanzdaten, Metadaten	Instanzdaten
n:m-Beziehungen	n:m-Beziehungen	n:m-Beziehungen	1:1-Beziehungen (CGLUE auch komplexe Beziehungen)	n:m-Beziehungen, Beziehungen zwischen Sets von Klassen
Beziehungstypen	weniger allgemein, allgemeiner, äquivalent, kompatibel und inkompatibel	äquivalent	äquivalent (CGLUE auch Vereinigung)	äquivalent
Evaluierung	Kataloge von Google und Yahoo	große Schemata z.B. xCBL, Webverzeichnisse	Kurskataloge, Klassifikation von Unternehmensprofilen	Produktkataloge von Amazon und Softunity

sensarten berücksichtigt: lexikalisches, domainspezifisches und strukturelles Wissen. Die Berechnung von Relationen zwischen Klassen erfolgt in der zweiten Phase, wobei dies als Problem logischer Erfüllbarkeit verstanden wird. CtxMatch unterstützt fünf verschiedene Beziehungstypen: weniger allgemein, allgemeiner, äquivalent, kompatibel und inkompatibel. Damit wird den konzeptuellen Unterschieden zweier hierarchischer Klassifikationen Rechnung getragen.

COMA++ [Aumueller et al. 2005] ist ein generisches Matchsystem und unterstützt das Matching von verschiedenen Schematypen, wie z.B. XML, relationale Schemas oder auch Ontologien. Diese werden intern in gerichtete Graphen überführt. Mittels der grafischen Benutzeroberfläche ist es dem Nutzer möglich, interaktiv in den Matchprozess einzugreifen.

Der Matchprozess besteht aus drei Phasen. Zuerst werden die relevanten Schemakomponenten, z.B. Blätter oder Pfade, identifiziert und auf diesen wird ein Matchalgorithmus ausgeführt. Ergebnisse verschiedener Matcher werden zu einem Mapping kombiniert. COMA++ ist dabei sehr flexibel, da verschiedene Matchstrategien, Matchalgorithmen und Kombinationsmöglichkeiten implementiert wurden, die konfigurierbar und beliebig kombinierbar ausführbar sind.

Eine große Anzahl der Matchalgorithmen berechnen die Stringähnlichkeit, z.B. Levenshtein und Soundex. Dahinter steckt die Annahme, je ähnlicher die Zeichenketten sind, desto ähnlicher sind es auch die dahinterstehenden Kategorien.

Um die terminologische Heterogenität abzudecken, können als Vorverarbeitungsschritt sprachenbasierte Techniken angewendet werden, die Wörter als Teil einer natürlichen Sprache verwenden. Dabei werden beispielsweise der Wortstamm und Teilwörter bestimmt. Zusätzlich ermöglicht die Nutzung von Verzeichnissen, die Synonyme und Abkürzungen enthalten, das Auffinden ebensolcher. Die Bedeutung von Homonymen kann durch die Verwendung des Kontextes, z.B. Pfadnamen, herausgefunden werden.

Weitere Matchalgorithmen verwenden die Struktur oder benutzen bereits erstellte Mappings. Zusätzlich erfolgte eine Erweiterung um instanzbasierte Matchverfahren [Engmann & Maßmann 2007], die entweder auf Eigenschaften der Instanzen oder auf dem Inhalt basieren.

Das von COMA++ generierte Matchergebnis ist ein Mapping, in dem jedes Element mit mehreren anderen Elementen korrespondieren kann. Somit können auch komplexe Beziehungen abgebildet werden. Beziehungstypen, wie z.B. Spezialisierung oder Generalisierung, werden vom System zwar nicht ausgeschlossen, aber auch nicht speziell unterstützt. Eine Korrespondenz hat keinen zugeordneten Beziehungstyp.

3.2 Instanzbasierte Ansätze

Instanzbasierte Ansätze matchen Kategorien basierend auf den Instanzen (den Produkten), die diesen zugeordnet sind. Dies ist durch die Annahme motiviert, dass die wirkliche Bedeutung einer Kategorie durch die assoziierten Instanzen besser definiert ist als durch die Metadaten wie z.B. den Kategorienamen. Sind zudem noch viele Instanzen vorhanden, so sind die Verfahren meist auch robust gegen einzelne falsch einsortierte Instanzen oder falsch erkannte Instanzduplikate.

GLUE [Doan et al. 2002] wurde für das Matching von Ontologien entworfen und ist ein System, das Lerntechniken verwendet, um halbautomatisch semantische Abbildungen zwischen Ontologien zu erzeugen. Die Architektur besteht aus drei Modulen: dem Distribution Estimator, dem Similarity Estimator und dem Relaxation Labeler.

Der Distribution Estimator errechnet für jedes Klassenpaar der gegebenen Ontologien die gemeinsame Verteilung. Dazu wird ein mehrstrategischer Lernansatz verwendet, das heißt, mehrere Base-Learner, die unterschiedliche Informationen aus den Instanzen oder der Struktur verwerten, und deren Voraussagen werden durch einen Meta-Learner kombiniert. Der Similarity Estimator berechnet aus diesen Verteilungen für jedes Kategoriepaar einen Ähnlichkeitswert. Der Relaxation Labeler nutzt die dadurch entstandene Ähnlichkeitsmatrix und bestimmt mithilfe domänenspezifischer Bedingungen und Heuristiken ein Mapping. Im Gegensatz zu CtxMatch müssen die domänenspezifischen Bedingungen jedoch von Domänenexperten erstellt werden.

Der Ansatz fokussiert auf 1:1-Korrespondenzen. Die erweiterte Version CGLUE unterstützt auch komplexe Mappings, wobei in [Doan et al. 2003] nur die Vereinigung (Union) von Instanzmengen umgesetzt wurde.

Der Ansatz von [Thor et al. 2007] bestimmt die Ähnlichkeit von Klassen aufgrund der sich überlappenden Instanzmengen und wandelt das Ontologie-Matchproblem teilweise in ein Instanz-Matchproblem um. Die Motivation dazu ist, dass das Matching von Instanzen auf spezifischen Datenwerten basiert und deshalb meist einfacher zu lösen ist als das Matching abstrakter Metadaten. Im Idealfall besitzen die Instanzen eine global eindeutige Objektkennung. So verwenden beispielsweise viele Onlineshops eine eindeutige Produktnummer, sogenannte EANs (European Article Number) bzw. UPCs (Universal Product Code), die ein schnelles und einfaches Auffinden von Produktduplikaten ermöglichen. Für den Fall, dass diese Identifizierung nicht gegeben ist, müssen Ansätze des Objekt-Matchings (Duplikaterkennung) angewendet werden, die z.B. die Attributwerte vergleichen.

Die durch das Instanz-Matching erzeugten Instanzkorrespondenzen werden verwendet, um Matches zwischen den zugehörigen Produktkategorien zu ermitteln. Je größer die Instanzüberlappung der Ontologien ist, desto vielversprechender ist dieser Ansatz. Das Verfahren bestimmt außerdem Beziehungen nicht nur zwischen einzelnen Klassen, sondern auch zwischen Mengen von Klassen. Somit werden auch Beziehungstypen zusätzlich zur Äquivalenz ermöglicht.

4 Evaluierungsergebnisse

Viele der Ontologie-Matching-Ansätze wurden in der Forschung entwickelt und auch evaluiert. Zu der Problematik des Matchings von Produktkatalogen gibt es bisher wenig Untersuchungen. Um einen Eindruck davon zu bekommen, inwiefern bestehende Ansätze in der Lage sind, dieses Problem zu lösen, wird das verwandte Problem des Matchings von Webverzeichnissen betrachtet.

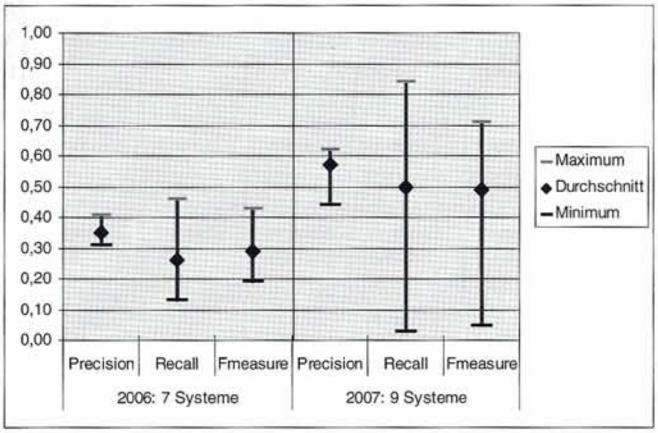
Die Ontology Alignment Evaluation Initiative (OAEI[1]) führt seit 2004 einen Wettbewerb durch, in dem Teilnehmer verschiedene Matchaufgaben lösen. Dies ermöglicht sowohl einen Vergleich der Teilnehmerergebnisse miteinander als auch der Entwicklung über die Jahre.

Unter den gestellten Aufgaben gibt es auch den Directory-Test, bei dem die Webverzeichnisse von Google, Yahoo und Looksmart miteinander gematcht werden müssen. Die Aufgabe bestand sowohl 2006 als auch 2007 darin, für mehr als 4500 Matchaufgaben Mappings zu bestimmen. Da keine Instanzen vorhanden sind, können nur metadatenbasierte Ansätze angewendet werden.

Im Jahr 2006 gab es 7 [Euzenat et al. 2006] und im Jahr 2007 9 Teilnehmer [Euzenat et al. 2007]. Die Ergebnisse für Precision, Recall und Fmeasure sind in Abbildung 4 dargestellt. Die Ansätze konnten im Jahr 2006 durchschnittlich nur ein Viertel aller gesuchten Korrespondenzen ermitteln (Recall 0,26), während fast zwei Drittel aller gefundenen Korrespondenzen falsch waren (Precision 0,35).

Die Ergebnisse ein Jahr später zeigen eine Verbesserung von über 60% für Precision (Durchschnittswert 0,57) und Fmeasure (Durchschnittswert 0,49). Der Recall verdoppelte sich auf 0,50.

Abb. 4: Ergebnisse des OAEI-Directory-Tests – im Vergleich 2006 und 2007



Das System mit den besten Ergebnissen konnte einen Fmeasure-Wert von 0,71 erreichen. 2006 war der höchste erreichte Fmeasure-Wert dagegen 0,43.

Es lässt sich feststellen, dass eine deutlichere Steigerung innerhalb eines Jahres erreicht wurde. Dies könnte auf eine Optimierung der Systeme bezüglich der Matchaufgabe und Erweiterungen um zusätzliche Techniken zurückzuführen sein.

Es ist jedoch auch zu erkennen, dass die Bandbreite der Ergebnisse noch sehr groß ist. Unterschiedliche Ansätze finden auch unterschiedliche Mappings. Nur 15% aller Korrespondenzen wurden von fast allen (8 der 9) Teilnehmer im Jahr 2007 gefunden. Es gibt somit noch Bedarf zur Weiterentwicklung der jeweiligen Ansätze.

5 Fazit

Sowohl im E-Commerce als auch in der Forschung stellt das Matching von Produktkatalogen ein wichtiges Problem dar. Heterogenitäten, Redundanz und mehrfach zugeordnete Instanzen können dabei das Matchproblem erschweren. Zur Lösung des allgemeinen Matchproblems wurden zahlreiche Techniken entwickelt, die Metadaten, Instanzen und auch Zusatzinformationen wie Thesauri verwenden. In diesem Beitrag wurde eine Auswahl von vier Lösungsstrategien in Bezug auf das Matching von Produktkatalogen betrachtet. Da die Evaluierung auf jeweils unterschiedlichen Daten erfolgte, wurden stattdessen Ergebnisse des OAEI-Directory-Tests betrachtet. Dabei konnte eine Verbesserung der durchschnittlichen Ergebnisse innerhalb eines Jahres von über 50% festgestellt werden.

Wünschenswert sind Matchaufgaben mit Instanzen oder eine Erweiterung des bestehenden Tests um diese, damit auch instanzbasierte Ansätze evaluiert werden können. Dabei sollten Instanzen nicht nur einfache, sondern auch komplexe Objekte einschließen, wie dies z.B. in Produktkatalogen der Fall ist.

Abschließend lässt sich feststellen, dass zusätzliche Untersuchungen nötig sind, um weitere vorhandene Matchverfahren in Bezug auf die Herausforderungen zu evaluieren und eine Weiterentwicklung bestehender Ansätze anzuregen.

1. *http://oaei.ontologymatching.org*

Literatur

[Agrawal & Srikant 2001] *Agrawal, R.; Srikant, R.:* On integrating catalogs. In: Proc. of the 10th Int. WWW Conference. China, 2001: 603-612.

[Aumueller et al. 2005] *Aumueller, D.; Do, H.-H.; Maßmann, S.; Rahm, E.:* Schema and ontology matching with COMA++. SIGMOD Conference. USA, 2005: 906-908.

[Bouquet et al. 2003] *Bouquet, P.; Serafini, L.; Zanobini, S.:* Semantic coordination: a new approach and an application. In: Proc. of the 2nd ISWC. USA, 2003:130-145.

[Doan et al. 2002] *Doan, A.; Madhavan, J.; Dhamankar, R.; Domingos, P.; Halevy, A.:* Learning to map between ontologies on the semantic web. In: The Eleventh International WWW Conference, USA, 2002.

[Doan et al. 2003] *Doan, A.; Madhavan, J.; Dhamankar, R.; Domingos, P.; Halevy, A.:* Learning to match ontologies on the Semantic Web. The VLDB Journal 12, 4 (Nov. 2003), 303-319.

[Elmagarmid et al. 2007] *Elmagarmid, A.; Ipeirotis, P.; Verykios, V.:* Duplicate Record Detection: A Survey; IEEE Transactions on Knowledge and Data Engineering, 2007, 1-16.

[Engmann & Maßmann 2007] *Engmann, D.; Maßmann, S.:* Instance Matching with COMA++. BTW Workshops. Germany, 2007: 28-37.

[Euzenat & Shvaiko 2007] *Euzenat, J.; Shvaiko, P.:* Ontology Matching. Springer-Verlag, 2007.

[Euzenat et al. 2006] *Euzenat, J.; Mochol, M.; Shvaiko, P.; Stuckenschmidt, H.; Šváb, O.; Svátek, V.; van Hage, W. R.; Yatskevich, M.:* Results of the Ontology Alignment Evaluation Initiative 2006. In: Proceedings of the Ontology Matching Workshop at ISWC'06, 2006.

[Euzenat et al. 2007] *Euzenat, J.; Isaac, A.; Meilicke, C.; Shvaiko, P.; Stuckenschmidt, H.; Šváb, O.; Svátek, V.; van Hage, W. R.; Yatskevich, M.:* Results of the Ontology Alignment Evaluation Initiative 2007. In: Proceedings of the Ontology Matching Workshop at ISWC'07, 2007.

[Rahm & Bernstein 2001] *Rahm, E.; Bernstein, P.:* A survey of approaches to automatic schema matching. The VLDB Journal 10, 4 (Dec. 2001), 334-350.

[Schulten et al. 2001] *Schulten, E.; Akkermans, H.; Botquin, G.; Dörr, M.; Guarino, N.; Lopes, N.; Sadeh, N.:* Call for Participants: The E-Commerce Product Classification Challenge. IEEE Intelligent Systems 16, 4 (Jul. 2001), 86-c3.

[Shvaiko & Euzenat 2005] *Shvaiko, P.; Euzenat, J.:* A survey of schema-based matching approaches. Journal on Data Semantics IV (2005), 146-171.

[Thor et al. 2007] *Thor, A.; Kirsten, T.; Rahm, E.:* Instance-based matching of hierarchical ontologies. BTW. Germany, 2007: 436-448.

Sabine Maßmann
studierte Informatik an den Universitäten Rostock und Leipzig. Seit 2006 ist sie Stipendiatin im Graduiertenkolleg Wissensrepräsentation an der Universität Leipzig und promoviert bei Prof. Dr. Rahm zu dem Thema »Schema- und Ontologie-Matching«.

Dipl.-Inform. Sabine Maßmann
Universität Leipzig
Abteilung Datenbanken
Postfach 100920
04009 Leipzig
massmann@informatik.uni-leipzig.de
www.informatik.uni-leipzig.de