Universität Leipzig
Fakultät für Mathematik und Informatik
Institut für Informatik

Preprocessing für das Matchen von Produktangeboten

Bachelorarbeit

Leipzig, September 2010

vorgelegt von Stefan Thomas
Studiengang Informatik

Betreuender Hochschullehrer: Prof. Dr. Erhard Rahm
Fakultät für Informatik und Mathematik, Institut für Informatik, Abteilung Datenbanken

Inhaltsverzeichnis

1 Einleitung

1.1 Motivation

Sowie Computer eine immer größer werdende Rolle im Leben einnehmen, so bekommen auch digital gespeicherte Daten immer mehr Aufmerksamkeit und erfreuen sich einer stetig steigenden Verwendung. Im Vergleich zu konservativen Methoden der Informationsverarbeitung ohne unterstützende Computertechnik, kann man in sehr kurzer Zeit große Mengen an Daten verarbeiten und so bestehende Datenbestände erweitern oder neue Datenbestände erzeugen. Damit einhergehen auch die Probleme, die eine solche Speicherung mit sich bringt: Eine manuelle Konsolidierung der Ergebnisse ist weder zeitlich noch finanziell umsetzbar. Vor allem doppelte Datensätze innerhalb eines Datenbestands stellen ein großes Problem dar.

Das Erkennen von Duplikaten ist daher insbesondere im kommerziellen Bereich ein wichtiger Schritt, um unnötige Kosten zu vermeiden und potentielle Schäden abzuwenden. Diesen Vorgang nennt man Dublettenerkennung, Objekt-Matching oder auch Entity-Matching. Ziel ist es Datensätze innerhalb eines Datenbestands zu identifizieren, welche sich auf dasselbe Realweltobjekt beziehen.

Es existieren bereits zahlreiche Strategien, um Objekt-Matching durchzuführen [KR10, EIV07]. Die meisten bauen auf Ähnlichkeitsfunktionen, wie zum Beispiel der Levenstein- und der Jaro-Winkler-Distanz für einfache und kurze Zeichenketten [CFR03, LN07], auf. Für längere Zeichenketten, insbesondere für Zeichenketten, die aus mehreren Wörtern bestehen, eignen sich besser tokenbasierte Ähnlichkeitsmaße, wie die Jaccard-Ähnlichkeit oder die Termfrequenz/Inverse Dokumentfrequenz (TF/IDF) [LN07]. Für andere Datentypen existieren ähnliche Distanzmetriken. Mit Hilfe dieser Abstandsmaße und eines oder mehrerer Schwellwerte ist es möglich, gewisse Unterschiede in der Informationsdarstellung zu tolerieren. Diese Ansätze können ihre volle Wirksamkeit aber nur bei gut strukturierten und einheitlichen Daten entfalten. Zum Beispiel schlägt TF/IDF fehl, wenn gewisse Token einer Zeichenkette zwar insgesamt häufig auftreten, aber diesen bei der Beurteilung, ob zwei Datensätze dasselbe Objekt referenzieren, dennoch eine sehr hohe Bedeutung zukommt.

Insbesondere Webdaten offenbaren häufig wenig über ihre Struktur und weisen zudem in aller Regel eine sehr starke Heterogenität auf – sowohl in Struktur als auch im Inhalt. Das bedeutet, es ist sinnvoll vor dem eigentlichen Objekt-Matching, die Struktur zu vereinheitlichen, die Daten zu bereinigen und implizite Informationen explizit zu machen. Die Vereinheitlichung der Struktur, welche auch als Schema bezeichnet wird, ist unter dem Namen Schema-Matching bekannt und wird innerhalb dieser Arbeit als gegeben angenommen und somit keine weitere Beachtung finden. Die beiden anderen Schritte, also die Datenbereinigung und die Datenextraktion, stellen das hier betrachtete Preprocessing dar.

1.2 Einordnung Produkt-Matching

Produkt-Matching ist ein Spezialfall des Objekt-Matching, siehe Abschnitt 1.1, für den bisher nur vergleichsweise wenige Ansätze publiziert wurden [BBS05, BGG+07]. Beim Produkt-Matching ist man daran interessiert alle Produktangebote eines oder mehrerer Händler zu bestimmen, welche sich auf das gleiche Produkt beziehen. Es handelt sich um einen entscheidenden, aber auch schwierigen Schritt bei der Datenintegration.

Als Datenintegration bezeichnet man den Vorgang des Zusammenführens von Informationen aus verschiedenen Datenquellen, wobei sich die Datenquellen häufig in der Struktur unterscheiden und somit eine einheitliche Datenstruktur während der Integration erst geschaffen werden muss. Ziel ist es, den Zugriff auf die Daten der einzelnen Datenbestände zu vereinheitlichen, um effizienter auf diese zugreifen zu können, als es ein direkter Zugriff auf die Datenquellen ermöglicht.

Notwendig ist Datenintegration insbesondere dort, wo unabhängig voneinander entwickelte Systeme bzw. deren Daten vereinigt werden sollen. Dies können Systeme mit identischem Zweck sein, wie es zum Beispiel der Fall beim Zusammenfassen von Produktangeboten verschiedener Händler ist. Es kann sich aber auch um Systeme mit unterschiedlichem Zweck handeln, wie etwa beim Zusammenführen der Daten aus einem Verkaufs- und einem Bewertungssystem.

In beiden Fällen können sich die Datenstrukturen erheblich voneinander unterscheiden und darüber hinaus kommt es beim Vereinigen von Datenbeständen naturgemäß zum Auftreten von Duplikaten. Dubletten bzw. Duplikate sind mehrfache Repräsentationen desselben Realweltobjekts. Bezugnehmend zum Produkt-Matching, ist ein typisches

Beispiel für ein Duplikat ein Produkt, welches von mehreren Händlern angeboten wird. Sobald eine einheitliche Struktur vorliegt, also nach der sogenannten Schemaintegration, muss demnach eine Dublettenerkennung durchgeführt werden. Dabei werden semantisch äquivalente Objekte identifiziert.

Ohne diesen Schritt käme es zum einen zu wirtschaftlichen Schäden, zum anderen sind ohne Objekt-Matching typische E-Commerce[1] Anwendungen gar nicht möglich. Für eine vollständige Konkurrenzanalyse ist es zum Beispiel erforderlich, die Preise zu denen bestimmte Produkte angeboten werden herauszufinden. Das heißt von *vielen verschiedenen* Anbietern müssen die Preise für die jeweiligen Produkte ermittelt werden. Dass dieser Schritt nicht trivial ist, sticht sofort ins Auge, wenn man bedenkt wie differenziert Produktangebote aussehen können. Nur die textuellen Beschreibungen in Abb. 1 sind beispielsweise kein zwingendes Argument für die semantische Gleichheit von Angebot 1 und 3 – im Gegenteil: Es wird eher eine Gleichheit vom zweiten und dritten Angebot impliziert. Die Preise unterscheiden sich zwar erheblich voneinander, liegen aber noch im Rahmen des Wettbewerbs.

Abb. 1: Produktangebote aus dem Internet

Sowohl die in [BBS05] veröffentlichten und auf Produktangaben von Froogle basierenden Evaluationsergebnisse mit F-Measure-Werten[2] von weniger als 0,6 als auch die in [BGG+07] betrachtete Laufzeitkomplexität unterstreichen die besondere Schwierigkeit des Produkt-Matching.

[1] E-Commerce bezeichnet „jede Art von geschäftlichen Transaktionen […] sowie elektronisch abgewickelte Geschäftsprozesse […], bei denen die Beteiligten auf elektronischem Wege […] miteinander verkehren […]“ [G8].

[2] F-Measure ist ein Maß, welches die Qualität einer Treffermenge bzgl. Genauigkeit *und* Trefferquote angibt.

1.3 Ziel dieser Arbeit

Das Ziel ist es, Strategien zu entwickeln, die eine systematische Vorverarbeitung von Produktangeboten erlauben. Die Extraktion von impliziten Informationen und Datenaufbereitung in Form von Datenbereinigung soll die Qualität von Produktangeboten verbessern. Spezieller Fokus liegt insbesondere auf Angeboten von Elektronikprodukten. Die Vorverarbeitung soll idealerweise vollautomatisch durchgeführt werden können, wobei dies aufgrund der Domänenabhängigkeit, selbst im Teilgebiet des Produkt-Matching, nicht zu 100% erreicht werden kann.

1.4 Gliederung

Die Arbeit ist wie folgt gegliedert: In Kapitel 2 werden grundlegende Definitionen eingeführt und die Ausgangslage anhand einer gegebenen Testmenge detailliert analysiert. Ausgehend von den gewonnen Erkenntnissen werden in Kapitel 3 Möglichkeiten der Vorverarbeitung vorgestellt. Die entwickelten Verfahren werden in Kapitel 4 anhand verschiedener Testszenarien evaluiert. Im abschließenden Kapitel wird eine Zusammenfassung gegeben und es werden Weiterentwicklungsmöglichkeiten aufgezeigt.

2 Analyse

In diesem Kapitel werden zunächst grundlegende Begrifflichkeiten eingeführt. Ausgehend von einer detaillierten Analyse der Ausgangslage werden Ansatzpunkte für eine notwendige und vielversprechende Vorverarbeitung ermittelt. Weiterhin werden sowohl innerhalb eines Produktangebots liegende als auch angebotsübergreifende Zusammenhänge betrachtet, welche beim Matchen und bei der Vorverarbeitung nützlich sein können. Als Ausgangsdaten dienen hierbei ca. 100.000 Elektronikproduktangebote eines Preisvergleichsportals. Die Angebote stammen von 60 verschiedenen Anbietern. Eine detaillierte Vorstellung der Ausgangsdaten erfolgt im Kapitel 4.

2.1 Definitionen

Def. Produktangebot: Ein Produktangebot (kurz: Angebot) ist eine Instanz eines Realweltprodukts.

Def. Attribut: Ein Attribut ist eine Eigenschaft, mit deren Hilfe Produktangebote näher beschrieben werden können. Zu diesem Zweck kann es für jedes Produktangebot einen Wert annehmen.

Def. Wert: Sei O eine Menge von Produktangeboten $o_1, o_2, \dots, o_n$ und seien $a_1, a_2, \dots, a_m$ Attribute. Dann Bezeichne $v_{i,j}$ den Wert des i-ten Angebots o_i für das Attribut a_j. Ein Wert beschreibt demnach ein spezifisches Produktangebot bzgl. eines bestimmten Attributs. Ein solcher Wert kann elementar sein oder aus mehreren Komponenten bestehen.

Def. Mapping: Sei O eine Menge von Produktangeboten. Eine Menge von Paaren von Elementen aus O wird als Mapping bzgl. O bezeichnet. Das vollständige Mapping ist das kartesische Produkt von O mit sich selbst.

Das Produktmatching-Problem lässt sich nun folgendermaßen formulieren:

> *Gegeben sei eine Menge von Produktangeboten O. Finde das größte Mapping M bzgl. O, so dass für alle $(o_k, o_l) \in M$ gilt: o_k und o_l sind Instanzen des gleichen Produkts.*

Im Folgenden sei O wieder eine Menge von Produktangeboten $o_1, o_2, \ldots, o_n$ und a_1 ein Attribut. a_1 ist genau dann *sauber*, wenn für alle $o_k \in O$ gilt: $v_{k,1}$ ist korrekt und es existiert kein $o_j \in O$ mit $o_j \neq o_k$, so dass $v_{k,1}$ und $v_{j,1}$ semantisch äquivalent sind. Ansonsten ist a_1 *unsauber*.

Gründe für inkorrekte Werte können Rechtschreibfehler, Tippfehler oder auch fehlerhafte Datenerhebung sein. Inkorrekt sei ein Wert auch dann, wenn dieser zu allgemein ist, das heißt, wenn der Wert zwar nicht falsch ist, es aber einen spezialisierteren Wert innerhalb der Wertemenge des Attributs gibt, der ebenfalls korrekt ist. Syntaktisch unterschiedliche, aber semantisch äquivalente Werte entstehen in der Regel dadurch, dass mehrere *legitime* Schreibweisen existieren.

a_1 ist genau dann *vollständig*, wenn für alle $o_k \in O$ gilt: o_k hat für a_1 einen Wert ungleich *null*, das heißt $v_{k,1}$ ist ein nichtleerer Wert. Ansonsten ist a_1 *unvollständig*.

a_1 ist genau dann *einfach*, wenn für alle $o_k \in O$ gilt: $v_{k,1}$ ist elementar. Ansonsten ist a_1 *zusammengesetzt*.

Titel	Hersteller	Kategorie	Eigenschaften
HP Photosmart C6180	HP	Drucker	weiß
HP Photosmart C6180	Hewlett-Packard	Drucker	schwarz
Canon PIXMA MP640	Canon	Drucker	weiß, WiFi
PIXMA ip4700		Drucker	
Kyocera FS 1300D	Kyocera	Drucker	
C7280 inkl. Ersatzpatronen		Drucker	
Kyocera FS 1300D	Kyozera	Peripherie	

Tab. 1: Angebote zur Verdeutlichung von Attributeigenschaften

Zur Verdeutlichung dieser Eigenschaften siehe Tab. 1. Diese zeigt exemplarisch sieben Produktangebote und vier Attribute. Das Herstellerattribut ist weder sauber, noch vollständig, denn *HP* und *Hewlett-Packard* sind semantisch äquivalent, *Kyozera* wahrscheinlich falsch geschrieben und für zwei Angebote ist kein Wert angegeben. Das Attribut *Kategorie* ist zwar vollständig, denn für alle Angebote ist ein Wert angegeben, aber unsauber, da *Peripherie* ein Überbegriff ist und der speziellere Wert *Drucker* eben-

falls korrekt ist. Beide Attribute sind darüber hinaus einfach, da jeder der Werte nur eine einzige Komponente darstellt. Das Titelattribut ist sowohl sauber als auch vollständig, aber nicht einfach, da beispielsweise *C7280 inkl. Ersatzpatronen* aus zwei Komponenten besteht: Name bzw. Kennung des Produkts und Zubehör. Auch die anderen Werte bestehen bei genauerer Betrachtung aus mehreren Komponenten, denn fast alle enthalten neben dem Namen bzw. der Kennung noch eine Herstellerangabe. Das Attribut *Eigenschaften* ist ebenfalls zusammengesetzt, da *weiß, WiFi* aus zwei Komponenten besteht. Außerdem ist dieses Attribut unvollständig, aber sauber.

Sowohl unvollständige als auch unsaubere Attribute wirken sich negativ auf die Matching-Qualität aus. Es ist somit anzustreben, semantisch äquivalente Attributwerte als solche zu erkennen und zusammenzuführen, sowie inkorrekte Attributwerte bzw. null-Werte zu entfernen und durch die korrekten Werte zu ersetzen. Diese Prozesse unterstützen den Matching-Vorgang und helfen damit die Qualität des Matching im Hinblick auf Precision[3] und Recall[4] zu verbessern und die Laufzeit zu verringern.

2.2 Produktangebot

Produktangebote werden teils aus strukturierten, teils aus semi-strukturierten Webdaten gewonnen. Die Ursache für die partiell vorhandene Strukturierung sind sogenannte Preisvergleichsportale, denn diese schreiben den Webseitenbetreibern oftmals den Aufbau der Angebote, das heißt welche Attribute verwendet werden sollen und welche davon obligatorisch bzw. fakultativ sind, vor. Die Extraktion aus Webdaten hat aber dennoch zur Folge, dass die Granularität der Attribute vergleichsweise gering ausfällt, die Qualität der zur Verfügung stehenden Werte häufig mangelhaft ist und ggfs. eine Vielzahl von Angeboten für ausgewählte Attribute keine Werte aufweist. Es erlaubt aber, einige Annahmen zu treffen. Zum Beispiel bietet ein Händler im Internet selten das gleiche Produkt mehrfach an. Diese Information lässt sich beim Matchen sehr gut verwenden, aber natürlich ist sie ggfs. auch beim Vorverarbeiten nützlich. Ein weiterer Nachteil bei der Webdatenintegration liegt darin, dass die Anzahl der unterschiedlichen Quellen erheblich größer ist, als dies bei einer klassischen Datenintegrationsaufgabe der Fall ist. Die Problematik der Datenheterogenität wird dadurch zusätzlich verschärft.

[3] Precision ist ein Maß, welches die Genauigkeit einer Treffermenge angibt.
[4] Recall ist ein Maß, welches die Trefferquote einer Treffermenge angibt.

Trotz der genannten Probleme gibt es mehrere Attribute, für welche Produktangebote häufig nichtleere Werte vorweisen. Wichtige Attribute, die zugleich selten leere Werte annehmen, sind: *Titel*, *Beschreibung*, *Preis*, *Kategorie*, *Händler* und *Hersteller*. Auf diese soll hier näher eingegangen werden.

Das Attribut *Händler* gibt an, wer das Produkt anbietet bzw. auf welcher Webseite es angeboten wird. Zu welchem Preis der Händler dieses anbietet gibt das Attribut *Preis* an. Produkte lassen sich regelmäßig bestimmten Kategorien zuordnen, zum Beispiel zu den Kategorien *Küchengeräte* oder *Zubehör für Digitalkameras*. Eine solche Zuordnung erfolgt durch das Attribut *Kategorie*. Das Attribut *Hersteller* gibt die Firma wieder, also den Namen des Unternehmens, welches das angebotene Produkt produziert. Diese Attribute sind alle einfach. In der Regel sind für jedes Angebot der Preis, die Kategorie und der Händler gegeben, das heißt die gleichnamigen Attribute sind im Gegensatz zum Herstellerattribut vollständig.

2.2.1 Preis, Händler, Kategorie und Hersteller

Aufgrund ihrer weitestgehenden Vollständigkeit und ihrer Einfachheit, eignen sich die Attribute *Preis*, *Kategorie*, *Händler* und *Hersteller* innerhalb des Matching-Vorgangs besonders zum Partitionieren der Ausgangsdaten bzw. zum Einschränken des Suchraums. Der Suchraum des Matching-Problems ist ursprünglich das kartesische Produkt der Eingangsmenge. Durch das sogenannte Blocking kann der Suchraum zum einen erheblich verkleinert werden und zum anderen ist im Falle einer Partition das Teilen des Matching-Problems in kleinere Teilprobleme möglich. Dies hat den entscheidenden Vorteil, dass so die Möglichkeit der parallelen Abarbeitung entsteht. Dies wiederum ermöglicht es selbst mit großen Datenmengen skalieren zu können. Es empfiehlt sich folglich eine weitergehende Betrachtung, um, falls eine Datenbereinigung notwendig ist, die Möglichkeiten dafür abzuschätzen.

Da der Preis praktisch für jedes Angebot als separates Attribut vorliegt, ist es nicht erforderlich Möglichkeiten zu erforschen den Preis beispielsweise aus anderen Attributen zu extrahieren. Es könnte aber nötig sein inkorrekte Preise zu erkennen und diese ggfs. zu ersetzen. So könnte ein Preis, welcher versehentlich 10-mal so hoch wie gewöhnlich angegeben ist, dafür verantwortlich sein, dass das Angebot nicht mehr korrekt gematcht werden kann, da beim Blocking die entscheidenden Elemente des Mappings entfernt

worden sind. Dieser Ansatz wird dennoch im Kapitel 3 nicht weiterverfolgt, denn das Attribut *Preis* ist im Großen und Ganzen sauber und es gibt durchaus auch Produkte, die mit einer großen Preisspanne angeboten werden, zum Beispiel subventionierte Handys. Das Erkennen eines wirklich inkorrekten Preises ist demnach äußerst schwierig.

Der Händler lässt sich selbst bei der Datengenerierung aus Webdaten sehr leicht feststellen und ist somit als stets korrekt anzusehen. Es ist demzufolge nicht notwendig eine Datenbereinigung durchzuführen. Wie bereits erwähnt, bietet ein Händler jedes Produkt in den meisten Fällen nicht mehrfach an. Daraus folgt, dass zwei Angebote mit identischem Wert für das Händlerattribut meist nicht Instanzen des gleichen Produkts sind. Dies ist insofern wichtig, als Angebote eines Händlers trotz unterschiedlicher Produkte häufig starke Ähnlichkeiten zueinander aufweisen. Diese überwiegen vielfach sogar die Ähnlichkeiten, die zwischen Angeboten besteht, die von unterschiedlichen Händlern stammen, obwohl sich die Angebote auf das gleiche Produkt beziehen.

Jeder Händler verwendet einen eigenen Kategorienbaum. Es ist daher kaum zu erwarten, dass sich die Kategorien der verschiedenen Händler und somit auch nicht die Angebote in den jeweiligen Kategorien entsprechen. Schwerwiegender ist allerdings, dass die Angebote vielfach sehr ungenau oder sogar falsch eingeordnet sind. So ist zum Beispiel ein Objektiv für eine Digitalkamera eigentlich in die Kategorie *Zubehör für Digitalkameras* einzuordnen. Häufig wird ein solches Produkt stattdessen aber direkt in die Kategorie *Digitalkameras* einsortiert. Beim Zusammenführen der Angebote mehrerer Händler ist es deshalb zielführend, dem unsauberen Attribut *Kategorie* Rechnung zu tragen. Dazu existieren bereits zahlreiche Ansätze. Vielversprechend ist es zum Beispiel über ausgewählte Produkte festzustellen in welchen Kategorien Angebote dieser Produkte zu finden sind und so herauszubekommen welche Kategorien zusammengehören. Die ausgewählten Produkte müssen dafür zwar innerhalb der Produktangebote eindeutig identifizierbar sein, aber für einen kleinen Teil der Angebote ist dies gegeben. Einzelheiten zu diesem Verfahren und einen Überblick über andere Möglichkeiten werden in [KRT07] vorgestellt.

Nicht jeder Händler gibt den jeweiligen Hersteller seiner angebotenen Produkte explizit an. Dies kann einerseits auf Unwissenheit zurückzuführen sein oder daran liegen, dass sich diese Information bereits im *Titel* oder auch in der *Beschreibung* des Angebots befindet. Im zweiten Fall ist es oft möglich leere Attributwerte durch nichtleere und kor-

rekte Werte zu ersetzen, da die hierfür nötige Information bereits in den Daten vorhanden ist. Sollte der Händler aber selbst nicht genügend Informationen zur Verfügung gehabt haben, um den Hersteller angeben zu können, so ist es für ein automatisches Verfahren ohne menschliche Kompetenz nahezu unmöglich die erforderlichen Informationen zu beschaffen, um sich auf einen Produzenten festlegen zu können bzw. überhaupt einen zu bestimmen.

Unabhängig von der Tatsache, dass das Herstellerattribut unvollständig ist, besteht auch ein Problem mit der Sauberkeit dieses Attributs. Viele Unternehmen haben unterschiedliche Schreibweisen oder verbreitete Abkürzungen, nennen sich im Laufe ihres Bestehens um, haben Tochterunternehmen, welche synonym verwendet werden, fusionieren oder ändern ihre Rechtsform: Es gibt unzählige Gründe dafür, dass das Attribut *Hersteller* unsauber ist.

Zur Verdeutlichung: *HP* ist die gebräuchliche Abkürzung von *Hewlett-Packard*, welches aber auch als *Hewlett Packard* oder *HewlettPackard* geschrieben oder inkl. seiner Rechtsform angegeben werden kann: *Hewlett-Packard GmbH*. Ein anderes Beispiel ist *Fujifilm* oder auch *Fujifilm Deutschland*, *Fujifilm digital*, *Fuji Film*, *Fuji-Film*, *FUJIFILM Electronic Imaging Europe GmbH* oder einfach nur *Fuji* und das ist nur eine kleine Auswahl der möglichen Werte für das Attribut *Hersteller* eines Angebots für ein Produkt dieses Unternehmens.

Hieran ist erkennbar wie dringlich eine Datenbereinigung für dieses Attribut ist.

2.2.2 Titel und Beschreibung

Titel sowie *Beschreibung* eines Produktangebots vereinen im Allgemeinen eine Fülle von Informationen und stellen somit zusammengesetzte Attribute dar. So beinhalten diese Attribute Produkteigenschaften wie Farbe und Größe, angebotsspezifische Eigenschaften wie die Verkaufseinheit[5] oder zusätzliche Beigaben und nicht zuletzt Produktcodes. In beiden Attributen kann aber auch Freitext ohne Bedeutung für den Matching-Prozess einen Großteil des Textes ausmachen.

Beim Vergleich von Werten dieser Attribute sind also insbesondere die Probleme präsent, die sich aus der Heterogenität der ursprünglichen Webdaten ergeben. So können

[5] Als Verkaufseinheit wird die Zusammenfassung mehrerer Artikel des gleichen Produkts bezeichnet.

sich sehr voneinander unterscheidende Attributwerte dennoch auf das gleiche Produkt beziehen, siehe Abb. 2. Sowohl *Titel* als auch *Beschreibung* sind daher als unsauber anzusehen.

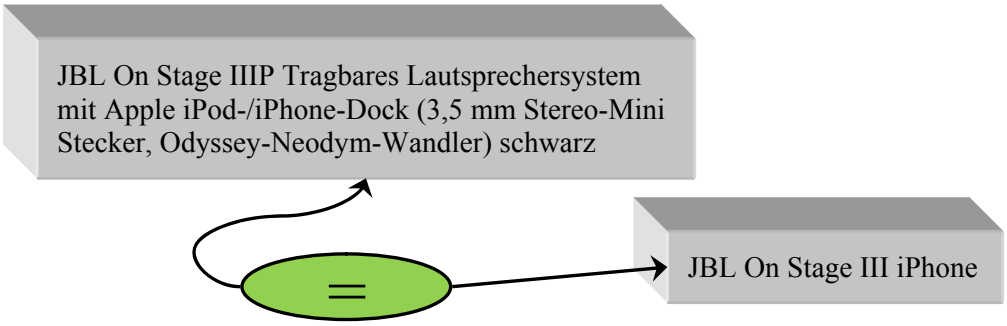

Abb. 2: Identische Produkte trotz unterschiedlicher Titel

Andererseits können selbst Angebote mit fast identischen Werten für *Titel* und *Beschreibung* Instanzen unterschiedlicher Produkte sein, siehe Abb. 3.

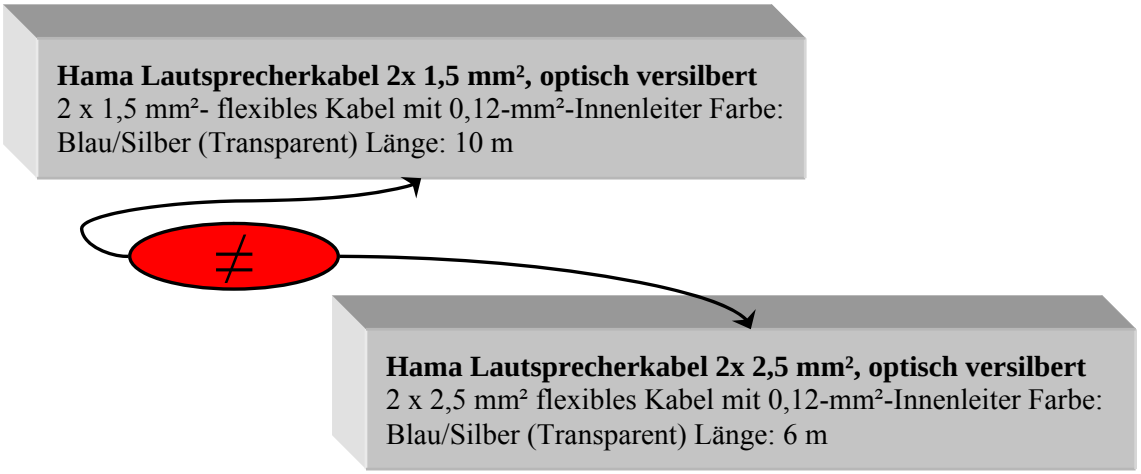

Abb. 3: Unterschiedliche Produkte trotz sehr ähnlicher Titel und Beschreibung

Manchmal erweist es sich aber selbst für den Menschen als schwierig festzustellen, ob zwei Titel das gleiche Produkt bezeichnen. Handelt es sich zum Beispiel in Abb. 4 um identische Produkte oder nicht? Um diese Frage beantworten zu können, muss man mindestens wissen, dass Verl.Ka abgekürzt für Verlängerungskabel und CI. für Cinch steht und ein solches Cinch-Verlängerungskabel auf einer Seite Stecker und auf der anderen Seite Kupplungen hat – für ein vollautomatisches Verfahren ohne weitere Wissensquellen nicht lösbar.

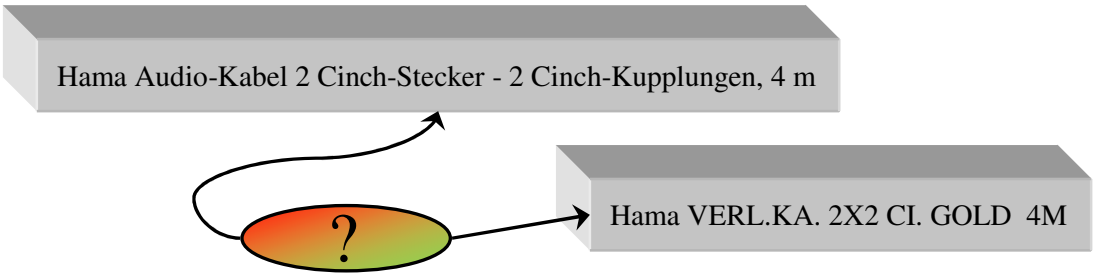

Abb. 4: Identische Produkte?

Der Titel ist, trotz der vorgestellten Probleme, das wichtigste Attribut für das Produkt-Matching und ist auch stets verfügbar. Das Attribut *Beschreibung* ist hingegen nicht vollständig, beinhaltet aber auch oft wichtige Informationen, um die Entscheidung zu beeinflussen, ob sich zwei Angebote auf das gleiche Produkt beziehen.

Ein direkter Vergleich von Werten dieser beiden Attribute ist – selbst unter Verwendung textbasierter Ähnlichkeitsmaße wie TF/IDF – nicht sehr erfolgversprechend. Auch der Versuch der direkten Datenbereinigung ist nicht zielführend, denn das ist ähnlich komplex wie das ursprüngliche Problem des Produkt-Matching. Konstruktiver ist es, aus dem Titel und auch aus der Beschreibung Informationen zu extrahieren und diese mit wiederum extrahierten Informationen anderer Angebote zu vergleichen. Eine Bereinigung dieser elementareren Informationen ist zweckmäßig und weniger komplex.

2.2.3 Produktcode

Ein Großteil der Produkte, insbesondere der Elektronikprodukte, ist charakterisiert durch einen Produktcode. Anders als in [T09] wird im Folgenden der Produktcodebegriff weiter gefasst. Demnach ist ein Produktcode eine vom Hersteller des Produkts gewählte Zeichenfolge zum Zweck der Produktidentifikation, welche meistens kein Wort des allgemeinen Sprachgebrauchs darstellt, sondern eine Aneinanderreihung von Buchstaben, Sonderzeichen und Ziffern ist, *die zusätzlich durch ein oder mehrere Leerzeichen unterbrochen sein kann*. Ähnliche Codes stellen dabei häufig Ähnlichkeiten zwischen Produkten dar und teilweise enthalten Produktcodes auch Eigenschaften des Produkts – zum Teil in kodierter Form.

Abb. 5 verdeutlicht die gemachten Aussagen. Sowohl *TVCCD-125PCOL* als auch *TVCCD-182HCOL* sind Kameras des Herstellers Monacor. Sie unterscheiden sich aber zum Beispiel darin, dass Erstgenannte eine Miniaturkamera mit geringer Auflösung ist und die zweite Kamera hochauflösende Bilder aufnehmen kann. Die beiden Produkt-

codes 1810TZ-412G25n und 1810TZ-414G50n sind ebenfalls sehr ähnlich zueinander und tatsächlich beziehen sich beide auf fast das gleiche Notebook von Acer. In diesen Codes sind außerdem Eigenschaften kodiert. So steht beispielsweise die *50* in *1810TZ-414G50n* für eine Festplattenkapazität von 500GB.

Abb. 5: Auswahl einiger Produktcodes

Nicht zu verwechseln ist der hier vorgestellte Produktcode mit dem Universal Product Code (UPC) oder der European Article Number (EAN). Dabei handelt es sich um zentral vergebene Nummerncodes, welche ein Produkt eindeutig identifizieren. Aufgrund dieser Eigenschaft, sind diese Nummern zwar ideal zum Matchen von Angeboten geeignet, aber im Regelfall ist diese Information nur bei einem geringen Teil der Angebote als separates Attribut gegeben – bei Elektronikangeboten liegt der Anteil bei 25 – 30%. Somit kann sich ein Matching-Verfahren nicht allein darauf stützen. Noch seltener werden der UPC oder die EAN im Titel oder in der Beschreibung zu einem Produkt angegeben. Das heißt, dass die Möglichkeit, per Vorverarbeitung einen UPC oder eine EAN zu ermitteln und so das Matching zu verbessern, ebenfalls stark eingeschränkt ist.

Es lässt sich jedoch festhalten, dass auch Produktcodes eine weitestgehend eindeutige Zuordnung zu einem Produkt ermöglichen. Insbesondere innerhalb eines Herstellers sind sie ein nahezu individueller Bezeichner. Außerdem gibt es nur sehr wenige Produktcodes, die gleichzeitig von mehreren Herstellern vergeben werden. Das bedeutet, dass sich Angebote, welche den gleichen Produktcode beinhalten, mit sehr hoher Wahrscheinlichkeit auf das gleiche Produkt beziehen. Andererseits bedeutet es auch, dass zwei Angebote, welche jeweils sehr unterschiedliche Produktcodes enthalten, nur mit sehr geringer Wahrscheinlichkeit das gleiche Produkt referenzieren.

Im Gegensatz zu UPC oder EAN werden Produktcodes sehr häufig in den Freitextattributen eines Angebots angegeben. Dies hat den Hintergrund, dass auch die Kunden über

diesen Code die angebotenen Produkte identifizieren. Zudem ist es teilweise schwierig, zwischen Produktcode und dem Namen des Produkts zu unterscheiden.

Zielführend ist es demzufolge diese Produktcodes im Text zu erkennen, zu extrahieren und als eigenständiges Attribut zur Verfügung zu stellen.

Abb. 6 zeigt eine Reihe von Beispielen. Es ist ersichtlich, dass es sehr unterschiedliche Typen von Produktcodes gibt. Manche bestehen ausschließlich aus Buchstaben, andere aus Buchstaben, Sonderzeichen und Ziffern und einige beinhalten eine große Anzahl an zusätzlichen Leerzeichen und erstrecken sich somit über viele Wortgrenzen. Diese Problematik wird im dritten Kapitel näher betrachtet.

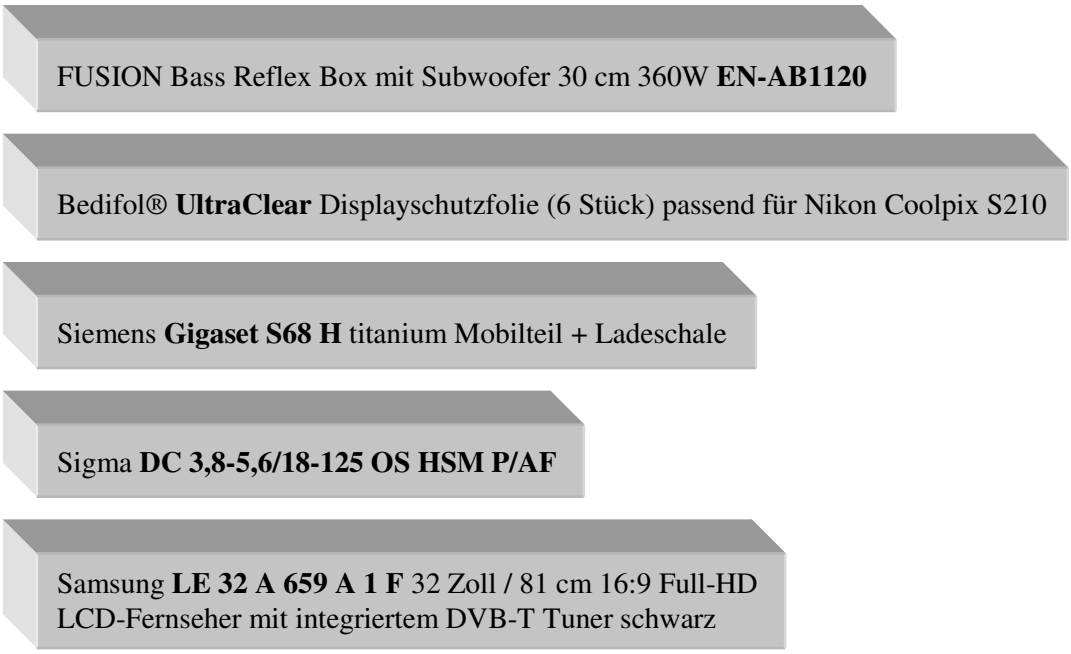

Abb. 6: Verschiedene Typen von Produktcodes

Die Extraktion des Produktcodes des *angebotenen* Produkts ist nicht trivial. In einem Angebot, genauer gesagt im Titel und in der Beschreibung eines Angebots, können mehrere Produktcodes vorkommen. Gründe hierfür sind zum Beispiel, dass es sich bei dem Angebot um ein aus mehreren einzelnen Produkten bestehendes Bundle handelt oder dass Produkte genannt werden für welche das angebotene Produkt geeignet ist. Eine Hochrechnung von manuell annotierten Datensätzen hat ergeben, dass für etwa 75 – 80% der Elektronikproduktangebote der Produktcode des angebotenen Produkts im Titel oder in der Beschreibung zu finden ist. Die Annotation erfolgte im Rahmen dieser Arbeit, um die in Kapitel 3 vorgestellte Strategie zur Identifikation und Extraktion von Produktcodes zu evaluieren.

Anhand von ausgewählten Beispielen soll im Folgenden aufgezeigt werden, welche Schwierigkeiten bei der Extraktion von Produktcodes berücksichtigt werden müssen. In Abb. 7 ist der Titel eines Angebots abgebildet. In diesem sind drei Produktcodes enthalten und zwar *BH-800*, *2600 Classic* und *2630*. Offensichtlich bezieht sich aber nur Erstgenannter auf das angebotene Produkt. Die restlichen Codes geben dagegen an für welche Produkte man das angebotene Produkt nutzen kann. Dies ist für den Käufer eine wichtige Information, aber störend bei der vollautomatischen Identifikation des korrekten Produktcodes. Auffällig sind an dieser Stelle zwei Aspekte: Der am weitesten vorn stehende Code ist der gesuchte Code und die unerwünschten Codes werden zusätzlich durch die Präposition *für* eingeleitet.

Nokia Bluetooth-Headset **BH-800** kaffee-schwarz *für* Nokia **2600 Classic**; **2630**;

Abb. 7: Nokia BH-800

Ein weiteres Wort, welches darauf hinweist, dass die darauf folgenden Produktcodes zu ignorieren sind, lautet *zu*. Empirisch lässt sich nachweisen, dass nach den Wörtern *für* und *zu* in nahezu keinem Fall ein relevanter Produktcode steht. Der zu durchsuchende Text kann dementsprechend erheblich verkürzt werden. Das Durchsuchen der Beschreibung eines Angebots nach einem Produktcode ist ebenfalls als wenig aussichtsreich einzustufen. Die Wahrscheinlichkeit ist relativ gering einen Produktcode in der Beschreibung zu finden, der nicht bereits im Titel auffindbar ist. Es besteht vielmehr die Gefahr, die Qualität des Ergebnisses zu verschlechtern, denn die Beschreibung weist eine erheblich höhere Heterogenität auf als der Titel, besitzt viel mehr Bezüge zu anderen Produkten und nicht zuletzt ist die Textlänge signifikant höher, was sich negativ auf das Laufzeitverhalten auswirkt.

Das Beispiel aus Abb. Abb. 8 besitzt ebenfalls drei Produktcodes, wobei keiner davon das angebotene Produkt identifiziert, da es sich um ein Zubehörteil für diese Produkte handelt. Hier werden die falschen Codes weder durch ein Wort wie *für* oder *zu* eingeleitet noch ist der Versuch den ersten Produktcode auszuwählen zielführend. Allerdings kann folgende Beobachtung gemacht werden: Die falschen Produktcodes bilden eine Sequenz und sind zudem sehr ähnlich zueinander. Je mehr Codes ähnlich zueinander bzw. an einer Sequenz beteiligt sind, desto unwahrscheinlicher ist es, dass einer dieser Codes der gesuchte Code ist.

Abb. 8: Becker GPS Antenne

Abb. 9 zeigt den Titel eines Produktangebots in dem nur ein einziger Produktcode vorkommt. Dieser lautet *KDL-40 E 4020* und ist auch der gesuchte Code des angebotenen Produkts. Nichtsdestotrotz kann es für ein automatisches Verfahren durchaus schwierig sein, hier die korrekten Grenzen des Produktcodes überhaupt zu erkennen. Es gibt viele Möglichkeiten, zum Beispiel: *KDL-40*, *KDL-40 E*, *KDL-40 E 4020* oder auch *KDL-40 E 4020 40*.

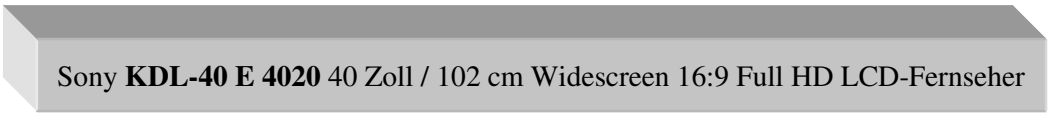

Abb. 9: Sony KDL-40 E 4020

Im Rahmen dieser Arbeit wird daher der Ansatz verfolgt, vor der Suche nach Produktcodes leichter und eindeutiger zu identifizierende Informationen zu entfernen. Solche Informationen sind beispielsweise Größenangaben, Stoppwörter sowie Begriffe, welche in vielen Angeboten unterschiedlicher Produkte verwendet werden. Abb. 10 verdeutlicht die Reduzierung des ursprünglichen Titels durch die Entfernung der beiden Größenangaben und des vielfach verwendeten Begriffs *16:9*.

Abb. 10: Sony KDL-40 E 4020 (vorverarbeitet)

2.2.4 Eigenschaften

Wie bereits erwähnt, können in einem Produktangebot sowohl produktspezifische als auch angebotsspezifische Eigenschaften enthalten sein. Ein Angebot könnte zum Beispiel mehrere Artikel eines Produkts beinhalten und so wenig vergleichbar sein mit einem zweiten Angebot, welches nur ein einziges Exemplar enthält. Da im vorangegangen Abschnitt das Entfernen von leichter zu identifizierenden Informationen als zweckdienlich klassifiziert wurde, ist es ohne Probleme möglich einen Teil dieser Informationen vor dem Entfernen in einem zusätzlichen Attribut zu speichern, um dadurch die Performance des Produkt-Matching zu verbessern. Nützliche Informationen sind

beispielsweise die erwähnte Verkaufseinheit, die Farbe eines Produkts oder, im Falle von Elektronikprodukten, technische Charakteristika, wie die Speicherkapazität, Belichtungszeiten, Temperaturbereiche usw. Diese technischen Eigenschaften treten im Text häufig als sogenannte physikalische Größenangaben auf und weisen daher eine Maßeinheit bzw. häufiger ein Einheitenzeichen auf. Als Referenz kann zu diesem Zweck das Internationale Einheitensystem [B06] herangezogen werden, in welchem eine Vielzahl an relevanten Maßeinheiten aufgeführt ist. Es gibt noch andere technische Charakteristika, die aber weniger gut dazu geeignet sind ein Produkt zu identifizieren und somit nicht unbedingt extrahiert werden müssen: Eigenschaften, wie zum Beispiel *16:9*, welche gemeinsam von mehreren Herstellern verwendet werden.

2.3 Zusammenfassung

Die Tab. 2 fasst die Ergebnisse der Analyse kurz zusammen. Es werden die diskutierten Attribute mit deren Eigenschaften aufgelistet und die jeweils festgestellten Maßnahmen, welche im Rahmen der Vorverarbeitung durchgeführt werden sollten, angegeben.

Attribut	Sauber?	Vollständig?	Einfach?	Durchzuführende Maßnahmen
Titel	nein	ja	nein	Extraktion von relevanten Informationen: - Produktcode - angebotsspezifische Eigenschaften - produktspezifische Eigenschaften
Beschreibung	nein	nein	nein	Extraktion von relevanten Informationen: - angebotsspezifische Eigenschaften - produktspezifische Eigenschaften
Preis	weitestgehend	ja	ja	keine Maßnahmen nötig
Kategorie	nein	ja	ja	Strategie findet sich in [KRT07]
Händler	ja	ja	ja	keine Maßnahmen nötig
Hersteller	nein	nein	ja	- leere Werte ersetzen - Schreibfehler korrigieren - semantisch äquivalente Werte identifizieren und zusammenführen

Tab. 2: Zusammenfassung der Analyse

3 Strategien zur Vorverarbeitung

In diesem Kapitel sollen Strategien zur Vorverarbeitung erarbeitet und vorgestellt werden, welche auf den Erkenntnissen der Analyse aus dem Kapitel 2 aufbauen. Der Fokus liegt auf der Ermittlung und Ergänzung von fehlenden Angaben des Herstellers und der Identifikation und Extraktion von Produktcodes der angebotenen Produkte. Als wichtigste Datenquellen werden für diesen Zweck die zusammengesetzten Attribute Titel und Beschreibung herangezogen. Um aus diesen Attributen gezielt Informationen extrahieren zu können, werden die textuellen Werte von Titel und Beschreibung zunächst so aufbereitet, dass die Extraktion von Produktcodes und Herstellerbezeichnungen sowohl effektiver als auch effizienter wird.

3.1 Vorbereitung von Titel und Beschreibung

Die entwickelte Strategie beruht auf der Ermittlung von elementaren Informationseinheiten bzw. der Grenzen von elementaren Informationseinheiten. Ziel ist es, möglichst viele Fixpunkte zu erhalten, über welche sich im Idealfall keine für das Preprocessing relevante Information erstreckt. Insbesondere soll damit die Identifikation der Grenzen von enthaltenen Produktcodes erleichtert werden.

Hierfür eignen sich zunächst hervorragend Satzzeichen. Diese müssen aber differenziert betrachtet werden. Zum Beispiel ist ein Bindestrich am Ende eines Wortes anders zu werten als ein Punkt, Komma, Ausrufezeichen, Schrägstrich oder Semikolon. Die zuletzt genannten Zeichen können in einem solchen Fall – entgegen einem Bindestrich – stets als solche Fixpunkte angesehen und, wie Abb. 11 zeigt, als Grenze zwischen elementaren Informationseinheiten interpretiert werden. Eine Ausnahme bildet der Punkt, wenn er nach einem einzeln stehenden Buchstaben oder einer Ziffer steht. In diesem Zusammenhang handelt es sich meist um einen Abkürzungspunkt oder um einen Ordnungspunkt. In beiden Fällen wird auf eine Klassifizierung als Fixpunkt verzichtet. Insbesondere hat dies den Hintergrund, dass die Abkürzung *f.* des Wortes *für* sowohl bei der Extraktion von Eigenschaften im Abschnitt 3.1.2 als auch bei der Ermittlung von leeren Werten des Herstellerattributs im Abschnitt 3.3.2 noch von Bedeutung ist.

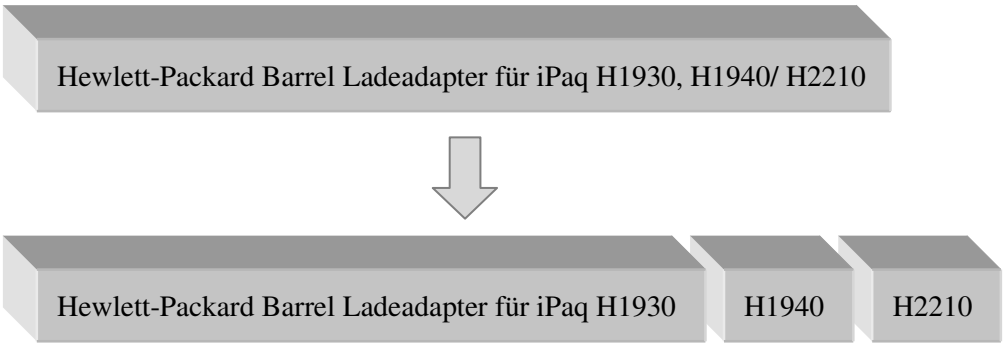

Abb. 11: Satzzeichen am Ende eines Wortes

Noch einmal auf Abb. 11 bezugnehmend, kommt es gelegentlich auch vor, dass das Leerzeichen nach einem solchen dort abgebildeten Komma fehlt, welches eigentlich zwei Produktcodes voneinander trennen soll – ein Komma ist aber praktisch nie Bestandteil eines Produktcodes. Somit ist die Interpretation eines Kommas, welches inmitten eines Wortes steht, als Grenze gerechtfertigt. Berücksichtigen muss man dabei aber, dass ein Komma zwischen zwei Ziffern keinen geeigneten Fixpunkt darstellt.

In Produktangeboten werden Sinneinheiten häufig durch einzeln stehende, also durch Leerzeichen eingeschlossene Trennzeichen voneinander separiert. Hierfür werden vorwiegend Interpunktionszeichen gewählt. Grundsätzlich *jedes,* durch Leerzeichen eingeschlossene Interpunktionszeichen als Grenze zu interpretieren wäre aber zu restriktiv, denn Interpunktionszeichen sind häufig auch Bestandteile von Produktcodes oder Teil einer technischen Eigenschaft, wie zum Beispiel der Bindestrich in *6 - 7 cm* oder der Schrägstrich in *9,5 / 11,8 / 3,5 cm*. Darüber hinaus ist dem Umstand gerecht zu werden, dass Händler in einem Angebot prinzipiell *jedes* Zeichen als Trennzeichen für Sinneinheiten verwenden können. Daher ist es sinnvoll, Zeichen für jedes Angebot zu ermitteln, welche, wie beispielsweise der Bindestrich in Abb. 12, *mehrfach* umgeben von Leerzeichen verwendet werden. Die Vorkommen genau solcher Zeichen können dann für das jeweilige Angebot als Grenze für das Preprocessing herangezogen werden. Hierfür werden ein Schwellwert für Interpunktions- und andere Sonderzeichen, und ein Schwellwert für alphanumerische Zeichen eingeführt. Ein zu geringer Schwellwert birgt eine hohe Wahrscheinlichkeit dafür, dass es sich bei dem Zeichen beispielsweise doch um den Teil eines bzw. mehrerer unterschiedlicher Produktcodes handelt. Da dies insbesondere auf alphanumerische Zeichen zutrifft, liegt der Schwellwert für Sonderzeichen unter dem für Buchstaben und Zahlen. Gute Ergebnisse erzielen beispielsweise ein Schwellwert von 3 für Sonderzeichen und ein Wert von 4 für Buchstaben und Zahlen.

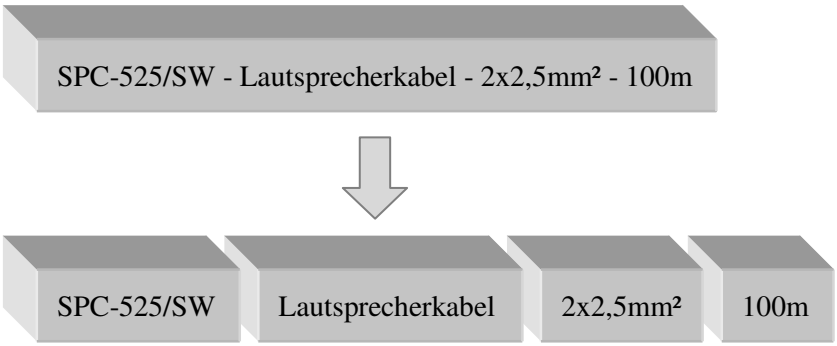

Abb. 12: Einzeln stehende Zeichen als Trennzeichen

Im Gegensatz dazu benötigt das Auftreten einer von Leerzeichen umgebenen Folge von zwei oder mehr identischen Sonderzeichen keine Mindesthäufigkeit. Solche Doppel- bzw. Mehrfachzeichen können ohne Einschränkungen als Grenze interpretiert werden, wie es beispielsweise in Abb. 13 mit dem Doppelzeichen ++ durchgeführt wird.

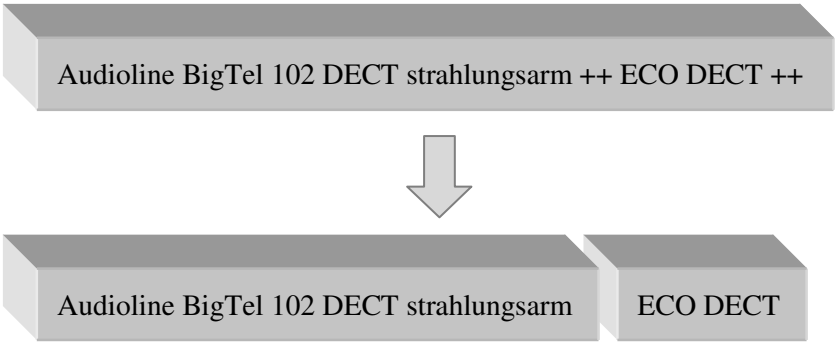

Abb. 13: Folge von identischen Sonderzeichen

Da Produktcodes und auch andere, im Rahmen des Preprocessing zu extrahierende Informationen, fast nie Anführungszeichen enthalten, aber vielfach in Anführungszeichen eingeschlossen werden und dabei vereinzelt – meist versehentlich wie in Abb. 14 dargestellt – das Leerzeichen vor den Anführungsstrichen bzw. nach den Ausführungsstrichen fehlt, wird das Anführungszeichen generell als Fixpunkt gedeutet.

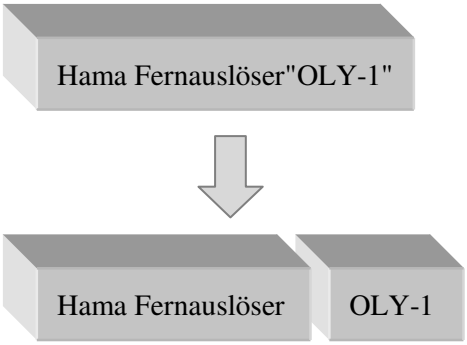

Abb. 14: Anführungszeichen schließen Produktcode ein

Außerdem werden runde, eckige und geschweifte Klammern am Anfang und am Ende eines Wortes als Grenze zwischen elementaren Informationseinheiten gewertet.

Wie am Ende von Kapitel 2.2.3 dargestellt, ist es für das Erkennen der Grenzen eines Produktcodes ebenso hilfreich, einfacher zu identifizierende Informationen zu entfernen und an deren Stelle Fixpunkte zu setzen. Verfahren zur Identifikation und Extraktion dieser Informationen werden in den folgenden Abschnitten 3.1.1 und 3.1.2 vorgestellt. Vor der Anwendung dieser Verfahren werden die deutschen Umlaute *ä*, *ö*, *ü* und *ß* normalisiert, das heißt durch ihre zweibuchstabigen Darstellungen *ae*, *oe*, *ue* und *ss* ersetzt.

3.1.1 Extraktion der Verkaufseinheit

Die Verkaufseinheit ist die wichtigste und zugleich häufigste angebotsspezifische Eigenschaft. Diese gibt an wie viele Artikel des gleichen Produkts zusammen angeboten und verkauft werden. Ein Angebot mit einer Verkaufseinheit von beispielsweise 5 unterscheidet sich demnach bedeutend von einem Angebot bei dem nur ein einziger Artikel verkauft werden soll, die Verkaufseinheit also 1 beträgt, obwohl die restlichen Angaben des Angebots unter Umständen weitestgehend identisch sind. Es handelt sich folglich um eine entscheidende Information im Rahmen des Produkt-Matching.

Um die Verkaufseinheit eines Angebots zu bestimmen, wird der Titel bzw. die Beschreibung nach gewissen Schlüsselwörtern durchsucht. Im Einzelnen sind dies die Wörter: *Stück*, *Stck*, *Stk*, *St.*, *Packs*, *Packungen*, *Set*, *Pack* und *Paar*. Da das *ü* bereits durch *ue* ersetzt wurde und auch der Abkürzungspunkt von *St.* zuvor entfernt wurde und an dessen Stelle eine Grenze zwischen elementaren Informationseinheiten getreten ist, wird genauer gesagt nach *Stueck* statt *Stück* gesucht und statt nach *St.* zu suchen, wird nach einem *St* unmittelbar vor einem Fixpunkt gesucht. Die Groß- und Kleinschreibung wird bei keinem der Schlüsselwörter beachtet. Unterschieden werden muss, ob es sich bei dem Angebot wirklich nur um *ein* Stück bzw. ein Paar, also *zwei* Stück handelt oder ob vor dem gefundenen Schlüsselwort noch eine Zahl steht. Set und Pack sind nur in einem solchen Fall verwertbar und müssen zudem noch das Suffix *er* am Ende der Zahl aufweisen, wie zum Beispiel in: *5er-Pack* oder *10-er Set*.

Für die auf diese Weise ermittelte Anzahl wird ein neues Attribut mit dem Namen *Verkaufseinheit* angelegt. Die Anzahl an Artikeln wird zusammen mit dem Schlüsselwort aus dem ursprünglichen Attribut entfernt. Im Beispiel in Abb. 15 wird eine Verkaufs-

einheit von 288 ermittelt und entsprechend als separater Attributwert gespeichert. Die komplette Zeichenkette *288 Stueck* kann daraufhin entfernt werden. Ohne diese Entfernung hätte *VE=288* fälschlicherweise sehr leicht als Produktcode identifiziert werden können.

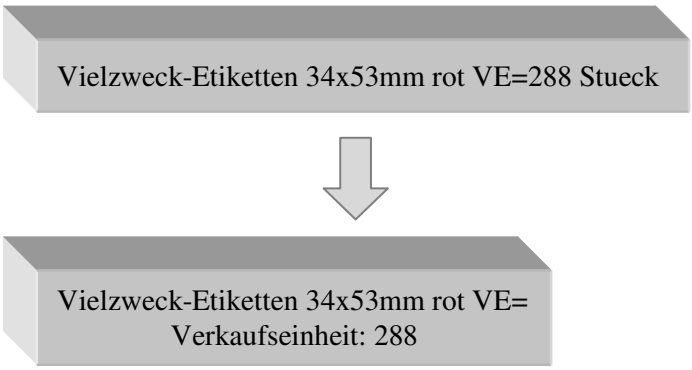

Abb. 15: Extraktion der Verkaufseinheit

3.1.2 Extraktion von Produkteigenschaften

Als wichtige Produkteigenschaften haben sich im Rahmen des Produkt-Matching die Farbe, technische Eigenschaften, wie die Längen- oder Geschwindigkeitsangabe, und Angaben über die Verwendungsmöglichkeiten des angebotenen Produkts herauskristallisiert. Mit Verwendungsmöglichkeiten sind Produkte gemeint, für die das angebotene Produkt verwendet werden kann oder geeignet ist. Diese Produkteigenschaften müssen zunächst im Titel und in der Beschreibung eines Angebots identifiziert und im Anschluss extrahiert sowie als Werte von eigenständigen Attributen zur Verfügung gestellt werden.

Da Farbnamen keinerlei syntaktische Auffälligkeiten besitzen, ist für die Identifikation eines solchen ein manuell annotierter Textkorpus nötig, also eine möglichst große Sammlung von Texten, in denen auftretende Bezeichnungen für Farben gekennzeichnet sind. Alternativ kann auch eine umfangreiche Liste möglicher Farben bereitgestellt werden. Jeder Eintrag einer solchen Liste muss dann mit den Wörtern des Titels und der Beschreibung der Angebote verglichen werden. Um eine hohe Verarbeitungsgeschwin-

digkeit sicherzustellen, ist es hier unumgänglich die Liste der Farben zuvor in einen PATRICIA-Trie[6] oder eine vergleichbar performante Datenstruktur umzuwandeln.

Die Identifikation von relevanten technischen Eigenschaften erfolgt, wie im Abschnitt 2.2.4 dargestellt, über Maßeinheiten sowie deren Einheitenzeichen. Es gibt gewisse Basiseinheiten und daraus abgeleitete Einheiten. Das Internationale Einheitensystem, auch SI-Einheitensystem genannt, benennt die folgenden sieben Basiseinheiten mit dem jeweils entsprechenden Einheitenzeichen: Meter (m), Gramm (g), Sekunde (s), Ampere (A), Kelvin (K), Mol (mol) und Candela (cd) [B06]. Diese können prinzipiell beliebig miteinander kombiniert werden, um weitere Einheiten abzuleiten – manche dieser abgeleiteten Einheiten besitzen eigene Namen.

Sowohl die Basiseinheiten als auch abgeleitete Einheiten mit eigenem Namen können durch Präfixe, wie Milli (m), Kilo (k) oder Mega (M), variiert werden. Darüber hinaus werden teilweise in Produktangeboten Schreibweisen von Einheiten bzw. Variationen von Einheiten verwendet, welche nicht konform mit dem SI-Einheitensystem sind – auch diese müssen beim Durchsuchen von Titel und Beschreibung berücksichtigt werden.

Tab. 3 zeigt für Elektronikprodukte eine Zusammenstellung relevanter Maßeinheiten, gruppiert nach der physikalischen Größe, die sie repräsentieren. Teilweise basieren diese Einheiten auf dem Internationalen Einheitensystem, teilweise sind sie aber auch gezielt aus beispielhaften Produktangeboten ermittelt. Nicht bei jeder Maßeinheit ist es konstruktiv, Präfixe, wie µ, m, k, M und G, zur Variation zuzulassen, und nicht jede der abgebildeten Einheiten erlaubt es in Kombination mit einer anderen Einheit, neue Maßeinheiten, wie zum Beispiel Nm oder m/s², abzuleiten. Das Vorhandensein dieser beiden Eigenschaften geht ebenfalls aus der Tabelle hervor.

[6] PATRICIA ist ein Akronym und steht für *Practical Algorithm to Retrieve Information Coded in Alphanumeric*. Mit Hilfe eines PATRICIA-Tries ist es möglich eine beliebige Menge an Wörtern kompakt abzuspeichern und in linearer Zeit festzustellen, ob ein gegebenes Wort in dieser Wortmenge vorkommt oder nicht.

Physikalische Größe	Einheit / Einheitenzeichen	Präfixe zugelassen?	Kombinierbar?	Physikalische Größe	Einheit / Einheitenzeichen	Präfixe zugelassen?	Kombinierbar?
Länge	m	ja	ja	Datenmenge	Byte	ja	ja
	Meter	nein	ja		byte	ja	ja
	inch	nein	nein		B	ja	ja
	Zoll	nein	nein		bit	ja	ja
Masse	g	ja	ja		bps	ja	nein
	t	ja	ja	Winkel	°	nein	ja
Kraft	N	ja	ja	Zeit	s	ja	ja
Frequenz	HZ	ja	ja		min	nein	ja
Energie	J	ja	ja		h	nein	ja
Leistung	Watt	ja	ja		d	nein	ja
	W	ja	ja		a	nein	ja
	Wmax	nein	nein	Temperatur	K	ja	ja
	Wrms	nein	nein		°C	nein	ja
Stromstärke	A	ja	ja	Fläche	Pixel	nein	nein
El. Spannung	V	ja	ja		Megapixel	nein	nein
El. Widerstand	Ohm	nein	ja		Mega Pixel	nein	nein
Lichtstärke	cd	ja	ja		px	nein	nein
Lichtstrom	lm	ja	ja		Mpix	nein	nein
Beleuchtungs-stärke	lx	ja	ja		mpix	nein	nein
					MP	nein	nein
Druck	Pa	ja	ja	Volumen	Liter	nein	ja
	bar	ja	ja		l	ja	ja
Schalldruck	dB	nein	ja				

Tab. 3: Zusammenstellung relevanter Maßeinheiten für Elektronikprodukte

Zu einer Maßeinheit gehört stets eine Maßzahl und diese muss – abgesehen von einem ggfs. vorhandenen Leerzeichen oder einem Bindestrich – unmittelbar vor der Maßeinheit stehen. Um auch kompliziertere Ausdrücke, wie zum Beispiel *9,5 / 11,8 / 3,5 cm*, welche mehrere durch Trennzeichen separierte Zahlen enthalten, extrahieren zu können, werden alle Zeichenketten akzeptiert, die durch die folgende Grammatik G erzeugt werden können. Die Terminale von G sind die Zahlen von 0 bis 9, das Komma, der Punkt, der Schrägstrich, der Bindestrich, das Leerzeichen sowie die Buchstaben x und X, die Nichtterminale sind die Elemente der Menge {S, M, Z, Z_0, V, D, T, L} und das Startzeichen von G ist S. G hat zudem die folgenden Regeln:

S	→	M|MTM|MTMTM
M	→	Z|VDZ_0
Z	→	ZZ_0|1|2|3|4|5|6|7|8|9
Z_0	→	Z_0Z_0|0|1|2|3|4|5|6|7|8|9
V	→	Z|0
D	→	,|.
T	→	LT|TL|LTL|/|x|X|-
L	→	␣

Das Beispiel in Abb. 16 zeigt ein Problem dieses Verfahrens auf: Fälschlicherweise wird ein Teil eines Produktcodes als physikalische Größe identifiziert.

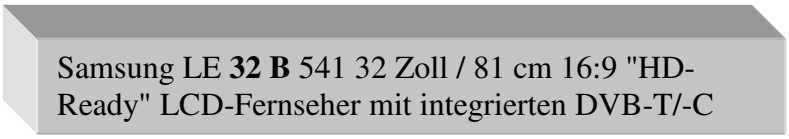

Abb. 16: Teil des Produktcodes stellt vermeintlich eine physikalische Größe dar

Aus diesem Grund wird auf die Extraktion und Entfernung der vermeintlichen physikalischen Größe verzichtet, wenn sich vor der Maßzahl ein Großbuchstabe befindet, da es sich in diesem Fall wahrscheinlich um einen falschen Treffer handelt. Leerzeichenseparierte Teile eines Produktcodes, auf die eine Ziffer folgt, enden viel häufiger mit einem Großbuchstaben als mit einem Kleinbuchstaben oder einer Zahl. Andererseits befindet sich bei einer wirklichen physikalischen Größe vor der Maßzahl wahrscheinlicher ein Sonderzeichen, wie beispielsweise ein Doppelpunkt, oder ein Wort der Alltagssprache, welches am Ende ausschließlich Kleinbuchstaben aufweist.

Abb. 17 zeigt beispielhaft die Extraktion sowohl einer Farbbezeichnung als auch einer Größenangabe.

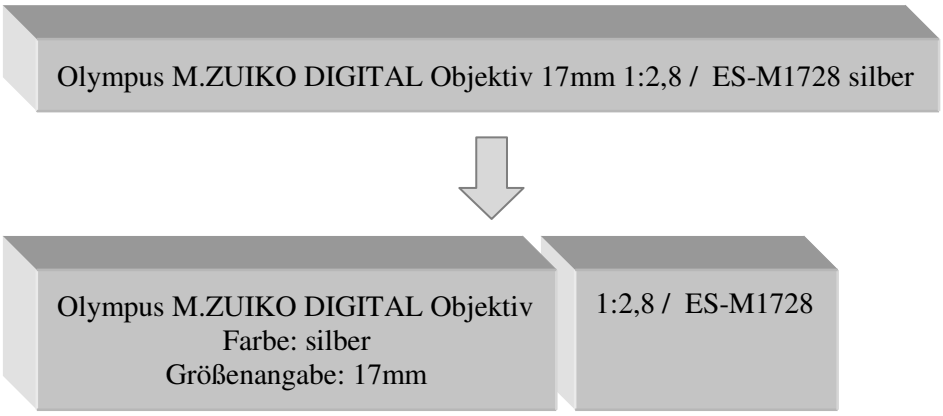

Abb. 17: Extraktion einer Farbe und einer technischen Eigenschaft

In Abschnitt 2.2.3 findet sich eine umfassende Analyse von potentiellen Störfaktoren, welche während der Produktcodeidentifikation und -extraktion auftreten können. Es ist demnach unerlässlich die Verwendungsmöglichkeiten eines Produkts vorher zu entfernen. Es lassen sich dafür zwei der dort genannten Auffälligkeiten effizient verwenden: Zum einen, dass es sich bei Sequenzen von Produktcodes fast immer um Verwendungsmöglichkeiten handelt und zum anderen, dass nach gewissen Schlüsselwörtern auftretende Codes ebenfalls Verwendungsmöglichkeiten sind.

Zu diesem Zweck eignet sich ein simpler Ansatz, bei dem eine leerzeichenbegrenzte Zeichenkette genau dann als Produktcode angesehen wird, wenn sie mindestens eine Ziffer, jedoch keine Leerzeichen enthält, kein Interpunktionszeichen am Ende aufweist und eine Mindestlänge von 2 besitzt. Eine Ausnahme bildet eine Zeichenkette, die ausschließlich am Anfang eine Ziffer aufweist – diese wird nicht als Produktcode angesehen.

Als Produktcodesequenz wird eine Folge von mindestens drei unmittelbar aufeinanderfolgenden elementaren Informationseinheiten bezeichnet, deren Folgenglieder jeweils einen Produktcode darstellen und somit keine Leerzeichen enthalten. Solche Sequenzen sind leicht identifizierbar und werden extrahiert.

Die zweite Auffälligkeit wird genutzt, indem zunächst nach einem Schlüsselwort gesucht wird. Als Schlüsselwörter gelten hierbei: *fuer*, *for*, *f.* und *zu*. Der Text zwischen dem gefundenen Schlüsselwort und dem darauf folgenden Fixpunkt wird auf das Vorkommen eines Produktcodes hin untersucht. Falls ein Produktcode gefunden werden

kann, wird der gesamte Text extrahiert. Falls der Text keinen Produktcode enthält, wird der Text nicht extrahiert, sondern nur entfernt.

Wie während der Analyse festgestellt, wird nach einer Verwendungsmöglichkeit in der Regel kein für das angebotene Produkt relevanter Produktcode genannt – ebenso ist ein relevanter Herstellername nicht zu erwarten. Dementsprechend können die elementaren Informationseinheiten, welche auf die extrahierte bzw. die entfernte Zeichenkette folgen, ebenfalls aus dem Titel bzw. der Beschreibung entfernt werden. Auf diese Weise wird der Suchraum für das in Abschnitt 3.2 vorgestellte Verfahren erheblich reduziert. Diese Vorgehensweise ist auch in Abb. 18 zu erkennen. Zunächst werden die Kommas zugunsten von Fixpunkten ersetzt und dann die Zeichenkette zwischen *für* und dem ersten darauf folgenden Fixpunkt extrahiert. Entfernt werden neben *für Canon PowerShot G5* zusätzlich die nachstehenden Informationseinheiten. Somit reduziert sich der Titel im Beispiel auf *Vikuiti DQC160*.

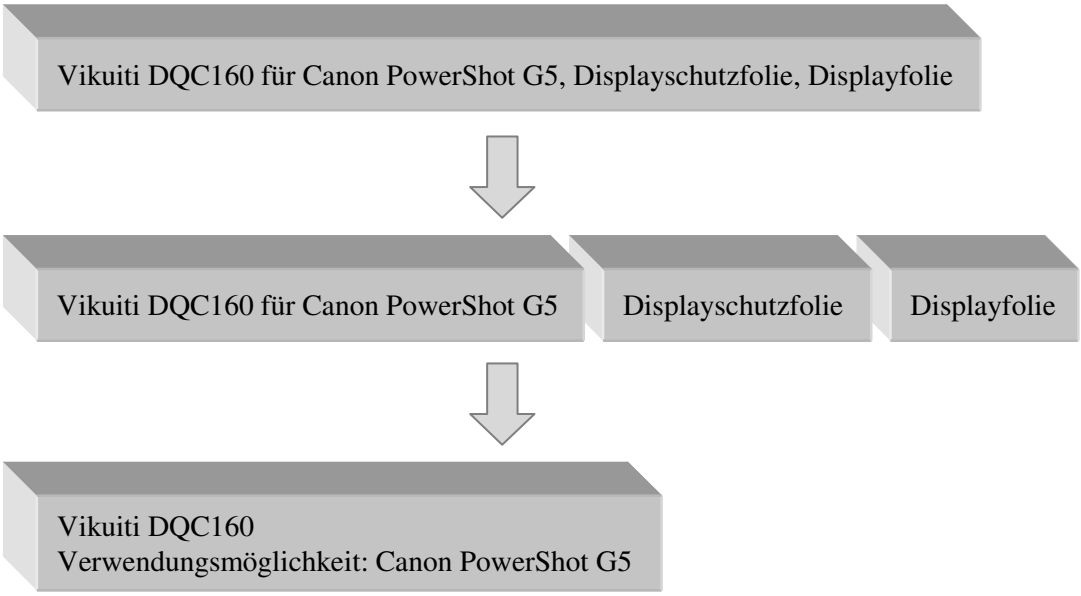

Abb. 18: Extraktion einer Verwendungsmöglichkeit

3.2 Extraktion von Produktcodes aus dem Titel

Die Identifikation und anschließende Extraktion von Produktcodes aus dem Titel des Angebots muss kontextbasiert erfolgen [T09]. Das im Rahmen der Bachelorarbeit entwickelte Verfahren identifiziert Produktcodes im Kontext jeweils eines Herstellers. Dies hat eine Reihe von Vorteilen, da viele ähnliche Produktcodes in diesem Kontext existieren und Produktcodes eines Herstellers signifikant seltener in den Angeboten von

Produkten anderer Hersteller vorkommen. Die Auswahl möglicher Produktcodes wird somit erleichtert und gleichzeitig kann die Anzahl der vermeintlichen Codes reduziert werden. Für die Verifikation der potentiellen Produktcodes wird im Abschnitt 3.2.3 der Kontext außerdem zusätzlich auf das Attribut *Kategorie* ausgedehnt.

3.2.1 Definitionen und Überblick

An dieser Stelle sei die Definition von *Produktcode* aus Abschnitt 2.2.3 wiederholt: Ein Produktcode (kurz: Code) ist eine vom Hersteller des Produkts gewählte Zeichenfolge zum Zweck der Produktidentifikation, welche meistens kein Wort des allgemeinen Sprachgebrauchs darstellt, sondern eine Aneinanderreihung von Buchstaben, Sonderzeichen und Ziffern ist, die zusätzlich durch ein oder mehrere Leerzeichen unterbrochen sein kann.

An die Produktcodedefinition lehnt sich die Definition des *Typs eines Produktcodes* (kurz: Produktcode-Typ oder Typ) an. Ähnlich wie in [T09] werden dabei die Zeichenklassen der einzelnen Zeichen des Produktcodes verwendet. Es werden Ziffern, Sonderzeichen und Buchstaben unterschieden, wobei Buchstaben zusätzlich nach Klein- und Großbuchstaben und Sonderzeichen nach Interpunktionszeichen und sonstigen Sonderzeichen differenziert werden. Aus dem Produktcode ergibt sich der Produktcode-Typ durch Ersetzen jedes Zeichens durch einen Repräsentanten für die jeweilige Zeichenklasse. Tab. 4 zeigt die hier verwendeten Ersetzungsregeln.

Zeichenklasse	Elemente der Zeichenklasse	Ersetzung
Kleinbuchstaben	a-z, ä,ö,ü,ß	1
Großbuchstaben	A-Z, Ä, Ö, Ü	2
Ziffern	0-9	3
Interpunktionszeichen	!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~	4
Sonstige Sonderzeichen	z.B.: ², ³	5
Leerzeichen	␣	␣

Tab. 4: Zeichenklassen und deren Ersetzung

Dadurch haben ähnliche Produktcodes den gleichen Typ und es kann zu einem Typ eine sehr große Anzahl an Codes geben. Diese Ähnlichkeit ist insbesondere im Kontext eines

einzigen Herstellers gegeben. Zur Verdeutlichung zeigt Tab. 5 einige Produktcodes verschiedener Hersteller zusammen mit dem jeweiligen Typ.

Hersteller	Produktcode	Produktcode-Typ
Sony	NWZ-A826P	222423332
Sony	NWZ-A826S	222423332
Blackberry	HDW-13019	222433333
Dicota	CamPocketLens	2112111112111
Samsung	LE 40 B 530 P7 WXZG	22 33 2 333 23 2222
Phillips	PET 723/12	222 333433
Phillips	42PES0001D	3322233332
Phillips	37PFL7603H	3322233332

Tab. 5: Auswahl einiger Produktcodes mit jeweils zugehörigem Typ

Ein *Produktcode-Kandidat* (kurz: Kandidat oder Code-Kandidat) sei zunächst jede im Titel vorkommende Zeichenkette, die formal die syntaktische Definition eines Produktcodes erfüllt, also aus Buchstaben, Sonderzeichen, Ziffern und Leerzeichen besteht. Diese Definition ist absichtlich so weit gefasst, um vorerst jeden potentiellen Produktcode auch erfassen zu können. Im weiteren Verlauf soll diese Definition natürlich eingeschränkt werden.

Eine *Produktcode-Kandidatenmenge* (kurz: Kandidatenmenge) ist eine Menge von Code-Kandidaten innerhalb eines Kontextes. Dieser Kontext kann ein einzelnes Angebot sein, alle Angebote von Produkten eines Herstellers umfassen oder gänzlich alle Produktangebote einschließen.

Sei K eine Kandidatenmenge. Dann ist jeder Typ eines Codes aus K ein *Typ-Kandidat* für K. Die Anzahl der Typ-Kandidaten für K ist regelmäßig signifikant kleiner als die Anzahl der Codes in K, da K meist viele ähnliche Codes enthält. Diese Beziehung wird auch umgekehrt verwendet, das heißt ein Typ-Kandidat hat innerhalb von K mehrere assoziierte Code-Kandidaten.

Auf Basis dieser Definitionen wird im Folgenden die prinzipielle Verfahrensweise bei der Extraktion von Produktcodes erläutert. Das Extrahieren des jeweiligen Produktcodes

aus den Angeboten läuft in separaten Schritten ab. Zunächst wird der Titel jedes Angebots nach potentiellen Produktcodes durchsucht und die vielversprechendsten Kandidaten ermittelt. Diese einzelnen Kandidatenmengen werden pro Hersteller zusammengefasst und anschließend per Internetsuchanfragen verifiziert. Nur bestätigte, nicht aber abgelehnte Kandidaten, können am Ende als Produktcode einem Angebot zugeordnet werden.

3.2.2 Ermittlung von Kandidaten eines Angebots

Der Kandidat-Begriff wurde sehr unpräzise eingeführt. Daher müssen nun vielversprechende Kandidaten ermittelt und die übrigen verworfen werden. Da sich laut Definition ein Produktcode auch über sogenannte Wortgrenzen hinweg erstrecken kann, schlägt ein simples Tokenisieren, das heißt ein Zerlegen der Eingabezeichenkette an Leerzeichen, und separates Weiterverarbeiten jedes einzelnen Token fehl. Es ist somit eine komplexere Herangehensweise gefragt. Insbesondere eignet sich ein Ansatz, bei dem Regeln für zu akzeptierende Typ-Kandidaten formuliert werden. Diese Regeln können domänenspezifisch angepasst werden.

Zunächst werden Stoppwörter sowie weitere Wörter bzw. Token, welche signifikant häufig in Angeboten von Produkten mehrerer, unterschiedlicher Hersteller vorkommen, aus dem Titel entfernt. Im Rahmen dieser Arbeit wird dazu eine Strategie angewandt, welche kontextbasiert für jede Kategorie die Verteilungshäufigkeit aller vorkommenden Token unter den Angeboten der verschiedenen Hersteller bestimmt. Mehrmaliges Auftreten innerhalb eines Angebots bleibt dabei unberücksichtigt, um die Verteilung nicht durch Mehrfachnennungen negativ zu beeinflussen. Ebenfalls unberücksichtigt bleiben Token, die ausschließlich aus Ziffern bestehen und Token, welche nicht mindestens bei drei unterschiedlichen Herstellern vorkommen.

Sei also K die betrachtete Kategorie, sowie A die Menge der Angebote und H die Menge aller Hersteller innerhalb von K. Aus dem Titel eines Angebots $a \in A$, mit $h_j \in H$ als Wert für das Herstellerattribut, wird jedes Token t_i entfernt, für das gilt:

$$\frac{\mathrm{freq_{i,j}}}{\sum_{k=1}^{|H|} \mathrm{freq_{i,k}}} < 0{,}5.$$

Dabei bezeichne $freq_{i,j}$ die Anzahl der Angebote von Produkten des Herstellers h_j, welche das Token t_i enthalten. Das heißt: Aus den Angeboten von Produkten eines Herstellers werden genau die Token entfernt, deren jeweilige relative Häufigkeit unter den Angeboten aller Hersteller in der betrachteten Kategorie nicht mindestens 50% beträgt. Es wird zudem berücksichtigt, dass sich kein Produktcode über das entfernte Token erstrecken kann.

Die Menge aller Produktcodes stellt im weiteren Sinne eine eigene Sprache dar – genauer gesagt eine formale Sprache. Diese Sprache gilt es zu formulieren. Dazu gibt es mehrere Möglichkeiten, wie zum Beispiel die Angabe einer formalen Grammatik oder eines regulären Ausdrucks sowie eine abschließende Liste aller Wörter der Sprache. Letzteres eignet sich hier aber nicht, um vollautomatisch Produktcodes zu extrahieren, denn dazu wäre es nötig eine solche Liste zu generieren. Dies ist jedoch maximal semi-automatisch durchführbar; außerdem ist es vermutlich gar nicht möglich eine vollständige Liste aller Codes aufzustellen, denn täglich werden neue Produkte mit jeweils neuen Produktcodes veröffentlicht. Diese unbekannten Codes zu erfassen erweist sich als schwierig. Konstruktiver ist die Formulierung einer Grammatik oder eines regulären Ausdrucks. Grammatiken weisen zwar eine größere Ausdrucksmächtigkeit auf, aber reguläre Ausdrücke sind für diese Sprache ausreichend mächtig. Aus diesem Grund und zusammen mit den Tatsachen, dass für reguläre Ausdrücke bereits zahlreiche Implementierungen in vielen Programmiersprachen existieren und sich reguläre Ausdrücke in aller Regel sehr kompakt notieren lassen, wird auf die Betrachtung von Grammatiken nicht weiter eingegangen.

Die Kandidatenmenge eines Angebots kann demnach durch reguläre Ausdrücke eingeschränkt werden. Es ist sicherlich nicht optimal, erst alle Kandidaten aus dem Titel zu ermitteln, um dann mit Hilfe der regulären Ausdrücke eine Eingrenzung auf wirklich vielversprechende Kandidaten vorzunehmen, denn wie bereits dargestellt, ist beinahe jede Teilzeichenkette ein Kandidat. Stattdessen sollte die Suche nach den vielversprechenden Kandidaten direkt im Titel unter Verwendung der regulären Ausdrücke durchgeführt werden. Aufgrund dessen, dass Produktcode-Typen analog zu Produktcodes ebenfalls eine eigenständige formale Sprache bilden, ist die Suche nach vorkommenden Typ-Kandidaten der unmittelbaren Suche nach Code-Kandidaten vorzuziehen. Die regulären Ausdrücke werden auf diese Weise kompakter und die Suche mit Hilfe

der kompakteren Ausdrücke effizienter, da weniger Vergleiche durchgeführt werden müssen.

Hierzu ist es erforderlich, die Ersetzungsregeln aus Tab. 4, in Analogie zur Bestimmung eines Produktcode-Typs, auf den Titel anzuwenden. Auf diesem, in gewisser Weise kodierten Titel, werden dann die Suchvorgänge mit den regulären Ausdrücken durchgeführt. Da durch die Ersetzungen stets genau ein Zeichen durch genau ein anderes ersetzt wird, ist ein Rückschluss auf die jeweiligen Code-Kandidaten ohne weiteres möglich. Dafür müssen nur die Anfangs- und die Endpositionen der gefundenen Typ-Kandidaten im kodierten Titel auf den ursprünglichen Titel übertragen werden.

Im Rahmen dieser Bachelorarbeit sind die folgenden regulären Ausdrücke[7] entstanden, die zusammen die Sprache der Produktcode-Typen für Elektronikprodukte darstellen. Für andere Produktdomänen sind ggfs. Anpassungen oder Erweiterungen nötig:

(1)
```
(?<=\b)(?:[123]{1,2} )*
[123][1234]*3(?:[1234]*[123])?
(?: [123]{1,2})*(?=\b)
```

(2)
```
(?<=\b)(?:[12]+ )?(?:[12] )?
[123][1234]*3(?:[1234]*[123])?
(?: [12])?(?: [12]+)?(?=\b)
```

(3)
```
(?<=\b)[123]{2,}5(?=\b)
```

(4)
```
(?<=\b)2?(1+2+)+1*(?=\b)
```

Die Zeilenumbrüche sind nicht Teil der Ausdrücke, sondern erfüllen rein darstellerischen Zweck. Zur Darstellung wurden außerdem drei Farben verwendet: rot, blau und grün. Rot stellt den obligatorischen Teil des Produktcode-Typs dar, grün optionale Teile und blau die Bedingung, dass Produktcodes bzw. Produktcode-Typen inmitten eines Wortes weder anfangen noch enden dürfen. (1) und (2) sind sehr ähnlich zueinander, da der obligatorische Teil identisch ist. Dieser fordert, dass der Hauptteil des Kandidaten aus Buchstaben, Zahlen und Interpunktionszeichen besteht, mindestens eine Zahl enthält und keine Interpunktionszeichen am Anfang oder am Ende aufweist. Bei (1) können sich vor und nach dem Hauptteil beliebig viele Token aus Zahlen und Buchstaben befinden, solange die Länge dieser Token stets 2 nicht überschreitet. Bei (2) kann sich vor und nach dem Hauptteil jeweils ein beliebig langes Token aus Buchstaben befinden. Ein zusätzlicher Einschub eines einzelnen Buchstaben ist dabei ebenfalls zulässig. (3)

[7] Die Syntax ist [F07] entnommen.

ermöglicht die Identifizierung von Codes wie *Duo²*, welche aus Buchstaben und Zahlen bestehen können und am Ende ein sonstiges Sonderzeichen aufweisen. (4) ist darauf ausgerichtet Produktcodes zu extrahieren, welche zwar nur aus Buchstaben bestehen, aber in der Wortmitte mindestens einen Großbuchstaben aufweisen.

Eine weitere Einschränkung der Kandidatenmenge lässt sich erreichen, indem berücksichtigt wird, was im Rahmen der Analyse im Abschnitt 2.2.3 festgestellt wurde: Je mehr ähnliche Codes innerhalb eines Angebots auftreten, desto wahrscheinlicher ist es, dass keiner dieser Codes das angebotene Produkt bezeichnet. Das Auftreten von ähnlichen Codes lässt sich an dieser Stelle sehr einfach identifizieren. Dazu müssen nur die Typ-Kandidaten mit mehreren Code-Kandidaten betrachtet und diejenigen mit zu vielen Code-Kandidaten aussortiert werden. Dabei hat sich ein Schwellwert von 3 als praktikabel erwiesen, siehe Kapitel 4. Das heißt: Typ-Kandidaten mit maximal drei assoziierten Code-Kandidaten werden akzeptiert und alle anderen inkl. der zugehörigen Code-Kandidaten werden verworfen.

Sobald für alle Angebote von Produkten eines Herstellers die jeweilige Kandidatenmenge bestimmt ist, können diese Mengen vereinigt werden, um so weitere unwahrscheinliche Kandidaten entfernen zu können. Wie bereits festgestellt, existieren innerhalb des Kontextes eines Herstellers viele ähnliche Produktcodes, das heißt Codes, die den gleichen Typ aufweisen. Um nur Code-Kandidaten mit hoher Erfolgsaussicht weiterzuverfolgen, können Typ-Kandidaten mit einer sehr geringen Anzahl an Code-Kandidaten, die unter einem gewissen Schwellwert liegt, unberücksichtigt bleiben.

Die Vereinigung der Kandidatenmengen ist aus Effizienzgründen auch für die folgenden Schritte, also für die Überprüfung der Kandidaten und für die Zuweisung von Produktcodes notwendig, da die Kandidatenmengen der einzelnen Angebote nicht disjunkt sind, sondern einen sehr hohen Grad an Überlappung aufweisen.

3.2.3 Verifikation der Produktcodeeigenschaft von Kandidaten

Der entscheidendste Schritt bei der Produktcodeextraktion ist die Überprüfung der Kandidaten. Diese erfüllen zwar die syntaktische Definition eines Produktcodes, aber keineswegs ist gesichert, dass die Hersteller diese Zeichenfolgen explizit ihren Produkten zugewiesen haben oder dass sie Produkte identifizieren können.

Für diese Verifikation werden die beiden folgenden zentralen Annahmen getroffen und im Anschluss untermauert:

> *Die Websuche nach einem Produktcode eines Produkts von einem bestimmten Hersteller resultiert in Suchtreffern, die den Namen genau dieses Herstellers enthalten.*

> *Die Websuche nach etwas anderem als einem Produktcode ergibt keine überdurchschnittliche Anzahl an Suchtreffern, die den Namen eines spezifischen Herstellers enthalten.*

Die Suche nach einem beliebigen Begriff, dem sogenannten Suchterm, mit einer Internetsuchmaschine, wie Google oder Yahoo, resultiert natürlich in Treffern in Form von Webseiten, die eine möglichst hohe Relevanz aufweisen – das heißt in gewisser Korrelation mit dem Suchterm stehen – denn das ist die Aufgabe einer solchen Suchmaschine. Zum Ende des ersten Quartals 2010 gab es, nach Angaben von VeriSign, einem führenden Anbieter von Internetinfrastrukturdiensten, 193 Milliarden registrierte Internetdomainnamen [V10]. Auf jeder dieser Domain können sich unzählige Webseiten befinden. Aufgrund dieser Größe bzw. der hohen Anzahl an Webseiten im Internet, spiegeln die Treffer auf den höchsten Rängen auch die wahrscheinlichste Bedeutung bzw. die wahrscheinlichsten Bedeutungen des Suchterms wider. Es gibt insbesondere auch sehr viele Webseiten, welche durch die Nennung eines oder mehrerer Produktcodes für den betrachteten Verifikationsprozess geeignet sind. Die Gründe für eine solche Nennung sind zahlreich. Es kann sich bei dem Internetauftritt um eine Verkaufsplattform, auf welcher die zugehörigen Produkte angeboten und verkauft werden, oder um eine Preisvergleichsplattform handeln. Andere Internetseiten befassen sich mit Produkttests oder dem Bewerten von Produkten. Nicht zuletzt besitzt auch der Hersteller selbst in aller Regel eine Homepage, auf der die Produkte vorgestellt werden.

Da Produktcodes meist keine Wörter des normalen Sprachgebrauchs sind, weist der Code für ein Produkt eines spezifischen Herstellers – in Ermangelung anderer Bedeutungen – eine außer Konkurrenz stehend hohe Korrelation mit dem Herstellernamen auf, denn auf Internetseiten, auf denen das jeweilige Produkt angeboten, getestet, bewertet oder vorgestellt wird, findet sich fast immer auch eine Erwähnung des Herstellers.

Andererseits gibt es nur sehr wenige andere Begriffe, welche sowohl in Angebotstexten verwendet werden als auch eine hohe Korrelation mit einem bestimmten Hersteller aufweisen. Hierzu sei auf die Grenzen des Verfahrens im Abschnitt 3.2.5 verwiesen.

Die Korrelation eines Kandidaten mit dem Herstellernamen wird nun berechnet, indem nach Möglichkeit die ersten 10 Treffer herangezogen werden und davon der Anteil der Treffer bestimmt wird, die den Herstellernamen enthalten. Als Suchraum für den Herstellernamen wird aber nicht die komplette Webseite verwendet, sondern nur die Internetadresse, der Titel und eine kleine Zusammenfassung der Internetseite oder ein kurzer Auszug aus deren Inhalt. Diese Informationen werden direkt durch die Suchmaschine zur Verfügung gestellt. Somit ist keine zusätzliche Abfrage der einzelnen Suchtreffer notwendig. Die Korrelation nimmt dabei stets einen Wert zwischen 0 und 1 an, wobei 0 für keinerlei Korrelation und 1 für maximale Korrelation steht. Die Entscheidung, ob ein Kandidat ein Produktcode ist oder nicht, wird dann mit Hilfe eines Schwellwerts getroffen.

Abb. 19 zeigt den prinzipiellen Ablauf des Verfahrens. Aus Übersichtsgründen sind die Ergebnisse der Websuche extrem verkürzt dargestellt, ohne dabei wesentliche Aspekte zu verdrängen. Der zu verifizierende Kandidat lautet *DTR 220*. Dementsprechend wird eine Websuche mit dem Suchterm *"DTR220"* ausgeführt. Das Einschließen in Anführungszeichen ist notwendig, da nur Suchtreffer förderlich sind, die eine exakte Übereinstimmung des Produktcodes aufweisen. Außerdem werden für die Websuche zuvor die Leerzeichen aus dem Produktcode-Kandidaten entfernt. Ohne diese Maßnahme wäre es ansonsten möglich, beinahe beliebige Zeichenketten als Produktcode zu verifizieren. Die Ursache dafür liegt darin, dass Wörter, welche in den Angeboten unmittelbar vor oder nach einem Produktcode stehen, auch häufig auf den durchsuchten Internetseiten unmittelbar vor bzw. nach dem Produktcode stehen. Die ersten 10 Treffer werden anschließend auf Vorkommen der Zeichenkette *Phillips* untersucht. Selbstverständlich spielt Groß- und Kleinschreibung dabei keine Rolle und auch eine Mehrfachnennung innerhalb eines Treffers, wie es beispielsweise beim zweiten Suchtreffer der Fall ist, hat keinen Einfluss auf die Berechnung der Korrelation. Da 9 der 10 Ergebnisseiten den Hersteller enthalten, ergibt sich als Wert für das Korrelationsmaß: 0,9. *DTR 220* wird nun als Produktcode von Phillips eingeordnet – vorausgesetzt der Schwellwert für zu akzeptierende Kandidaten ist nicht größer als 0,9.

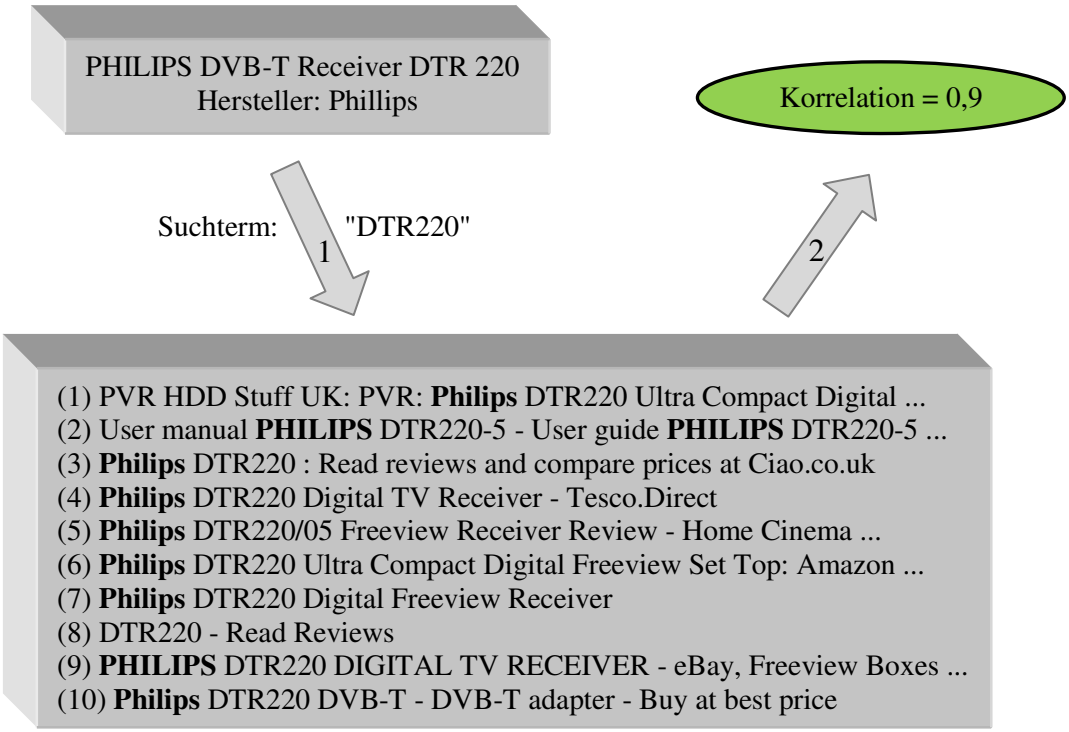

Abb. 19: Ablauf der Websuche

Ein bestimmter Produktcode wird zwar, wie bereits mehrfach angeführt, nur selten gleichzeitig von mehreren Herstellern zur Bezeichnung ihrer Produkte verwendet, aber bestenfalls ist in einem solchen Fall das Suchergebnis lediglich leicht verfälscht – im schlimmsten Fall suggeriert das Suchergebnis, dass es gar keine Korrelation zwischen einem der Hersteller und dem Produktcode gibt, obwohl diese vorhanden ist. Das gleiche trifft auf Codes zu, welche sehr wohl in der Alltagssprache verwendet werden bzw. noch weitere Bedeutungen tragen, wie es beispielsweise bei Codes, die ausschließlich aus Ziffern bestehen, der Fall ist. *2300* ist zum Beispiel der Produktcode eines Handys des bekannten Herstellers Nokia, dennoch ergibt die Websuche nach *"2300"* in Abb. 20 nur einen einzigen Treffer, welcher *Nokia* enthält. Um diese Mehrdeutigkeit zumindest teilweise zu kompensieren, bietet es sich an, die Suchanfrage zusätzlich auf die Kategorie auszudehnen. Es wird dann nicht nur nach dem Kandidaten gesucht, sondern nach dem Kandidaten zusammen mit der Kategorie. In dem konkreten Beispiel bedeutet dies, dass der Suchterm nun *"2300" Handys* ist. Aus Platzgründen seien auch hier wieder nur die Titel der Suchtreffer abgebildet. Die zweite Suchanfrage wird abgeschickt, weil die mit der ursprünglichen Websuche ermittelte Korrelation von 0,1 zu niedrig ist um *2300* als Produktcode des Herstellers Nokia bestätigen zu können. Im Gegensatz dazu erreicht die zweite Suchanfrage eine zufriedenstellende Korrelation zwischen *Nokia* und *2300* von 0,7.

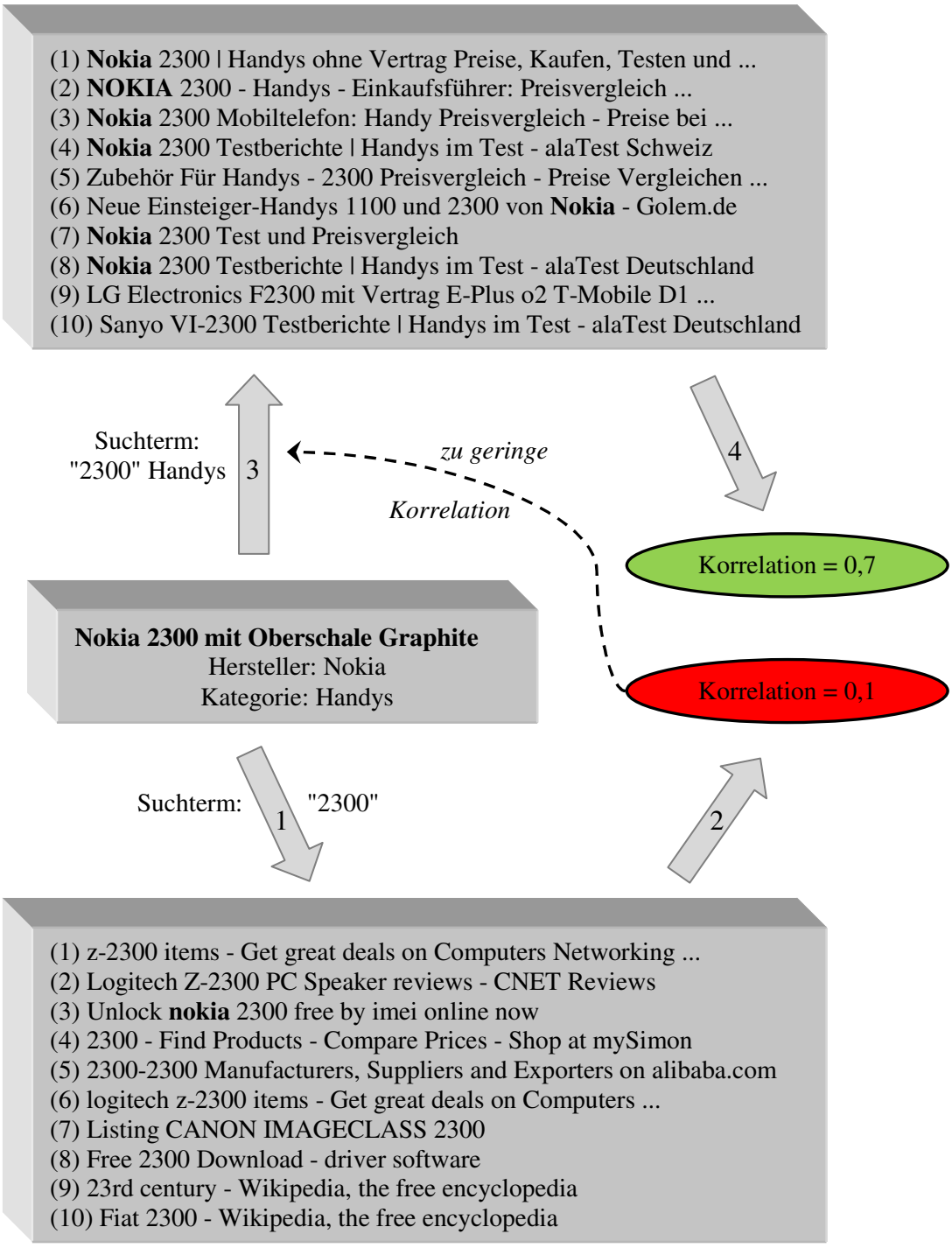

Abb. 20: Ablauf der Websuche bei mehrdeutigen Produktcodes

3.2.4 Zuweisen von Produktcodes

Den einzelnen Angeboten eines Herstellers müssen noch Produktcodes aus der Menge der bestätigten Codes zugewiesen werden. Das isolierte Zuweisen hat den Vorteil, dass unterschiedliche Schreibweisen von Codes durch die analoge Verwendung von Interpunktions- und Leerzeichen als optisches Trennelement ignoriert werden können. Es

findet also gleichzeitig eine gewisse Datenbereinigung statt. Wegen zu restriktiver Regeln selbst nicht als Kandidat erkannte Zeichenketten können so noch ermöglichen, dass der richtige Produktcode zugewiesen wird. Das heißt: In Angeboten von hoher Datenqualität werden Kandidaten identifiziert und als Produktcodes verifiziert, die im Anschluss auch bei Angeboten verwendet werden, deren Titelattribute eine geringere Qualität aufweisen und es daher nicht erlauben, eine direkte Extraktion vornehmen zu können. Auf diese Weise wird der Kontext des Herstellers noch einmal effektiv genutzt.

Um dies zu ermöglichen, werden zunächst alle verifizierten Produktcodes von jeglichen Leer- und Interpunktionszeichen befreit. Im Anschluss wird der Titel jedes Angebots noch einmal durchsucht, und zwar nach Vorkommen der bestätigten und bereinigten Produktcodes des jeweiligen Herstellers. Dabei werden Leer- und Interpunktionszeichen innerhalb des Titels ignoriert – nur an den Grenzen des Produktcodes muss sich jeweils ein Leerzeichen befinden, da sonst unerwünschte Übereinstimmungen auftreten. Falls mehrere verifizierte Codes gefunden werden, wird der im Titel zuerst genannte Produktcode gewählt. Stimmen die Anfänge zweier verifizierter Produktcodes überein und ist einer im anderen enthalten, also ein Fragment des anderen, gibt es möglicherweise mehr als einen erstgenannten Code im Titel. In einem solchen Fall wird der längere Code gewählt.

Angenommen es liegen für die beiden Hersteller Samsung und Nokia die bestätigten Produktcodes aus Tab. 6 vor. Durch Bereinigung dieser Codes fallen bereits die beiden Codes *6500c* und *6500-c* für Produkte von Nokia zusammen.

Hersteller	Produktcode	Bereinigter Produktcode
Nokia	6500 classic	6500classic
	6500	6500
	6500c	6500c
	6500-c	6500c
Samsung	32B541	32B541
	LE-32-B-541	LE32B541

Tab. 6: Bestätigte Produktcodes zweier Hersteller

Die Tab. 7 zeigt beispielhaft fünf Angebote, auf deren Titelattribut die im Kapitel 3.1 vorgestellten Verfahren angewendet wurden. In diesen Angeboten müssen die verifizierten und bereinigten Codes lokalisiert werden, wobei Leer- und Interpunktionszeichen in den Titeln zu ignorieren sind. Im Angebot mit der ID *A1* werden *6500classic* und *6500* gefunden. Beide beginnen an der gleichen Position im Titel, aber da erstgenannter Code länger ist als *6500*, wird dem Angebot der Produktcode *6500classic* zugewiesen. Im vierten Angebot werden ebenfalls 2 Codes lokalisiert: *32B541* und *LE32B541*. Zugewiesen wird der zuletzt genannte, da dieser im Titel weiter vorn beginnt als der wahrscheinlich nur ein Fragment darstellende Code *32B541*. Diese Vermutung wird durch das Angebot A5 bestätigt, in dem nur der Produktcode *LE32B541* gefunden werden kann, da *32B541* nicht von Leerzeichen umschlossen vorkommt.

ID	Titel	Hersteller	Lokalisierte verifizierte Codes
A1	Nokia 6500 classic black	Nokia	**6500classic**, 6500
A2	Nokia 6500-c - silber	Nokia	**6500c**
A3	NOKIA 6500 c 5 Tracks Silber	Nokia	6500, **6500c**
A4	Samsung LE 32 B 541 32 Zoll / 81 cm 16:9 "HD-Ready" LCD-Fernseher mit integrierten DVB-T/-C Digitaltuner	Samsung	32B541, **LE32B541**
A5	SAMSUNG LCD-Fernseher LE32B541 weiß 32 Zoll (82 cm) 16/9, "Full HD" -, HDMI x3	Samsung	**LE32B541**

Tab. 7: Angebote und ihnen zugewiesene verifizierte Produktcodes

Um auch bei einem Angebot ohne spezifizierten Hersteller in der Lage zu sein, einen Produktcode zuweisen zu können, ist es möglich auf die bestätigten Produktcodes *aller* Hersteller zurückzugreifen. Die Suche nach Vorkommen dieser Codes erfolgt ansonsten analog zur Suche in einem Titel eines Angebots mit nichtleerem Wert für das Herstellerattribut. Da durch die im Vorfeld durchgeführten Vorbereitungen des Titelattributs – insbesondere durch die Entfernung von Verwendungsmöglichkeiten – der größte Teil an falschen Kandidaten bereits eliminiert wurde, handelt es sich um einen vielversprechenden Ansatz, um mit fehlenden Werten des Herstellerattributs umzugehen.

Ein ähnliches Vorgehen ist auch bei Angeboten eines Herstellers möglich, bei denen kein einziger Kandidat verifiziert werden kann, denn das bedeutet, dass mit einer hohen

Wahrscheinlichkeit der angegebene Hersteller inkorrekt ist. In Kapitel 3.3.1 werden zwar Strategien erarbeitet, um falsche Herstellerbezeichnungen zu erkennen und für das Verifizieren in gewisser Hinsicht zu berichtigen, aber nichtsdestotrotz ist die Berichtigung eines völlig ungeeigneten Werts, wie zum Beispiel *Sonstige*, auch für die dort vorgestellten Verfahren nicht möglich. Solch ein inkorrekter Wert des Herstellerattributs wird genauso behandelt wie ein fehlender Wert.

3.2.5 Grenzen des Verfahrens

Das vorgestellte Verfahren funktioniert nicht bei allen möglichen Produktcodes. Wie bereits angedeutet, hat es Schwierigkeiten bei der Verifikation von Codes, die noch weitere Bedeutungen tragen oder gleichzeitig Produkte verschiedener Hersteller bezeichnen. Am Ende von Abschnitt 3.2.3 ist zwar eine Möglichkeit vorgestellt worden, um diese Mehrdeutigkeiten zu kompensieren, aber diese schlägt fehl, wenn sich die fraglichen Produkte auch in der Produktkategorie ähneln.

Eine weitere Schwachstelle sind firmenspezifische Verfahren oder Spezifikationen. Diese werden verständlicherweise oft in Angebotstexten genannt, um das jeweilige Produkt näher zu beschreiben und um es von Produkten anderer Hersteller abzugrenzen. Leider sind sie syntaktisch kaum von möglichen Produktcodes unterscheidbar und aufgrund der Firmenspezifik lässt sich natürlich auch aus der Websuche eine hohe Korrelation zum Herstellernamen ermitteln und so werden Zeichenketten als Produktcode verifiziert, die eigentlich gar keine Produktcodes sind.

Problematisch sind außerdem Codes, die ausschließlich aus Buchstaben bestehen und zudem keine Besonderheiten bzgl. Groß- und Kleinschreibung aufweisen. Diese in die Kandidatenmengen aufzunehmen ist kontraproduktiv. Zum einen würde dies die Kandidatenmengen um ein Vielfaches vergrößern, da dann auch jedes Wort des allgemeinen Sprachgebrauchs als möglicher Produktcode weiter untersucht werden müsste. Im gleichen Verhältnis würde folglich die Anzahl der notwendigen Internetsuchanfragen zunehmen und somit die Effizienz der Produktcodeextraktion signifikant sinken. Zum anderen würde dadurch die Wahrscheinlichkeit, auf firmenspezifische Begriffe zu stoßen, erheblich steigen. Dies wiederum würde die Effektivität der Produktcodeextraktion in größerem Maße senken als die zusätzlich gefundenen Produktcodes diese steigern könnten.

Ungeeignet für dieses Verfahren sind zudem Produkte, bei denen der Name des Herstellers noch weitere, verbreitete Bedeutungen trägt. So lässt sich für einen Hersteller mit dem Namen *Digital* beinahe jede Zeichenkette verifizieren, die mit Digitaltechnik in Bezug steht und daher in signifikanter Häufigkeit im Zusammenhang mit dem Wort *digital* im Internet genannt wird.

3.3 Herstellerattribut

Da die in Kapitel 3.2 vorgestellte Strategie zur Extraktion von Produktcodes enorm von der Qualität des Herstellerattributs, also von dessen Sauberkeit und Vollständigkeit abhängt, ist es essenziell, fehlende Herstellerangaben zu ergänzen und die bestehenden inkorrekten Werte des Herstellerattributs bestmöglich zu korrigieren. Zunächst wird hierzu eine grundsätzliche Vereinheitlichung vorgenommen. Dazu gehört es, alles in Groß- bzw. Kleinbuchstaben umzuwandeln, Umlaute durch ihre zweibuchstabigen Darstellungen zu ersetzen, und Interpunktionszeichen, wie den Bindestrich oder den Unterstrich, zu entfernen und stattdessen ein Leerzeichen einzufügen. Auch doppelte Interpunktions- bzw. Leerzeichen müssen dabei eliminiert werden. Der Vorteil dieser Normalisierung ist es, dass auf diese Weise bereits einige zuvor unterschiedliche Werte zusammenfallen, zumindest wird aber das Feststellen von synonymen Herstellerbezeichnungen erleichtert. Von dieser Tatsache profitiert auch die Extraktion von Herstellernamen in Abschnitt 3.3.2.

3.3.1 Matchen von Herstellernamen

Vor allem falsche oder ungebräuchliche Schreibweisen eines Herstellers sind problematisch. Diese erschweren die Verifikation der Produktcodes oder können diese sogar gänzlich verhindern, wenn die jeweilige Schreibweise im Zusammenhang mit dem zu verifizierenden Produktcode-Kandidaten zu selten im Internet zu finden ist. Im Vorfeld ist es demnach wichtig, unterschiedliche Bezeichnungen für ein und denselben Hersteller festzustellen. Man kann in diesem Fall auch von Objekt-Matching sprechen, wobei die Objekte nur ein einziges Attribut besitzen und zwar den Namen des Herstellers. Für Zeichenketten gibt es, wie im Einleitungskapitel bereits angeführt, eine Reihe von Ähnlichkeitsmaßen. In diesem Fall bietet sich ein Ähnlichkeitsmaß basierend auf der Levenstein-Distanz an. Die Levenstein-Distanz gibt für zwei gegebene Zeichenketten die Mindestanzahl der notwendigen elementaren Editiervorgänge an, um aus der einen Zei-

chenkette die jeweils andere zu erhalten. Als elementar gelten dabei die Operationen: Löschen und Hinzufügen eines Zeichens sowie das Ersetzen eines Zeichens durch ein anderes Zeichen. Für die im Rahmen der Evaluation in Kapitel 4 ermittelten Ergebnisse wird auf eine, auf die Länge der Zeichenketten normalisierte Version des Levenstein-Abstands zurückgegriffen und ein Schwellwert von 0,25 verwendet. Das heißt: Sobald nur drei von vier oder noch weniger Buchstaben übereinstimmen, werden die Hersteller nicht vereint.

Mit Hilfe einer solchen Distanzmetrik ist es aber im Grunde nicht möglich etwas anderes als bloße Schreibfehler festzustellen. Insbesondere größere syntaktische Abweichungen, wie sie zum Beispiel bei Abkürzungen oder bei grundsätzlich verschiedenen Schreibweisen auftreten, bleiben dabei unberücksichtigt.

Insbesondere für letzteres eignet sich ein simpler, aber effektiver Ansatz, bei dem zwei Herstellernamen gematcht werden, wenn einer der beiden Namen in dem anderen enthalten ist. Dadurch kommt es allerdings auch zu falschen Treffern und zwar dann, wenn ein Hersteller beispielsweise wiederum den Namen *Digital* trägt, welcher Teil völlig anderer Firmenbezeichnungen sein kann. Aber die negativen Folgen, die aus den gelegentlich auftretenden falschen Treffern resultieren, überwiegen keinesfalls die Vorteile der vielen Herstellernamen, die auf diese Weise gematcht werden können. Die Produktcodeverifikation profitiert schließlich immens von kurzen und prägnanten Werten für das Herstellerattribut und wenn das Unternehmen nicht gerade wirklich *Digital* heißt, also ein sehr verbreitetes Wort der Alltagssprache darstellt, sind kaum negative Folgen zu erwarten.

Eine weitere Strategie, um ebenfalls syntaktisch sehr unterschiedliche Herstellernamen zu matchen, ist es, für jeden solchen Namen eine Websuche zu starten und die ersten Treffer der Suchergebnisse – genauer gesagt die Uniform Resource Locator (URL) dieser Treffer – miteinander zu vergleichen. Bei Übereinstimmung der sogenannten Top-URL zweier Suchergebnisse werden die beiden Herstellernamen als semantisch äquivalent angesehen. Dieses Verfahren hat eine erheblich geringere Trefferquote als der zuvor erläuterte Ansatz, aber dafür ist es nahezu ausgeschlossen, dass auf diese Weise gematchte Namen nicht denselben Hersteller bezeichnen. Besonders typische Abkürzungen, wie *HP* für *Hewlett Packard,* werden auf diese Weise zuverlässig erkannt.

Die Entscheidung für genau einen der jeweils miteinander korrespondierenden Herstellernamen als alleinigen Vertreter für jede dieser Gruppen stellt ein sehr schwieriges Problem dar. Dieses ist unter dem Namen Datenfusion bekannt. Aufgrund der Komplexität und der Tatsache, dass Datenfusion in diesem Fall sogar kontraproduktiv sein kann, werden die korrespondierenden Herstellernamen in einem separaten Attribut lediglich gesammelt. Das Vorhalten vieler unterschiedlicher Schreibweisen schränkt zwar die Effizienz des Verifikationsprozesses ein wenig ein, aber dafür wird gerade die Effektivität erheblich gesteigert.

Das folgende Beispiel soll die hier vorgestellten Strategien verdeutlichen: Gegeben seien sechs Produktangebote *A1* bis *A6*. Für das Herstellerattribut besitzen *A1* und *A6* den Wert *Altec*, *A2* den Wert *Altec Lansing*, *A3* den Wert *Hewlett Packard*, *A4* den Wert *HewlettPeckard* und *A5* den Wert *HP*. Die fünf Herstellernamen müssen nun paarweise auf semantische Äquivalenz hin untersucht werden. Hierzu zeigt Tab. 8 zum einen die normalisierten Levenstein-Distanzen und zum anderen die Top-URLs aus den Internetsuchanfragen. Aber zunächst ist festzustellen, dass *Altec* ein Teil von *Altec Lansing* ist – diese beiden Werte werden daher als semantisch äquivalent angesehen.

	Altec Lansing	Hewlett Packard	HewlettPeckard	HP	Top-URL aus Websuche
Altec	0,61	0,80	0,71	1,00	http://www.altec.com/
Altec Lansing	-	0,80	0,93	1,00	http://www.alteclansing.com/
Hewlett Packard	-	-	0,13	0,87	http://www.hp.com/
HewlettPeckard	-	-	-	0,86	http://www.scribd.com/doc/9608419
HP	-	-	-	-	http://www.hp.com/

Tab. 8: Analyse der Herstellernamen

Aus den Levenstein-Distanzen ergibt sich, dass nur *Hewlett Packard* und *HewlettPeckard* ähnlich genug sind, um ebenfalls als semantisch äquivalent zu gelten, da die restlichen Abstände zu groß sind. Der Tab. 8 ist weiterhin zu entnehmen, dass *HP* und *Hewlett Packard* identische Top-URLs aufweisen. Somit bezeichnen auch diese Herstellernamen den gleichen Hersteller. Die so miteinander korrespondierenden Werte

des Herstellerattributs werden nun für jedes der sechs Angebote in einem neuen Attribut mit dem Namen *Hersteller_alternativ* zusammengeführt und die einzelnen Zeichenketten jeweils durch ein vorher festgelegtes Trennzeichen miteinander verbunden. Das endgültige Ergebnis ist in Tab. 9 zu sehen.

ID	Hersteller	Hersteller_alternativ
A1	Altec	Altec ; Altec Lansing
A2	Altec Lansing	Altec ; Altec Lansing
A3	Hewlett Packard	Hewlett Packard ; HewlettPackard ; HP
A4	HewlettPackard	Hewlett Packard ; HewlettPackard ; HP
A5	HP	Hewlett Packard ; HewlettPackard ; HP
A6	Altec	Altec ; Altec Lansing

Tab. 9: Ergebnis des Hersteller-Matching

3.3.2 Ersetzen von fehlenden Werten

Um bei einem Angebot mit leerem Wert für das Herstellerattribut einen korrekten Wert zu bestimmen, also den Hersteller des angebotenen Produkts zu ermitteln, kommen im Folgenden zwei Strategien zum Tragen. Die Erste kann und wird vor der Produktcodeextraktion durchgeführt und wirkt sich somit positiv auf diese aus. Die zweite Strategie beruht auf Basis der ermittelten Produktcodes.

Zunächst einmal lassen sich potentielle Herstellernamen im Titel und in der Beschreibung suchen und mit den bereits bekannten Herstellernamen der anderen Produktangebote vergleichen. Dazu werden alle Werte, die das Herstellerattribut annimmt, in den Attributen Titel und Beschreibung der Angebote gesucht, bei denen der Hersteller nicht explizit gegeben ist. Dieses Verfahren kann zwar theoretisch eine quadratische Laufzeit besitzen, aber da die Anzahl der unterschiedlichen Hersteller vergleichsweise gering ist, nimmt das Laufzeitverhalten mehr linearen Charakter in der Anzahl der Angebote an. Dabei muss natürlich darauf geachtet werden, dass der gefundene Hersteller auch wirklich das angebotene Produkt herstellt und nicht etwa nur deshalb genannt wird, weil das angebotene Produkt für Produkte genau dieses Herstellers geeignet ist oder dieser zusammen mit einem seiner Produkte angegeben wird, um das angebotene Produkt zu spezifizieren. Abb. 21 und Abb. 22 zeigen jeweils ein entsprechendes Beispiel.

LG ist ein bekanntes Unternehmen, aber nicht der Produzent des angebotenen Reiseladegeräts aus Abb. 21. Wendungen wie *passend zu* und *passend für,* oder wie in diesem Fall nur das Wort *für* unmittelbar vor der Herstellerbezeichnung, sind eindeutige Anzeichen dafür, den folgenden Herstellernamen nicht zu extrahieren. Auch Abkürzungen, wie zum Beispiel *f.*, müssen berücksichtigt werden. Durch die in Abschnitt 3.1.2 vorgestellten Strategien zur Vorbereitung von Titel und Beschreibung ist es an dieser Stelle nicht notwendig, weitere Maßnahmen zu ergreifen, denn entsprechende Informationseinheiten sind bereits aus den beiden Attributen entfernt.

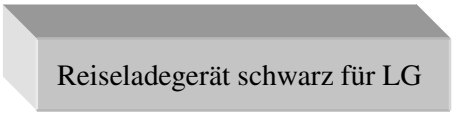

Abb. 21: Angebotenes Produkt ist für Produkte eines anderen Herstellers geeignet

Auch im zweiten Beispiel hat das genannte Unternehmen JVC das angebotene Produkt aus Abb. 22 nicht hergestellt, sondern soll in Einheit mit dem aufgeführten Produktcode nur klar stellen, dass es sich bei dem angebotenen Produkt um einen Akku handelt, der baugleich zum Akku von JVC mit dem Produktcode *BN-V607* ist. Wichtige Indizien sind hierfür Ausdrücke wie *kein Original* oder *nicht original*, welche an einer beliebigen Stelle im Titel oder in der Beschreibung stehen können. In diesen Fällen wird ebenfalls auf die Übernahme des Herstellernamens verzichtet.

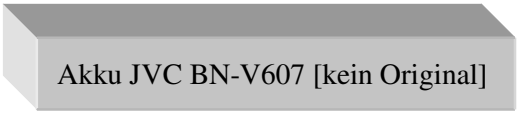

Abb. 22: Spezifikation mit Hilfe eines Produkts eines anderen Herstellers

Die zweite Strategie basiert auf der Feststellung, dass spezifische Produktcodes nur in wenigen Fällen von mehr als einem Hersteller verwendet werden. So lassen sich durch den Produktcode Rückschlüsse auf den Hersteller ziehen. Diese Strategie ist anwendbar bei Angeboten, bei denen sich aus Titel und Beschreibung zwar kein Herstellername, aber aus dem Titel dennoch ein Produktcode extrahieren lässt. Der Vergleich von diesem Code mit den Produktcodes aller anderen Angebote ermöglicht es, bei einer Übereinstimmung den Hersteller des jeweiligen Angebots zu übernehmen. Dies wird natürlich nur dann vollzogen, wenn der Code eindeutig einem Hersteller zuordenbar ist, aber nicht, wenn dieser bei Angeboten von Produkten unterschiedlicher Hersteller vorkommt.

4 Evaluation

Die im Kapitel 3 erarbeiteten Strategien zur Vorverarbeitung sollen in diesem Kapitel evaluiert werden. Zu diesem Zweck werden in den Abschnitten 4.1 und 4.2 Maße zur Bewertung der Ergebnisse eingeführt und die Testdaten vorgestellt. Für diese Testdaten wird in Abschnitt 4.3 die Qualität der extrahierten Produktcodes untersucht und in Abschnitt 4.4 der Einfluss der Vorbereitungsmaßnahmen diskutiert. Zum Schluss wird im letzten Abschnitt 4.5 noch ein kurzer Überblick gegeben und darin auf das Laufzeitverhalten und die Herstellerrecherche eingegangen.

4.1 Evaluationsmaße

Das Evaluieren der vorgestellten Vorverarbeitungsstrategien für das Matchen von Produktangeboten setzt entsprechende Evaluationsmaße voraus. Bekannte und etablierte Maße zur Bewertung eines Verfahrens sind Precision, Recall und F-Measure.

Ein Verfahren erzeugt Antworten bzgl. einer Eingabe. Eine solche Antwort kann entweder korrekt oder inkorrekt sein. Sei E eine Eingabe. Dann bezeichne N die Anzahl aller möglichen korrekten Antworten bzgl. E, A die Anzahl aller durch das Verfahren erzeugter Antworten bzgl. E, und A_t die Anzahl der davon korrekten Antworten.

Precision ist ein Maß, welches den Anteil der korrekt gegebenen Antworten des betrachteten Verfahrens bzgl. aller Antworten des Verfahrens angibt. Es gilt:

$$Precision = \frac{A_t}{A}$$

Recall ist ein Maß, welches den Anteil der korrekt gegebenen Antworten des betrachteten Verfahrens bzgl. aller möglichen korrekten Antworten angibt. Es gilt:

$$Recall = \frac{A_t}{N}$$

Sowohl *Precision* als auch *Recall* nehmen stets eine reelle Zahl zwischen 0 und 1 als Wert an. Im Idealfall haben beide den Wert 1, was bedeutet, dass jede gegebene Antwort korrekt ist und jede mögliche korrekte Antwort gegeben wurde. Normalerweise schließt ein hoher Wert für eines der beiden Maße einen hohen Wert für das jeweils andere aus. Aufgrund von diesem Zusammenhang, wird ein Maß benötigt, das mit

einem Wert die Gesamtqualität eines Verfahrens ausdrücken kann. Dies wird beispielsweise durch das F-Measure erreicht, welches das harmonische Mittel von Precision und Recall ist und somit gleichermaßen von beiden Werten abhängt. Es gilt:

$$F\text{-}Measure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

Auch das F-Measure nimmt als Wert stets eine reelle Zahl zwischen 0 und 1 an.

4.2 Testdaten und Testszenarios

Die Testdaten bestehen aus 102.182 Datensätzen und stellen Angebote mit einer großen Variation von Elektronikprodukten dar, wobei etwa 15% der angebotenen Produkte nur im weitesten Sinne Elektronikprodukte sind: Kameratasche, Displayschutzfolien usw. Die übrigen statistischen Daten können Tab. 10 entnommen werden. Es werden drei aus den Testdaten abgeleitete Szenarios betrachtet: *Vollständig*, *Handys* und *Zubehör*. Diese unterscheiden sich in ihrer Schwierigkeit, die Produktcodes und fehlende Hersteller korrekt zu ermitteln, erheblich voneinander. Das Szenario *Handys* weist beispielsweise wenige der Probleme aus Abschnitt 2.2.3 auf und die angebotenen Produkte haben größtenteils sehr kurze und einfache Produktcodes. Im Gegensatz dazu sind im Zubehörszenario die aus der Analyse bekannten Probleme omnipräsent. Außerdem wird in vielen Angeboten der Produktcode des angebotenen Produkts nicht genannt und teilweise sind die Codes syntaktisch sehr komplex, siehe Beispiel 4 in Abb. 6.

	Vollständig	Handys	Zubehör
Anzahl Angebote	102.182	5.693	52.518
Anzahl Herstellernamen	1.335	82	860
Anzahl Kategorien	71	1	11
Manuell mit Produktcode annotierte Angebote (Anteil Angebote mit Produktcode)	1.174 (78,3%)	761 (96,6 %)	576 (66,5%)
Anzahl /Anteil Angebote mit fehlendem Wert für Herstellerattribut	2.355 / 2,3%	8 / 0,0014%	636 / 1,2%

Tab. 10: Überblick über die Testdaten und daraus abgeleiteter Testszenarios

Die Werte für Precision, Recall und F-Measure werden in den folgenden Abschnitten stets für jedes der drei Testszenarios bestimmt. Dadurch soll verhindert werden, dass zufällig auftretende Unregelmäßigkeiten innerhalb eines Szenarios zur Überbewertung der Resultate führen.

4.3 Qualität der extrahierten Produktcodes

Um die erreichte Qualität der mit dem vorgestellten Verfahren extrahierten Produktcodes bewerten zu können, wird sie mit der Qualität der extrahierten Produktcodes eines bestehenden Ansatzes verglichen. Aus diesem Grund wird im Abschnitt 4.3.2 ein Baseline-Algorithmus und seine Funktionsweise vorgestellt, und im darauffolgenden Abschnitt ein Vergleich zwischen den Ergebnissen des Baseline-Algorithmus und des im Rahmen dieser Bachelorarbeit implementierten Algorithmus vorgenommen.

4.3.1 Bestimmung von Schwellwerten

Zunächst werden Schwellwerte bestimmt, mit denen die besten Ergebnisse hinsichtlich Precision und Recall bzw. hinsichtlich F-Measure zu erwarten sind. Hierzu werden die in Tab. 11 dargestellten, zu Testkomplexen zusammengefassten Testfälle betrachtet. Im Einzelnen sind dies die Komplexe *Websuche* (*W*), *Angebot* (*A*) und *Hersteller* (*H*). Die jeweils zugeordneten Testfälle unterscheiden sich in genau einem der drei in Tab. 11 angegebenen Parameter – die übrigen Parameter bleiben konstant. Der Testfall *R* steht als Referenz und wird in jedem der Testkomplexe mit einbezogen.

Testfall	Schwellwert für Produktcodeverifikation per Websuche (siehe 3.2.3)	Max. Anzahl an Code-Kandidaten pro Typ und Angebot (siehe 3.2.2)	Min. Anzahl an Code-Kandidaten pro Typ und Hersteller (siehe 3.2.2)
R	0,1	3	1
W0	0,0	3	1
W3	0,3	3	1
W5	0,5	3	1
W7	0,7	3	1
W9	0,9	3	1
W10	1,0	3	1
A1	0,1	1	1
A2	0,1	2	1
A4	0,1	4	1
A5	0,1	5	1
A+	0,1	∞	1
H2	0,1	3	2
H3	0,1	3	3

Tab. 11: Übersicht über Testfälle zur Bestimmung der Schwellwerte

Die Ergebnisse des Testkomplexes *W* sind in Abb. 23 dargestellt. Diese zeigen, dass bereits ein Schwellwert von 0,1 für die Verifikation von Produktcodes ausreichend ist. Die Precision ist mit 0,79 in *Vollständig,* 0,93 in *Handys* und 0,79 in *Zubehör* annähernd so gut wie bei Verwendung höherer Schwellwerte – für Schwellwerte größer als 0,7 wird die Precision sogar übertroffen. Im Gegenzug profitiert der Recall signifikant von einem wenig restriktiven Schwellwert. Dies ist insbesondere im Szenario *Vollständig* zu erkennen. Ein Schwellwert von 0 im Testfall *W0* bedeutet, dass *jeder* Code-Kandidat eines Herstellers automatisch ein Produktcode ist. Auffällig ist, dass bei diesem Testfall in allen drei Szenarios vor allem die Precision trotzdem hohe Werte aufweist. Zurückzuführen ist dies auf die umfassende Vorbereitung des Titelattributs, wodurch zum einen der Großteil potentiell falscher Kandidaten entfernt wird und zum anderen konsequent die Grenzen von Produktcodes festgestellt werden.

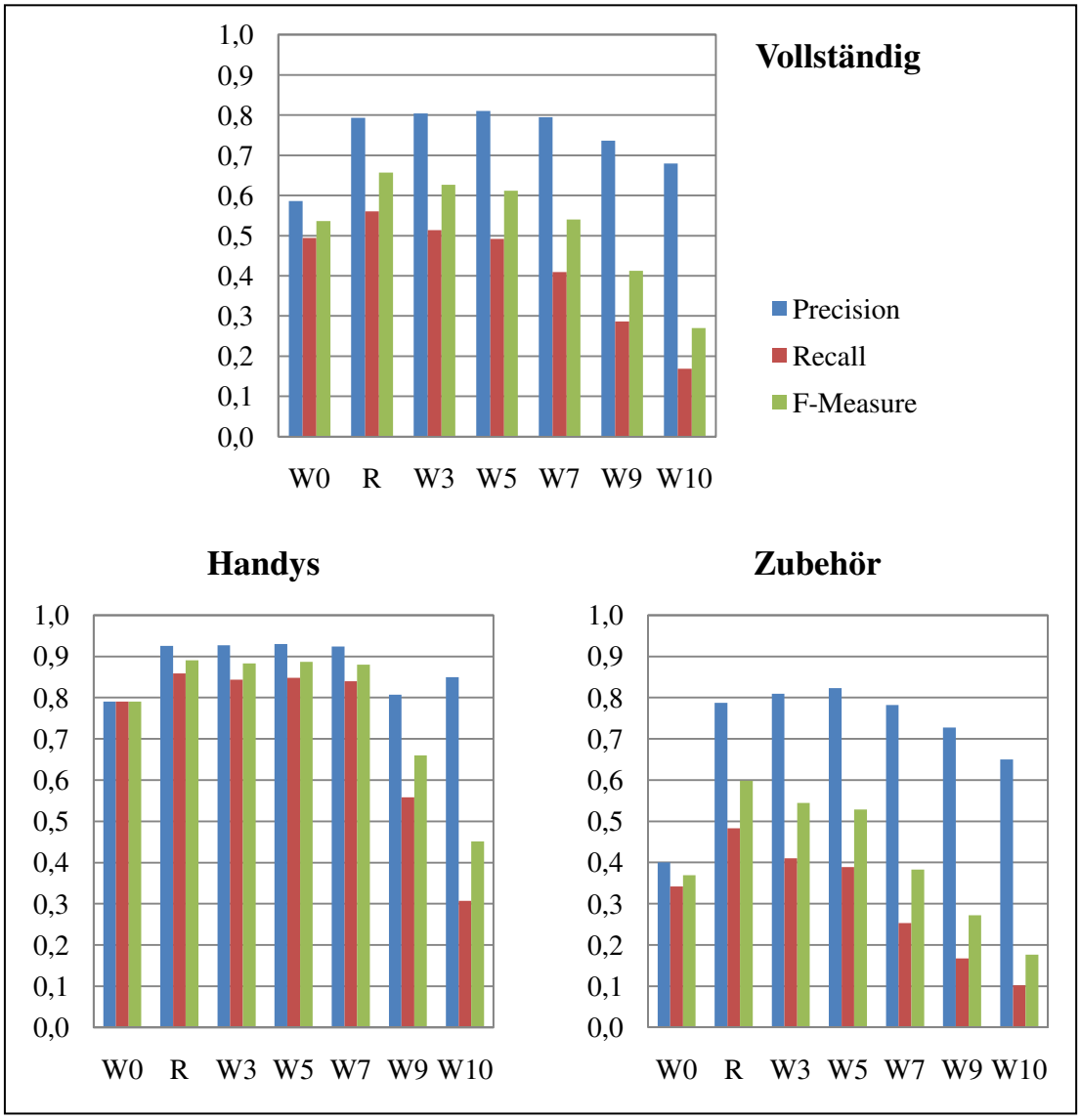

Abb. 23: Ergebnisse des Testkomplexes W

Die Resultate der Testfälle aus dem Komplex *A* sind in Abb. 24 abgebildet. Demnach führt ein Schwellwert von 3 für die maximal erlaubte Anzahl an unterschiedlichen Code-Kandidaten pro Typ und Angebot in allen drei Szenarios sowohl für Recall als auch für Precision zu geringfügig besseren Werten. Die Unterschiede fallen mit durchschnittlich 0,01 für die Precision und 0,02 für den Recall deshalb so minimal aus, da durch die konsequente Extraktion von Verwendungsmöglichkeiten nur noch wenige Angebote existieren, in denen mehrere ähnliche Code-Kandidaten zu finden sind.

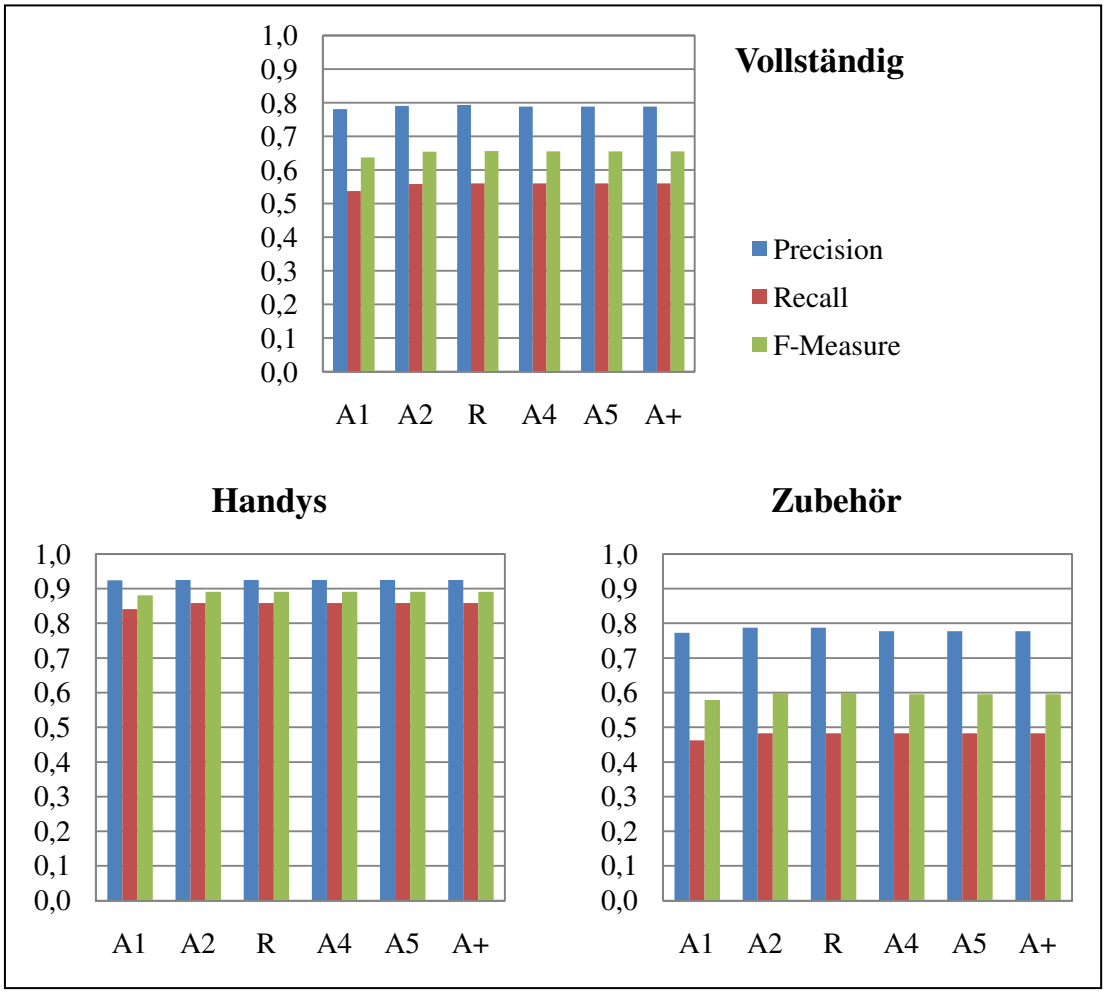

Abb. 24: Ergebnisse des Testkomplexes A

In Abb. 25 sind schließlich die Ergebnisse des Testkomplexes *H* dargestellt. Erneut bestätigt sich, dass die Vorbereitungen aus den Abschnitten 3.1 und 3.2.2 großen Einfluss auf die Auswirkungen des betrachteten Schwellwerts haben, denn obwohl der Recall in *H1* und *H2* im Vergleich zu *R* nachvollziehbar sinkt, bleibt die Precision in *Vollständig* und *Handys* relativ konstant bzw. sinkt in *Zubehör* sogar signifikant – zu erwarten wäre eigentlich ein Anstieg der Precision mit steigendem Schwellwert. Ein weiterer Grund für die Verringerung der Precision um 0,16 beim Übergang von *R* nach *H1* im Szenario

Zubehör ist, dass durch die Forderung der Existenz von ähnlichen Code-Kandidaten häufig nur noch Fragmente der syntaktisch komplexen Codes extrahiert werden.

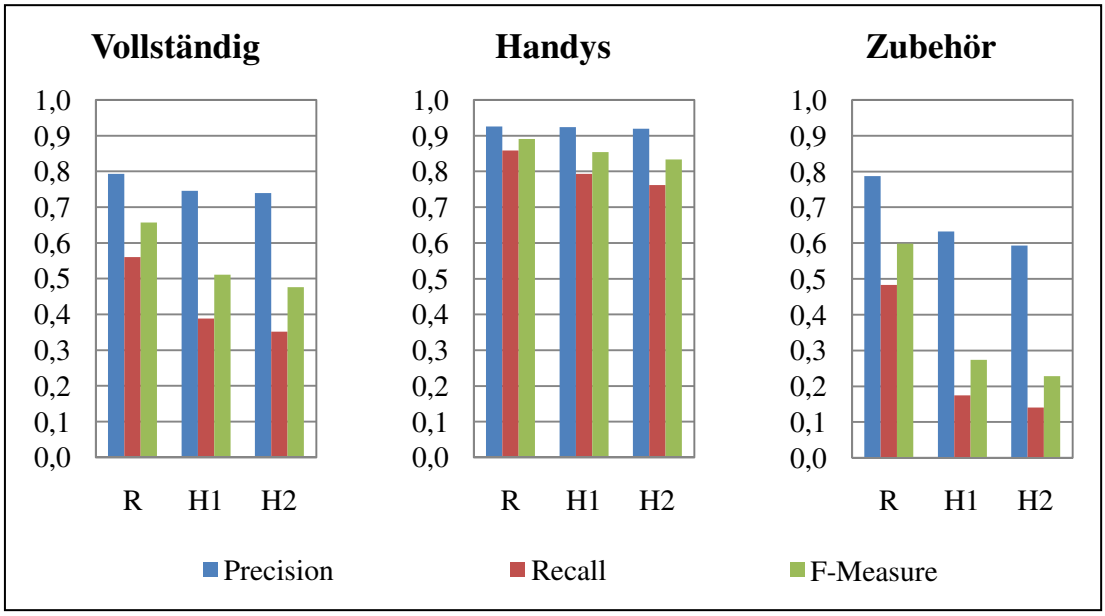

Abb. 25: Ergebnisse des Testkomplexes H

4.3.2 Baseline-Algorithmus

Der Baseline-Algorithmus zum Erkennen und Extrahieren von Produktcodes ist [T09] entnommen. Da das in dieser Bachelorarbeit vorgestellte Verfahren durch Weiterentwicklung aus dem Baseline-Algorithmus entstanden ist, wird bei diesem eine ähnliche Strategie verfolgt. Demnach ist ein Produktcode eine Kombination von Ziffern, Buchstaben und Sonderzeichen, wobei mindestens eine Ziffer enthalten sein und der Typ dieser Kombination mit dem mindestens einer anderen Kombination übereinstimmen muss. Die andere Kombination muss hierzu in einem weiteren Angebot eines Produkts desselben Herstellers vorkommen. Der Typ einer solchen Kombination bestimmt sich, indem Buchstaben durch eine 1, Ziffern durch eine 2 und Sonderzeichen durch eine 3 ersetzt werden. Im Vorfeld wird ebenso versucht, semantisch äquivalente Herstellernamen zu vereinen, um eine möglichst hohe Anzahl unterschiedlicher Kombinationen pro Herstellerkontext zu erhalten.

4.3.3 Vergleich mit Baseline-Algorithmus

Der Produktcodebegriff ist in dieser Arbeit weiter gefasst als in [T09], siehe 2.2.3. Daher ist es angemessen, neben der exakten Übereinstimmung der jeweils automatisch

extrahierten Produktcodes mit den manuell annotierten Produktcodes ebenfalls eine Betrachtung nur teilweiser Übereinstimmungen vorzunehmen, denn auch ein partieller Produktcode kann ein Produkt weitestgehend eindeutig identifizieren.

Wie in Abb. 26 zu erkennen ist, erreicht der Baseline-Algorithmus (*BL*) in keinem der Szenarios die Performance des im Rahmen dieser Arbeit entwickelten Algorithmus (*R*). Lediglich im vergleichsweise simplen Szenario *Handys* – und auch nur unter Einbezug der partiellen Treffer – ist es dem Baseline-Algorithmus möglich annähernd die Qualität von *R* zu erzielen. Ohne Berücksichtigung partieller Treffer erreicht *R* für *Vollständig* einen F-Measure von 0,66, für *Handys* einen F-Measure von 0,89 und für *Zubehör* einen F-Measure von 0,6. Somit liegt das F-Measure von *R* zwischen 43% und 133% über dem von *BL*.

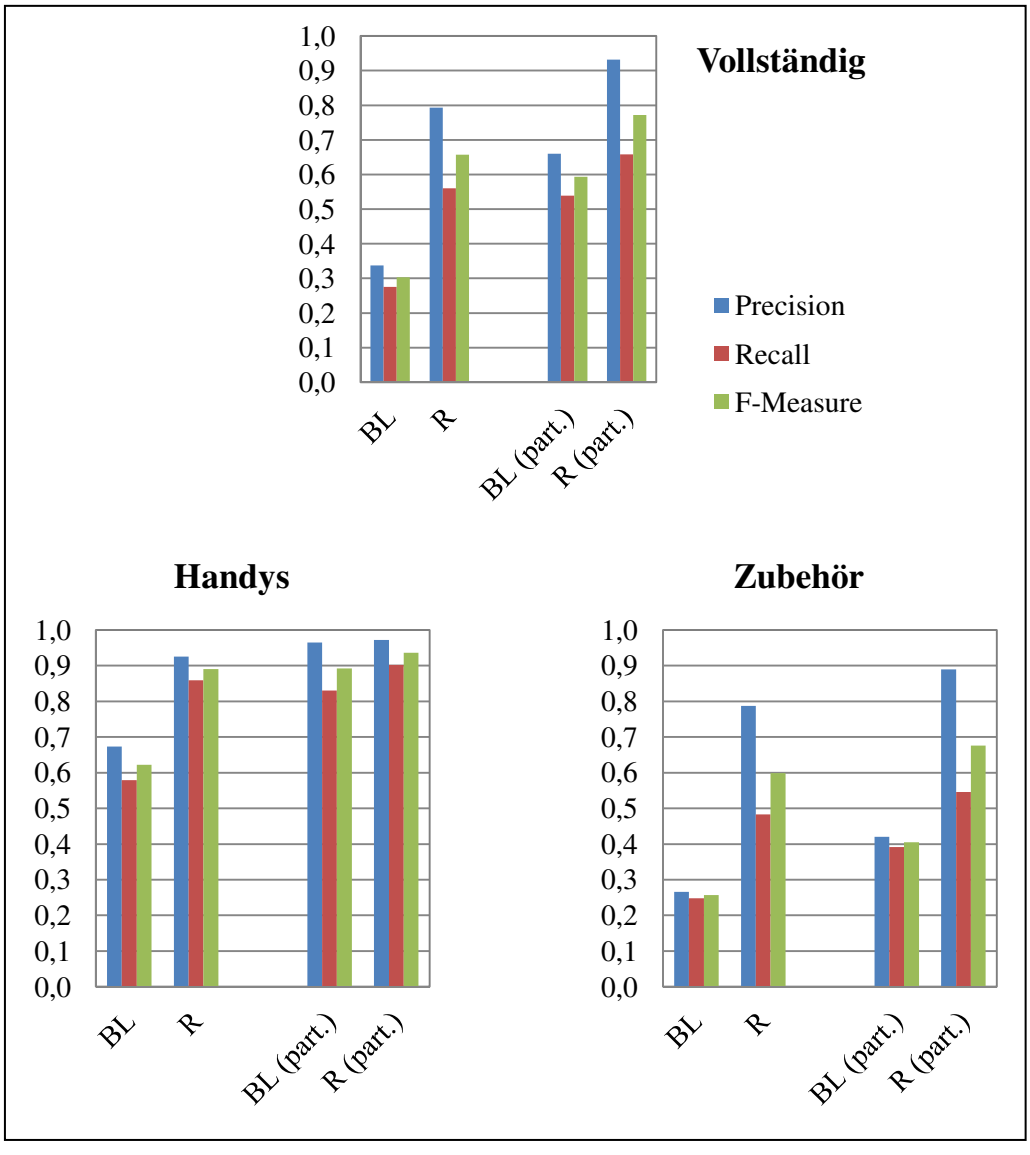

Abb. 26: Ergebnisse des Vergleichs zwischen Baseline und vorgestelltem Verfahren

4.4 Einfluss Vorbereitungsmaßnahmen

Es sollen nun die einzelnen Strategien zur Vorbereitung des Titelattributs evaluiert werden. Dies geschieht, indem der Einfluss der Vorbereitungsmaßnahmen auf die Qualität der Produktcodes, welche aus dem Titel extrahiert werden, untersucht wird. Zu diesem Zweck werden Testfälle abgeleitet, in denen einzelne Strategien nicht zum Tragen kommen, siehe Tab. 12. Ebenso wird untersucht welche Folgen ein nicht durchgeführtes Hersteller-Matching hat. Als Schwellwerte werden wieder die gewählt, welche im Abschnitt 4.3 die besten Ergebnisse geliefert haben, also denen des Testfalls *R* aus Tab. 11 entsprechen.

Testfall	Bestimmte Zeichen als Fixpunkt (siehe 3.1)	Extraktion von Eigenschaften (siehe 3.1.1 u. 3.1.2)	Entfernen von Stoppwörtern (siehe 3.2.2)	Entfernen von anderen häufig auftretenden Wörtern (siehe 3.2.2)	Matchen von Herstellern (siehe 3.3.1)
R	ja	ja	ja	ja	ja
V1	nein	ja	ja	ja	ja
V2	ja	nein	ja	ja	ja
V3	ja	ja	ja	nein	ja
V4	ja	ja	nein	nein	ja
V5	nein	nein	nein	nein	ja
V6	ja	ja	ja	ja	nein

Tab. 12: Übersicht über Testfälle zur Evaluation der Vorbereitungsmaßnahmen

Das Resultat der Evaluation ist in Abb. 27 dargestellt. Jede der Vorbereitungsmaßnahmen führt zu einem Anstieg des F-Measure. Vor allem der Testfall *V2* verdeutlicht durch den deutlichen Einbruch der Precision im Vergleich zu *R* den Einfluss der Maßnahmen. Insgesamt ergibt sich eine Steigerung des F-Measure um 0,05 in den Szenarios V*ollständig* und *Zubehör* bzw. um 0,1 im Szenario *Handys*.

Der geringe Einfluss des Hersteller-Matching auf die Qualität der Produktcodeextraktion lässt sich folgendermaßen erklären: Im Falle eines wenig verbreiteten oder falsch geschriebenen Herstellernamens kann meist kein Code-Kandidat verifiziert werden. In einem solchen Fall wird aber versucht, wie am Ende von Abschnitt 3.2.4 erläutert,

Produktcodes anderer Hersteller den Angeboten zuzuweisen – bei Vorhandensein einer gängigen Alternativbezeichnung werden also die negativen Konsequenzen kompensiert. Diese Zuweisungsart ist aber weniger effizient und birgt zudem die Gefahr, auch Codes völlig anderer Hersteller zu berücksichtigen.

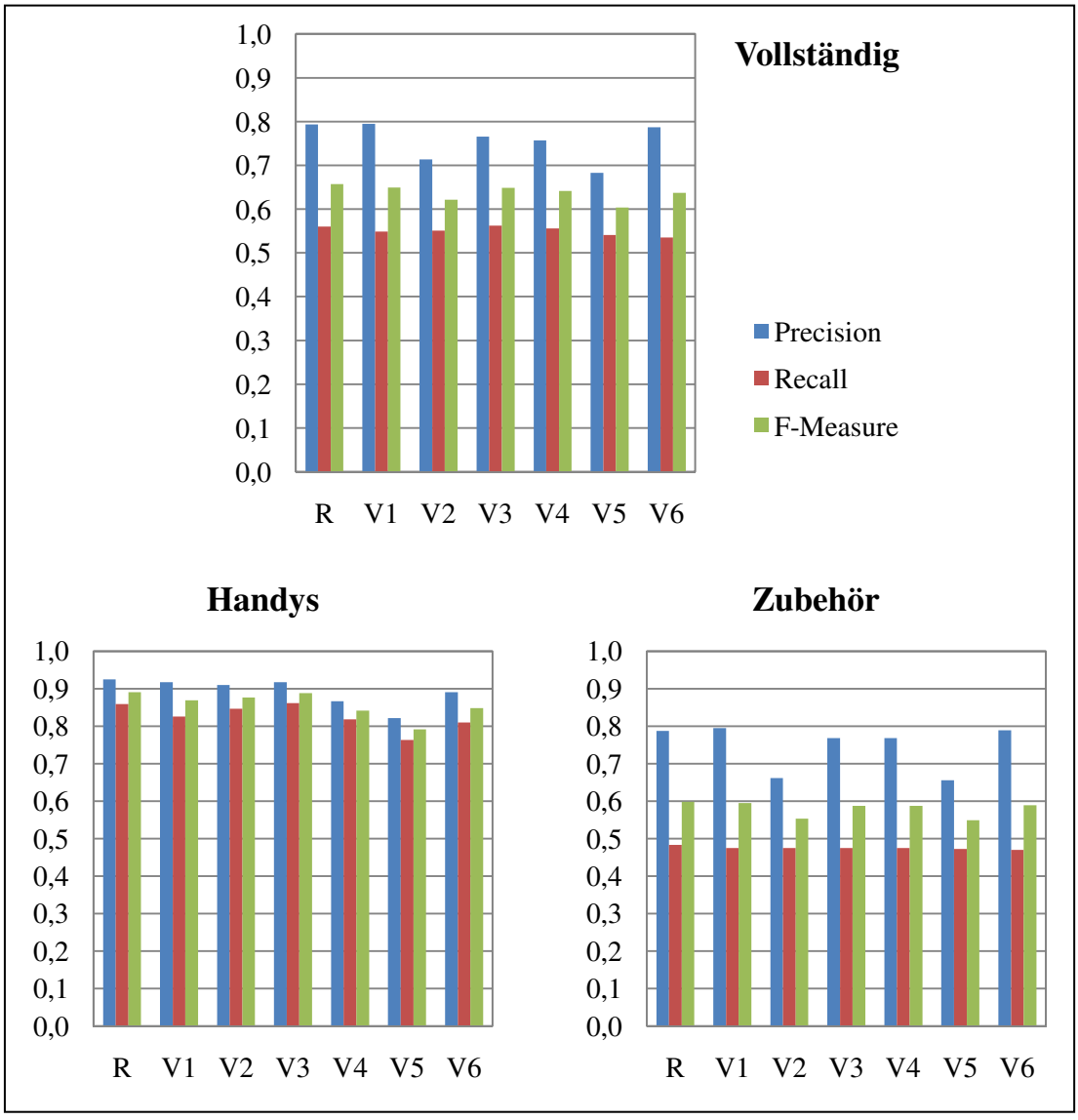

Abb. 27: Einfluss der Vorbereitung auf Produktcodeextraktion

Neben der qualitativen Verbesserung durch die Vorbereitung des Titelattributs wird auch die Effizienz signifikant erhöht, denn durch die Vorverarbeitung verringert sich die Anzahl der notwendigen Internetsuchanfragen je nach Testszenario auf 44 – 56% der ursprünglich notwendigen Anzahl.

4.5 Überblick

Die Tab. 13 gibt einen Überblick über die erreichten Resultate der in Kapitel 3 entwickelten Strategien bzgl. der drei Testszenarios. Aus dieser geht unter anderem hervor, dass die entwickelten Algorithmen ein lineares Laufzeitverhalten bzgl. der Anzahl der Angebote annehmen. Dabei bleibt die Zeit unberücksichtigt, welche vergeht, während auf die Antworten der Internetsuchanfragen gewartet wird.

Es wird für knapp ein Drittel der Angebote mit fehlendem Wert für das Herstellerattribut ein Wert ermittelt. Eine stichprobenartige Untersuchung hat ergeben, dass über 90% dieser Werte korrekt sind.

	Vollständig	Handys	Zubehör
Anzahl Angebote	102.182	5.693	52.518
Anzahl Angebote mit extrahiertem Produktcode (Precision / Recall)	53981 (0,79 / 0,56)	5.196 (0,93 / 0,86)	19127 (0,79 / 0,48)
Anzahl Angebote mit extrahierten Eigenschaften	70.373	4.400	36.248
Anzahl verschiedener Hersteller vorher → nachher	1.335 → 1157	82 → 68	860 → 700
Anzahl Angebote mit fehlendem Wert für Herstellerattribut vorher → nachher	2.355 → 1666	8 → 0	636 → 449
Verarbeitete Angebote pro Sekunde auf Testrechner[8] (exkl. Websuche)	284	247	289
Anzahl Internetsuchanfragen	100.032	2.705	32.442

Tab. 13: Überblick über Resultate der Testszenarios

[8] AMD Athlon 64 X2 3100 MHz 6000+ – 3 GB RAM – Microsoft Windows 7 Professional (32 bit).

5 Schluss

In diesem finalen Kapitel werden die Ergebnisse der Arbeit kurz zusammengefasst und Weiterentwicklungsmöglichkeiten aufgezeigt.

5.1 Zusammenfassung

Da Produkt-Matching ein komplexes Problem darstellt, wurde eine ausführliche Analyse der Ausgangslage durchgeführt, um Ursachen zu identifizieren, die das Matchen von Produktangeboten erschweren. Mit Hilfe daraus resultierender Erkenntnisse wurden zunächst Ansatzpunkte und später ausgereifte Strategien zur Vorverarbeitung entwickelt, welche das Matching unterstützen sollen. Als wichtigste Strategie hat sich hierzu die Extraktion von Produktcodes bewährt, da Produktcodes hinreichend oft (75 – 80%) in den Produktangeboten von Elektronikprodukten genannt werden und somit ein nützliches Identifizierungsmerkmal für Produkte darstellen. Das vollautomatische Erkennen und Extrahieren von Produktcodes stellt dennoch eine herausfordernde Aufgabe dar: Umfangreiche syntaktische Unterschiede, Mehrfachnennungen innerhalb eines Angebots und Abgrenzung zu anderen firmenspezifischen Bezeichnern sind Probleme, die bewältigt werden müssen.

Es konnte gezeigt werden, dass die Zuhilfenahme einer Internetsuchmaschine bei der Extraktion von Produktcodes signifikant bessere Ergebnisse hinsichtlich Precision und Recall liefert, als eine Extraktion ohne externe Wissensquelle dies ermöglicht. Unterstützt wird das Extrahieren der Produktcodes durch eine Reihe weiterer Strategien, welche im Vorfeld die Ausgangsdaten für den Extraktionsprozess vorbereiten. Im Rahmen dieser Vorbereitung werden gleichzeitig weitere für das Produkt-Matching relevante Informationen extrahiert. Das angestrebte Ziel ist es, aus Textattributen möglichst jede relevante Information in einem eigenen Attribut auszulagern, da diese Darstellung für bewährte Produkt-Matching-Verfahren am einfachsten zu handhaben ist.

Alle der vorgestellten Strategien wurden im Rahmen dieser Bachelorarbeit implementiert und in das bestehende Framework des WDI-Lab integriert. Diesbezüglich konnte auch die eingangs geforderte *vollautomatische* Vorverarbeitung – zumindest für Elektronikangebote – erreicht werden.

5.2 Weiterentwicklungsmöglichkeiten

Der Fokus bei dieser Arbeit lag auf Angeboten von Elektronikprodukten. Die Möglichkeit zur Übertragung der Erkenntnisse auf andere Produktdomänen hat dennoch von Beginn an Berücksichtigung gefunden. Zum einen äußert sich dies durch die konsequente Parametrisierung der Implementation – vor allem durch die Möglichkeit, Regeln für zu akzeptierende Typ-Kandidaten in Form von regulären Ausdrücken anzugeben. Zum anderen sollten bereits die ermittelten bzw. entwickelten Parameter in der Lage sein, ähnliche Probleme zufriedenstellend zu lösen, da die Parameter so weit wie möglich abstrahiert wurden.

Durch die sehr starke Abhängigkeit der Produktcodeextraktion von den Attributen Hersteller und Kategorie birgt ein iteratives Bereinigen dieser Attribute ein großes Verbesserungspotential für die Produktcodesuche – insbesondere für den Recall. Zum Beispiel könnte man die Ergebnisse der Websuche nach den Herstellernamen effektiver nutzen. Yahoo bietet die Möglichkeit an, die zu einer Webseite indexierten Terme zu verarbeiten. Diese Schlüsselterme könnte man nach alternativen Schreibweisen eines Herstellers durchsuchen und mit den übrigen Werten des Herstellerattributs vergleichen. Eine erste manuelle Untersuchung dieser Ergebnisse hat gezeigt, dass dies ein sehr vielversprechender Ansatz ist.

Auch der bestehende Ansatz der Websuche für das Matchen von Herstellernamen bietet Möglichkeiten zur Verbesserung. So ist ein exakter Vergleich der URLs möglicherweise zu restriktiv und das URL-Stemming[9] ein Verfahren, um eine bessere Performance zu erhalten.

[9] Als URL-Stemming bezeichnet man das Reduzieren eines URL auf den Namen des Servers.

Kurzzusammenfassung

Digital gespeicherte Daten erfreuen sich einer stetig steigenden Verwendung. Eine manuelle Konsolidierung dieser Daten ist im kommerziellen Bereich aus Kosten- und Zeitgründen praktisch nicht mehr durchführbar. Ein Verzicht auf Dublettenerkennung ist aber ebenso wenig eine Alternative. Es existieren bereits viele Ansätze um Objekt-Matching voll- bzw. zumindest semi-automatisch durchzuführen, aber insbesondere Datenbasen, welche aus Webdaten gewonnen werden, weisen eine derart hohe Heterogenität auf, dass bestehende Ansätze an ihre Grenzen stoßen. Insbesondere Produkt-Matching ist hiervon betroffen. Um Produkt-Matching-Verfahren zu unterstützen, werden hier Möglichkeiten der Vorverarbeitung vorgestellt. Es wird speziell eine Strategie entwickelt, mit der es möglich ist, gezielt Produktcodes in Textattributen zu erkennen und zu extrahieren. Diese und weitere Strategien wurden implementiert und in das bestehende Framework des WDI-Lab integriert.

Literaturverzeichnis

[B06] Bureau International des Poids et Mesures (Herausgeber): *The International System of Units*. 8. Auflage. Paris: Stedi Media 2006

[BBS05] Basu, S.; Bilenko, M.; Sahami, M.: *Adaptive Product Normalization: Using Online Learning for Record Linkage in Comparison Shopping*. Proc. of International Conference on Data Mining (ICDM) 2005

[BGG+07] Benjelloun, O.; Garcia-Molina H.; Gong H.; Kawai H.; Larson, T. E.; Menestrina D.; Thavisomboon S.: *D-Swoosh: A family of algorithms for Generic, Distributed Entity Resolution*. Proc. of International Conference on Distributed Computing Systems (ICDCS) 2007

[CFR03] Cohen, W. W.; Fienberg, S. E.; Ravikumar, P.: *A Comparison of String Distance Metrics for Name-Matching Tasks*. Proc. of Workshop on Information Integration on the Web (IIWeb) 2003

[EIV07] Elmagarmid, A. K.; Ipeirotis, P. G.; Verykios, V. S.: *Duplicate record detection: A survey*. In: IEEE Trans. Knowl. Data Eng. 19(1). 2007

[F07] Friedl, J. E. F.: *Reguläre Ausdrücke*. 3. Auflage. Köln: O'Reilly 2007

[G8] Gabler Verlag (Herausgeber): *Gabler Wirtschaftslexikon, Stichwort: E-Commerce*, online im Internet: http://wirtschaftslexikon.gabler.de/Archiv/400/e-commerce-v8.html

[KR10] Köpcke, H.; Rahm, E: *Frameworks for Entity Matching: A Comparison*. Data & Knowledge Engineering 2010

[KRT07] Kirsten, T.; Rahm, E.; Thor, A.: *Instance-based matching of hierarchical ontologies*. Proc. of 12. GI-Fachtagung für Datenbanksysteme in Business, Technologie und Web 2007

[LN07] Leser, U.; Naumann, F.: *Informationsintegration*. Heidelberg: dpunkt.verlag 2006

[T09] Thor, A. (2009) *From string-based to text-based object matching* (PDF-Datei, Stand: 16.12.2009) Internet: http://dbs.uni-leipzig.de/file/Thor_Oberseminar_WS09-10.pdf (Zugriff: 01.03.2010)

[V10] VeriSign (Herausgeber) (2010) *The Domain Name Industry Brief* (PDF-Datei, Stand: Juni 2010) Internet: http://www.verisign.com/domain-name-services/domain-information-center/domain-name-resources/domain-name-report-june10.pdf (Zugriff: 27.08.2010)

Abbildungsverzeichnis

Tabellenverzeichnis

Selbstständigkeitserklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe, insbesondere sind wörtliche oder sinngemäße Zitate als solche gekennzeichnet. Mir ist bekannt, dass Zuwiderhandlung auch nachträglich zur Aberkennung des Abschlusses führen kann.

Ort Datum Unterschrift