Erkennung bibliographischer Dubletten mittels Trigrammen : Messungen zur Performanz

Detection of Bibliographic Duplicates with Trigrams: Measuring the Performance
Reconnaissance de doublets bibliographiques à l'aide de trigrammes: Mesures de la performance

Harald Jele

Die Bildung von Trigrammen wird in der automatisierten Dublettenerkennung häufig in Situationen angewandt, in denen „sehr ähnliche“ aber nicht idente Datensätze als Duplikate identifiziert werden sollen.
In dieser Arbeit werden drei auf Trigrammen beruhende Erkennungsverfahren (das Jaccard-Maß, der euklidische Abstand sowie der Ähnlichkeitswert des KOBV) praktisch angewandt, sämtliche dabei notwendigen Schritte umgesetzt und schließlich der Verbrauch an Zeit und Ressourcen (=die „Performanz“) gemessen.
Die hier zur Anwendung gelangte Datenmenge umfasst `392.616` bibliographische Titeldatensätze, die im Österreichischen Bibliothekenverbund erbracht wurden.

Trigrams are frequently used in the automated recognition of duplicate title entries; particularly in a situation comparing „very simular“ but not equal items.
In this paper three methods of detecting duplicates on the base of their trigrams (the Jaccard similarity coefficient, the euclidean distance and the similarity coefficient from KOBV) are shown in a practical way, all necessary steps where implemented in detail and finally the amount of used time and resources (=the „performance“) are measured.
All calculations where done with `392.616` bibliographic title records from the Austrian Central Catalog.

Dans la reconnaissance automatique de doublets, on se sert fréquemment de trigrammes, tout et particulièrement dans les situations où il s'agit d'identifier des phrases de données „très ressemblantes“, mais non identiques.
Dans ce travail, trois procédés de reconnaissance reposant sur des trigrammes (la mesure Jaccard, la distance euclidienne ainsi que le coefficient de similarité de KOBV) sont utilisés de façon pratique, toutes les démarches nécessaires sont mises en œuvre et finalement les besoins en temps et ressources (=la „performance“) sont mesurés.
Tous ces calculs ont été effectués sur `392.616` phrases de données titres issues du Catalogue Central des Bibliothèques Autrichiennes.

1 Einleitung

Die Vorstellung, Ähnlichkeiten zwischen zu vergleichenden (bibliographischen) Datensätzen oder gar zwischen umfangreichen Dokumenten[1] aufgrund von

[1] Verfahren zur Bestimmung von Textähnlichkeiten bzw. -gleichheiten sind in bekannter Weise innerhalb der automatisierten Indexierung und Klassierung von Dokumenten zu finden. Daneben werden aber auch Programme zunehmend attraktiver, die aufgrund eines vorhandenen Datenpools *Text-Plagiate* erkennen sollen.
Eine Übersicht dazu findet sich z.B. in Kramer (2004).
Die automatische Erkennung und Korrektur von Tippfehlern ist eine weitere, häufig anzutreffende Anwendung solcher Verfahren (vgl. dazu auch Zamora, Pollock & Zamora (1981))

Ähnlichkeiten Ihrer „Bestandteile“ feststellen und messen zu können, ist eine in der Literatur sowie in der Realisierung praktischer Werkzeuge zur Erkennung von Titel-Dubletten weit verbreitete.
Bereits in den frühen 1970er-Jahren[2] konnte empirisch gezeigt werden, dass unter bestimmten Umständen ein günstig gewählter Ausschnitt von `55 Bytes` pro bibliographischem Datensatz ausreichen kann, um ca. `5%` eines Teilbestands von `214.000` Titelsätzen aus der damals aktuellen Datenbank des *OCLC* als Dubletten zu bestimmen.

Der Wahl des „günstigen Ausschnitts“ als die zu definierende und im Weiteren als repräsentativ angesehene Teilmenge eines Datensatzes einerseits, sowie andererseits der Wahl der anzuwendenden Verrechnungsmethoden sind dabei besondere Aufmerksamkeit zu widmen.

In der vorliegenden Arbeit wurde eine zu ladende Menge von ca. `10.000` bibliographischen Datensätzen mit einer vorhandenen Menge von ca. `437.533` und in weiterer Folge mit einer größeren von ca. `4.5 Mio.` verglichen. Das angestrebte Ziel dabei war, letztlich die zu vergleichenden Mengen dublettenfrei zusammenzuführen.
Dabei sollten vorrangig zwei Teilmengen identifiziert werden:

- die Menge der eindeutigen Dubletten (=jene Titeldatensätze, die aufgrund der gewählten Kriterien bereits vorhanden waren und zu denen ausschließlich die zu ergänzenden Bestandsangaben hinzugefügt wurden)
- jene Menge an Titeldatensätzen, die aufgrund der Verrechnungsmethode keine Ähnlichkeiten zu den bereits bestehenden zeigten. Diese Datensätze wurden als noch nicht vorhanden deklariert und zu den bestehenden Titeldatensätzen (als neue) hinzugefügt

Die entsprechend dieser Vorgehensweise entstehende dritte Menge an Titeldatensätzen, die zwar Ähnlichkeiten, jedoch keine hinreichenden Übereinstimmungen aufwiesen, wurden so gekennzeichnet, dass diese im Anschluss auf Ähnlichkeiten intellektuell geprüft werden konnten.
Bei der Mengenbildung konnten `6.100` Datensätze als Dubletten identifiziert und eine in einem nächsten Schritt noch zu bestimmende Menge als neue, hinzuzufügende Datensätze bewertet werden.

Als eine eindeutige Titel-Dublette wurde hier ein Datensatz gewertet, dessen Einträge zur `ISBN/ISSN`, zur Jahres- sowie zur Auflagenzahl (gleichzeitig) vollständig mit einem zweiten übereinstimmten. Die Möglichkeit eines dann immer noch bestehenden Unterschieds wurde vernachlässigt und im Rahmen unserer Kenntnis der Datenlage als ein kalkulierbarer Fehler angesehen.

Zur Ermittlung bzw. Messung von Ähnlichkeiten wurden drei Verfahren herangezogen, deren Ergebnisse gegenüber gestellt wurden:

- das *Jaccard-Maß* (der Jaccard-Koeffizient) wird aufgrund seiner einfachen Berechnungsmethode häufig verwendet. Sowohl im Information Retrieval einfacher (bibliographischer) Datensätze als auch zur Verarbeitung komplexer und umfangreicher Dokumentenstrukturen wird das Jaccard-Maß als ein die Ähnlichkeit bestimmender Koeffizient eingesetzt (vgl. Salton & McGill 1987, S.217)
- der *euklidische Abstand* zweier Zeichenketten-Vektoren (eigentlich: der euklidische Abstand einer geeignet gewählten Vektordarstellung der Trigrammabfolge dieser Zeichenketten) wird häufig zur Verrechnung bibliographischer Daten herangezogen. Dieser scheint ausreichend variabel interpretiertbar und auf sehr unterschiedliche, mitunter auch gewichtete Berechnungsskalen einfach anwendbar zu sein. Die Berechnung des euklidischen Abstands wird sowohl im Online- als auch im Batchverfahren eingesetzt
- der *Ähnlichkeitswert* im Modell des `KOBV` wird hier als Sonderform einer gewichteten Berechnung der Vektor-Distanz gesehen – und als solche behandelt und beschrieben: Da sich die wesentlichen Unterschiede zur Berechung gegenüber jener des euklidischen Abstands auf der Ebene von bibliographischen Datensätzen (nicht aber auf der Ebene des Vergleichs der Zeichenketten) zeigt, wird dieser Ähnlichkeitswert auch entsprechend im Zuge der Besprechung des euklidischen Abstands beschrieben.[3]

Beide Modelle sind als mathematische Modelle zur Auswertung von Vektordistanzen zwischen zu bildenen Mengen von Trigrammen definiert. Das heißt, dass beiden miteinander zu vergleichenden Datensätzen Inhalte entnommen, diese in weiterer Folge als Zeichenketten („*Strings*“) interpretiert und wie (mathematische) Vektoren behandelt werden.

Bei der Wahrnehmung deutlicher Unterschiede in den Ergebnissen der drei Modelle wurde jeweils das für die Entscheidungsfindung „schlechtere“ Ergebnis (=das pessimistischere) gewählt.

[2] vgl. dazu z.B. die in Hickey (1979) publizierten Ergebnisse, die im Wesentlichen auf frühere Ansätze zurückgehen.
Aufgegriffen und fortgesetzt wurde diese Idee bei der Bildung des `USBC` (=Universal Standard Book Code), der wesentlich später beispielgebend auch in jene Ansätze eingeflossen sind, die z.B. in Goyal (1987) genannt werden

[3] dieser Ähnlichkeitswert kann zudem als ein zeitgenössisches und sehr prominentes Beispiel gesehen werden. Das Verfahren des `KOBV` ist eines von sehr wenigen, das seit vielen Jahren zur Dublettenbehandlung beim Online-Retrieval in der `KOBV`-Suchmaschine (vgl. Kuberek 1999) eingesetzt wird.
`KOBV` = Kooperativer Bibliotheksverbund Berlin-Brandenburg.
Das dort implementierte Verfahren basiert im Wesentlichen auf dem Ansatz, wie er in Schneider (1999) dargelegt ist

Im Unterschied zu den in der Praxis gängigen Methoden der Zusammenführung bibliographischer Daten wurde hier kein Merge-Verfahren angewandt.

Bei einem solchen wird definiert, welche Kategorien aus dem dubletten bibliographischen Datensatz mit übernommen werden, bevor dieser ausgeschieden wird. Im Vorfeld war aufgrund der Datenlage zudem eindeutig geklärt, dass dublette Datensätze aus der Anfragemenge immer – und nie welche aus der Abfragemenge – ausgeschieden würden. Daher musste in weiterer Folge auch kein Gewinner/Verlierer-Konzept entwickelt werden, über das bestimmt wäre, welcher Datensatz bei Duplizität weitergeführt und welcher ausgeschieden würde.

Um gültige Aussagen auch zur Verrechnung größerer Datenmengen treffen zu können, wurde im Projektverlauf entschieden, im vorliegenden Text ausschließlich die Performanz[4] der hier referierten Modelle zu messen.
Die Gültigkeit der erzielten Ergebnisse wird in einem nachzureichenden Text aufgrund der im Verfahren entstandenen umfangreichen empirischen Datenlage eigens ausführlich geprüft.

2 Der Einsatz von Trigrammen

Bei der Bildung von Trigrammen wird die Annahme verfolgt, dass die Ähnlichkeit zwischen zwei Zeichenketten dann gegeben ist, wenn ihre jeweiligen „Bestandteile“ (=die einzelnen Zeichen in ihrer Abfolge) hinreichende Übereinstimmungen zueinander aufweisen.[5]
Zumindest für die mitteleuropäischen Sprachen scheint die Bestimmung von Ähnlichkeiten durch die Bildung von Trigrammen vorteilhafter als die Anwendung von 2- oder 4-Grammen zu sein. Dies lassen zumindest die empirischen Ergebnisse aus der gängigen Literatur vermuten.
4-Gramme „reagieren“ im Üblichen zu stark auf „Buchstabendreher“ innerhalb von Begriffen. Das heißt, dass durch die Verdrehung bloß zweier Zeichen innerhalb einer Zeichenkette das berechenbare Ähnlichkeitsmaß gegenüber der Berechnung mit Trigrammen deutlicher abnimmt als typischer Weise gewünscht. Damit tritt auch der Fall ein, dass morphologisch sehr ähnliche Wörter, die jedoch völlig verschiedenes bedeuten, mitunter ähnlicher gemessen werden als zwei idente Begriffe, die sich ausschließlich durch einen unbeabsichtigten Buchstabendreher unterscheiden.

2.1 Die Bildung von Trigrammen

Einem bibliographischen Datensatz werden in einem ersten Schritt Inhalte von bestimmten Kategorien entnommen, die im Wesentlichen Wörter bzw. Begriffe und Zahlen, sowie standardisierte Zeichenketten (wie z.B. die `ISBN` oder ein anderer, zum Vergleich geeigneter Standard-Eintrag wie evtl. eine Katalogkartennummer) darstellen. Diese werden anschließend sequentiell zu Einheiten von jeweils drei Schriftzeichen[6] zerlegt, wobei die jeweils zwei letzten Zeichen eines Trigramms am Beginn des nächsten wiederholt werden. In unserem Fall werden führende sowie den Zeichenketten nachfolgende Leerzeichen (*Blanks*) mit in die Bildung der Trigramme aufgenommen. Sollte die Ähnlichkeit zweier Ein-Wort-Begriffe[7] zueinander berechnet werden, wird darauf geachtet, diese Leerzeichen in jedem Fall den Zeichenketten am Wortanfang und -ende hinzuzufügen.

Dieses Verfahren kann an folgenden Begriffe einfach gezeigt werden. Die Wörter `Bauernmarkt` und `Marktbauern` werden in ihre Trigramme zerlegt:

- `_ba, bau, aue, uer, ern, rnm, nma, mar, ark, rkt, kt_`
- `_ma, mar, ark, rkt, ktb, tba, bau, aue, uer, ern, rn_`

Zur weiteren Berechnung werden diese als mathematische Vektoren (hier $\vec{A}$ und $\vec{B}$) verstanden:

- $\vec{A} = \{\_ba, bau, aue, uer, ern, rnm, nma, mar, ark, rkt, kt\_\}$
- $\vec{B} = \{\_ma, mar, ark, rkt, ktb, tba, bau, aue, uer, ern, rn\_\}$

Bei der anschließenden Berechnung der Ähnlichkeit dieser beiden Zeichenketten wird in den gewählten Verfah-

[4] der im Text verwendete Begriff „Performanz“ wird hier wie das englische „Performance“ als Angabe der Leistung sowie als Maß für den Verbrauch von Ressourcen in der Informatik gesehen und entsprechend verstanden

[5] Wegst bietet auf seiner Homepage eine einfache Möglichkeit, sich mit der Bildung von Trigrammen als Spezialfall von N-Grammen zur Bestimmung von Ähnlichkeiten zwischen Zeichenketten zu beschäftigen: sowohl für 2- als auch für 3-Gramme sind dort online Berechnungsverfahren frei editierbarer Zeichenketten zugänglich
vgl. `http://www.tillmann-wegst.de/`

[6] das sind Buchstaben bei Wörtern und Begriffen sowie Buchstaben und Ziffern innerhalb von freien oder standardisierten Zeichenketten. Sonderzeichen und typographische Satzzeichen werden üblicherweise ignoriert bzw. durch ein vorgeschaltetes Normierungsverfahren ersetzt (vgl. dazu *Kap.3*)

[7] die Behandlung von Ein-Wort-Begriffen gilt in Bezug auf die Behandlung der Leerzeichen wohl zu den Ausnahmen als zu den Regelfällen. Üblicherweise sind Leerzeichen zwischen den einzelnen Begriffen an den Wortgrenzen ja vorhanden und müssen in irgendeiner Form (sie ignorieren oder nicht) behandelt werden. Das heißt, dass das weitere Verfahren abhängig von ihrem (=den Ein-Wort-Begriffen) Auftreten gewählt wird. Bei Ein-Wort-Begriffen tauchen Leerzeichen in den zu bearbeitenden Zeichenketten üblicherweise nicht auf und müssen entsprechend einem konsistenten Verfahren an den beiden Wortgrenzen (Wortanfang und -ende) nachträglich hinzugefügt werden

ren von den in dieser Weise gebildeten Vektoren ausgegangen.

3 Zeichennormierung

Zwei eigentlich dublette Titel-Datensätze, die in weiterer Folge maschinell miteinander verglichen werden, können (unerwünscht) eine mathematisch geringe Ähnlichkeit zueinander dann aufweisen, wenn z.B. die in den jeweiligen Datensätzen angewandten Schreibweisen von einander abweichen. Allein leicht voneinander abweichende Varianten einer auch sehr ähnlichen Schreibweise können dazu führen, dass ein sinnvoller Schwellenwert nicht mehr ermittelbar ist, ab dem zwei miteinander zu vergleichende Datensätze als Dubletten eingeschätzt werden müssen.
Sollten den Erfassungsmethoden der zu vergleichenden Datensätze zudem verschiedene bibliothekarische Regelwerke zugrunde liegen, sind mitunter unterschiedliche Schreibweisen sogar prolongiert.

Aus diesem Grund werden in den gängigen Verfahren zur Ermittlung von bibliographischen Dubletten „Normierungen“ eingesetzt, die eine feldspezifische Behandlung von Zeichen vorsehen, bevor diese sinnvoll miteinander verglichen werden können. Die Funktion der Normierung ist dabei allein in der „Nivellierung der vorhandenen Differenzen zu sehen, möglichst ohne in die Semantik verändernd einzugreifen“ (Rusch 1999, S.3).

Aufgrund des Einsatzes unterschiedlicher Zeichensätze der hier zu vergleichenden Datenbestände (`ISO-8859-1 = Latin1` versus `UTF-8`) wurde nach umfangreichen Analysen der sich daraus ergebenden Probleme entschieden, für den Vergleich beide in einen normierten Bestand überzuführen, der ausschließlich aus Zeichen aus dem `7-bit` Zeichensatz `ASCII`, umgewandelt in Großbuchstaben, besteht.
Im Einzelnen bedeutet dies (vgl. ebda. S.5):

- Löschen von Diakritika, Akzenten, Sonderzeichen, Steuerzeichen (wie z.B. die entsprechenden Stoppwortzeichen) sowie sämtlicher Zeichensatzzeichen, die außerhalb des definierten Zeichensatzes von `ASCII-7-bit` liegen
- Umsetzen der deutschen Umlaute sowie des ß nach `AE`, `OE`, `UE`, `SS`
- Entfernen doppelter Leerzeichen (Blanks)
- Löschen der führenden sowie der abschließenden Leerzeichen
- Umsetzen der Klein- in Großbuchstaben
- Trunkierung der Feldeinträge an der feldspezifisch sinnvollen Stelle (z.B. bei Personennamen hinter dem ersten Buchstaben des zweiten Vornamens und bei bei Erscheinungsorten nach dem fünften Zeichen des ersten Wortes)
- Entfernen recherchierter Informationen, mit denen die Daten zur katalogisierenden Vorlage ergänzt und dazu entsprechend den angewandten Katalogisierungsregeln in eckigen Klammern hinzugefügt wurden

Die konkrete Anwendung der feldspezifischen Normierungsfunktionen folgte im Wesentlichen den Vorschlägen von Rusch (vgl. 1999, S.21-22). In ihrem Text wurde versucht, sämtliche, notwendige Vorgänge durch acht, sich teilweise ein wenig überschneidende Normierungsfunktionen abzubilden.
Exemplarisch können zur Veranschaulichung folgende Beispiele wiedergegeben werden:[8]

- Kategorie *Personenname*
  (entsprechend `MAB2-Kat.100`):
  `Berendt, Hans A.` → `BERENDT HANS A`
  `Huizinger, Franz Ernst ≫von≪` →
  `HUIZINGER FRANZ E`
- Kategorie *Körperschaftsname*
  (entsprechend `MAB2-Kat.200`):
  `Zentrum für Umfragen, Methoden und Analysen <Mannheim>` → `ZENTRUM FUER UMFRAGEN METHODEN UND ANALYSEN MANNHEIM`
- Kategorie *Sachtitel*
  (entsprechend `MAB2-Kat.331`):
  `≫Das≪ A -- [bis] Z der Gesundheitsvorsorge` → `DAS A Z DER GESUNDHEITSVORSORGE`
  `Über den grünen Klee [der Wiesen]` →
  `UEBER DEN GRUENEN KLEE`
- Kategorie *Ausgabebezeichnung*
  (entsprechend `MAB2-Kat.403`):
  `Ausgabe 2001` → `2001`
  `2., vollst. durchgesehende und überarb. Aufl.` → `2`
- Kategorie *Erscheinungsorte*
  (entsprechend `MAB2-Kat.410`):
  `Frankfurt am Main [u.a.]` → `FRANK`
  `Frankfurt a.M.` → `FRANK`
  `München` → `MUENC`
- Kategorie *Verleger*
  (entsprechend `MAB2-Kat.412`):
  `Gruner & Jahr` → `GRUNER JAH`
  `American Mathem. Soc.` → `AMERICAN MAT`
- Kategorie *Erscheinungsjahr*
  (entsprechend `MAB2-Kat.425`):
  `2000 [erschienen] 1999` → `2000`
  `[19]56` → `56`
  `c 1980` → `1980`

[8] die hier genannten Routinen sind vollständig und – bezogen auf die Vorgaben im Text von Rusch (1999) – kommentiert unter folgendem Link zugänglich:
`http://www.uni-klu.ac.at/~hjele/publikationen/ngramme/routinen/index.html`

- Kategorie *Umfangsangabe*
  (entsprechend `MAB2-Kat.433`):
  `211 S., [21] Bl.` → `211`
  `XVI, 123 S. : Ill., graph. Darst., Kt.` → `123`
  `Getr. Pag.` → `GETR PAG`
- Kategorie *ISBN*
  (entsprechend `MAB2-Kat.540`):
  `3-468-10120-1` → `3468101201`
  `ISBN 0-415-05603-9` → `0415056039`

4 Die gewählten Berechnungsverfahren

Vektormodelle werden in der Literatur häufig zu Ähnlichkeitsberechungen von Zeichenketten herangezogen. Das *Jaccard-Maß* bzw. der *Jaccard-Koeffizient* lassen sich als Kennziffer leicht berechnen und scheinen für das Retrieval genauso gut wie für komplexere Aufgaben geeignet zu sein (vgl. Salton & McGill 1987, S.217).

Der *euklidische Abstand* zweier Vektoren ist zur Berechnung von Ähnlichkeiten zweier oder mehrerer Zeichenketten neben dem Jaccard-Maß ein brauchbarer Ansatz. Zudem kann im deutschsprachigen Raum durchaus eine gewisse Renaissance dieser Methode zur Verrechnung bibliographischer Datensätze seit dem Erscheinen der Arbeit von Hylton (1996) wahrgenommen werden.

Wie bereits eingangs angemerkt, wird die Berechnung des *Ähnlichkeitswerts* des `KOBV` hier als Spezialfall einer Berechnung des euklidischen Abstands gesehen, dessen Eigenheiten sich „erst“ auf der Ebene des Vergleichs von Datensätzen (nicht aber auf der Ebene des Vergleichs von Zeichenketten) zeigen. Auf der Ebene der Berechnung der Vektordistanzen werden nach Lohrum, Schneider & Willenborg (1999) die euklidischen Abstände (im Gegensatz z.B. zum Ansatz von Hylton (1996)) auf einer Skala mit Werten zwischen `0` und `1` abgebildet.
Aus diesem Grund wird bei den nachfolgenden Beispielen im Wesentlichen nur die Transformation der Werte auf eine solche Skala berücksichtigt. Die Gewichtung bzw. der Einfluss einer solchen wird anschließend in *Kap.5* besprochen.

4.1 Das Jaccard-Maß

Bei der Beschreibung dieses Berechnungsverfahrens werden die Konventionen und Schreibweisen in Anlehnung an Ferber (2003, S.78-80) sowie Salton & McGill (1987, S.213-217) wiedergegeben. Jener (nämlich Ferber) weist darauf hin, dass auch einfache Anfragen des booleschen Retrievals als Spezialfälle eines Vektorraummodells verstanden werden können (vgl. S.64). Aus diesem Grund darf der Umstand weiter nicht überraschen, dass sich in der Anwendung der Berechnungsverfahren beider Modelle (des booleschen sowie jenes, das Zeichenketten als mathematische Vektoren definiert) äquivalente Verfahrensweisen feststellen lassen.
Unter einer Zeichenkette wird im Retrieval in erster Linie ein Term – bzw. aus alltagssprachlicher Sicht „ein (Such-)Begriff“ – verstanden. In weiterer Folge (abhängig vom gewählten Berechnungsverfahren) können darunter jedoch auch Kombinationen von Zeichenketten (z.B. sog. *Chunks*) verstanden werden.

Bei der Berechnung des Jaccard-Maßes werden zwei miteinander zu vergleichende Zeichenketten (Strings) – hier als w und q bezeichnet – bestimmt:

- w repräsentiert in unserem Fall jene Zeichenketten, die aus den in der vorliegenden Datenmenge (=der Datenbasis) vorhandenen bibliographischen Datensätzen gebildet und die in weiterer Folge *abgefragt* werden (=der „Abfragevektor“ oder „Dokumentvektor“)
- q steht für die Zeichenketten, die jenen bibliographischen Datensätzen entnommen werden, deren Vorkommen in der bereits bestehenden Menge (an Datensätzen) überprüft werden soll. Im Information Retrieval wird dieser String auch als der „Anfragevektor“ bezeichnet ($q \sim Query$)

Für das weitere Verfahren werden zur Verrechnung der beiden Zeichenketten (auch entsprechend der booleschen Logik) ausschließlich die Werte 0 und 1 als Gewichte zugelassen. Dieser Umstand bedeutet für die *Abfrage* bzw. den Abfragevektor formal ausgedrückt, man verwendet $w_i \in 0, 1^n$, wobei $w_{i,j} = 1$ gilt, wenn die angefragte Zeichenkette j im Datensatz i auftritt, und $w_{i,j} = 0$ bedeutet, dass die Zeichenkette j nicht im Datensatz i auftritt.

Das Jaccard-Maß wird durch das Verhältnis zwischen den Skalarprodukten des Anfragevektors und des Dokumentenvektors angegeben, wobei das Skalarprodukt durch folgenden Ausdruck definiert wird:

$$w_i \cdot q = \sum_{k=1}^{n} w_{i,k} q_k$$

Das Verhältnis der Skalarprodukte wird anschließend gebildet als:

$$s_j(w_i, q) = \frac{\sum_{k=1}^{n} w_{i,k} q_k}{\sum_{k=1}^{n} w_{i,k} + \sum_{k=1}^{n} q_k - \sum_{k=1}^{n} w_{i,k} q_k}$$

Betrachtet man dieses Maß – wie oben angeführt – für mit den Werten 0 und 1 gewichtete Vektoren, dann steht im Zähler immer genau die Anzahl der Begriffe/Zeichenketten aus der Anfrage, die sowohl im Anfragestring als auch (=logisch *UND*) im abgefragten Datensatz vorkommen. Diese Ergebnismenge entspricht der

Größe der Schnittmenge aus der Anfrage und dem Datensatz.
Im Nenner hingegen steht die Anzahl der Begriffe/Zeichenketten, die in der Anfrage *oder* im geprüften Datensatz vorkommen, also die Größe der Vereinigungsmenge von Anfrage und Datensatz.

Bei der Berechnung der Ähnlichkeit zwischen den beiden zum Vergleich herangezogenen Vektoren ist in diesem Fall der Nenner immer größer oder gleich 1, wenn einer der beiden Vektoren einen Eintrag mit dem Wert 1 enthält. Der Ähnlichkeitswert liegt beim Jaccard-Maß demnach immer zwischen 0 (=gänzlich unähnlich) und 1 (=übereinstimmend) (vgl. Ferber 2003, S.78).

Im hier vorliegenden Ansatz werden Zeichenketten (also im Wesentlichen Begriffe sowie standardisierte Einträge der zu vergleichenden Kategorien) den entsprechenden Feldeinträgen in normierter Form entnommen und in Folgen von Trigrammen zerlegt.
Anschließend werden diese Abfolgen von Trigrammen miteinander entsprechend der booleschen Logik verglichen und daraus die sich ergebende Schnittmenge sowie die Vereinigungsmenge gebildet, die für die weitere Berechnung des Jaccard-Maßes notwendig sind.

Das für die Anwendung des Jaccard-Maßes zu berücksichtigende Problem, dass seine geometrischen Funktionskurven Unstetigkeiten aufweisen (vgl. Jones & Furnas 1987), kann hier vernachlässigt werden.
Dieses tritt nur dann auf, wenn die einzelnen, zur Berechnung herangezogenen Vektoren andere Werte außer 0 und 1 annehmen können. Dann allerdings ist es möglich, dass selbst sehr voneinander verschiedene (eigentlich völlig „beliebige“) Vektoren auf einer Linie gleicher Ähnlichkeitswerte zu finden sind.[9]

4.2 Der euklidische Abstand

Zur Berechnung der Ähnlichkeit zweier Zeichenketten kann der euklidische Abstand einer geeignet gewählten Vektordarstellung der Trigrammabfolge dieser Zeichenketten verwendet werden.
Die beiden zu vergleichenden Zeichenketten werden daher als Vektoren ihrer Trigrammabfolge aufgefasst.

Der Vektor $\vec{A}$ einer Zeichenkette wird dazu in eine Anzahl m_A Teile (Trigramme) zerlegt. Daneben existiert eine zweite Zeichenkette, dessen Vektor $\vec{B}$ mit $\vec{A}$ auf der Basis seiner Trigramme verglichen werden soll. Die Definitionen gelingen formal durch folgenden Ausdruck:

$$\vec{A} = (A_1, A_2, \ \dots \ , A_{m_A})$$
$$\vec{B} = (B_1, B_2, \ \dots \ , B_{m_B})$$

Gebildet wird nun die Menge X, in der alle in $\vec{A}$ oder $\vec{B}$ vorkommenden Trigramme enthalten sind:

$$X = \{x_1, x_2, \ \dots \ , x_{m_X}\}$$

Für alle in X enthaltenen Trigramme wird anschließend überprüft, an welcher Stelle (Position) das jeweilige Trigramm im Vektor $\vec{A}$ vorkommt. Wobei s die Position eines Trigramms im Vektor $\vec{A}$ bezeichnet.

Wenn ein Trigramm an der Position s in $\vec{A}$ vorkommt, so wird dieses Vorkommen (in unserem Fall) binär in einer Matrix vermerkt (kodiert).
Für die Elemente der Matrix gilt

$$A_{si} = \begin{cases} 1 & \text{für } x_i = A_s \\ 0 & \text{für } x_i \neq A_s \end{cases}$$

$s = 1, 2, \dots, m_A$, $i = 1, 2, \dots, m_X$.

Die Häufigkeit, mit der ein Trigramm in $\vec{A}$ vorkommt, wird als Zeilensumme der Matrix durch

$$a_i = \sum_{s=1}^{m_A} A_{si} \qquad i = 1, 2, \dots, m_X$$

bestimmt.

Zur Veranschaulichung dieses Vorgangs ist folgende Übersicht nützlich:

	A_1	A_2	$\cdots$	A_{m_A}	
x_1	A_{11}	A_{12}	$\cdots$	A_{1m_A}	a_1
x_2	A_{21}	A_{22}	$\cdots$	A_{2m_A}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
x_{m_X}	A_{m_X1}	A_{m_X2}	$\cdots$	$A_{m_Xm_A}$	a_{m_X}

Nach der analogen Behandlung des Vektors $\vec{B}$ lässt sich der euklidische Abstand durch folgenden Ausdruck bilden:

$$\|\vec{D}\| = \sqrt{\sum_{i=1}^{m_X} (a_i - b_i)^2}$$

Die Ähnlichkeit zweier, miteinander zu vergleichender Zeichenketten wird angenommen, wenn der Abstand der beiden Vektoren (eigentlich: die sich ergebende Differenz aus den vorhandenen Trigrammabfolgen) einen zu definierenden Schwellenwert (T, Threshold) nicht übersteigt. Dieser Schwellenwert wird üblicherweise aufgrund empirischer Messungen bzw. durch Schätzungen bestimmt.
Hylton (1996, S.47) hat für die Anwendung seines Algorithmus den Schwellenwert aufgrund einer Stichprobe

[9] siehe dazu auch die entsprechende *Abb.3.26* in Ferber (2003, S.80)

mit (nur) 50 Paaren an zu vergleichenden Zeichenketten festgesetzt als:

$$T = 2.486 + 0.025 * n$$

wobei n die Anzahl der zu vergleichenden Teilstrings in der Vereinigung beider Vektormengen darstellt.[10]

Zur Verdeutlichung folgt an dieser Stelle ein Beispiel, an dem gezeigt wird, wie mehrfach vorhandene Trigramme verarbeitet werden.
Zu vergleichen sind die beiden Wörter „`Flachdach`“ und „`Kuhfladen`“.

$$\begin{aligned} \vec{A} &= (\texttt{fla, lac, ach, chd, hda, dac, ach}) \\ \vec{B} &= (\texttt{kuh, uhf, hfl, fla, lad, ade, den}) \end{aligned}$$

Die Menge X, die alle in Vektor $\vec{A}$ oder $\vec{B}$ vorkommenden Trigramme enthält ist

$$X = \{\texttt{fla, lac, ach, chd, hda, dac, kuh, uhf, hfl, lad, ade, den}\}$$

Für den Vektor $\vec{A}$ ergibt sich auf die Elemente der Menge X folgende Matrix:

$X \vert \vec{A}$	fla	lac	ach	chd	hda	dac	ach	a_i
fla	1	0	0	0	0	0	0	1
lac	0	1	0	0	0	0	0	1
ach	0	0	1	0	0	0	1	2
chd	0	0	0	1	0	0	0	1
hda	0	0	0	0	1	0	0	1
dac	0	0	0	0	0	1	0	1
kuh	0	0	0	0	0	0	0	0
uhf	0	0	0	0	0	0	0	0
hfl	0	0	0	0	0	0	0	0
lad	0	0	0	0	0	0	0	0
ade	0	0	0	0	0	0	0	0
den	0	0	0	0	0	0	0	0

Für den Vektor $\vec{B}$ ergibt sich auf die Elemente der Menge X folgende Matrix:

$X \vert \vec{B}$	kuh	uhf	hfl	fla	lad	ade	den	b_i
fla	0	0	0	1	0	0	0	1
lac	0	0	0	0	0	0	0	0
ach	0	0	0	0	0	0	0	0
chd	0	0	0	0	0	0	0	0
hda	0	0	0	0	0	0	0	0
dac	0	0	0	0	0	0	0	0
kuh	1	0	0	0	0	0	0	1
uhf	0	1	0	0	0	0	0	1
hfl	0	0	1	0	0	0	0	1
lad	0	0	0	0	1	0	0	1
ade	0	0	0	0	0	1	0	1
den	0	0	0	0	0	0	1	1

Daraus errechnet sich der euklidische Abstand mit

$$\begin{aligned} \|\vec{D}\| &= \sqrt{(1-1)^2 + (1-0)^2 + \ldots + (0-1)^2} \\ &= \sqrt{0^2 + 1^2 + 2^2 + 3 \cdot 1^2 + 6 \cdot (-1)^2} \\ &= \sqrt{14} \approx 3.74 \end{aligned}$$

4.3 Beispiele

Anhand einiger Zeichenketten werden die eben beschriebenen Berechnungsmethoden konkret gezeigt, sowie deren Ergebnisse kommentiert gegenüber gestellt.
Nach der Zerlegung in Trigramme erfolgt zuerst die Ermittlung des Jaccard-Maßes, dann die Berechnung des euklidischen Abstands der entsprechenden Zeichenketten-Vektoren in vereinfachter Form.[11]
Um die Werte besser einschätzen zu können, wird abschließend zu jedem Beispiel das in *Fußnote 10* genannte Verfahren zum Normalisieren der euklidischen Abstands auf eine Skala von 0 bis 1 angewandt.

4.3.1 Beispiel 1: Vergleich der beiden Begriffe „Bauernmarkt“ und „Marktbauern“

Die Trigramm-Bildung ergibt – wie in *Kap.2.1* bereits gezeigt – die Zeichenketten

- `bau, aue, uer, ern, rnm, nma, mar, ark, rkt`
- `mar, ark, rkt, ktb, tba, bau, aue, uer, ern`

Entsprechend der hier gewählten Vorgaben für die Berechnung des Jaccard-Maßes wird dieses aus dem Mengenvergleich zwischen der booleschen Schnittmenge und der Vereinigungsmenge der beiden in Trigramme zerlegten Zeichenketten gebildet:

[10] ein ähnliches Verfahren zur Festsetzung eines Schwellenwerts wird im Beitrag von Lohrum, Schneider & Willenborg (1999, S.5-7) vorgestellt, wobei dort der Kehrwert des euklidischen Abstands auf eine Werteskala zwischen 0 und 1 normalisiert und der Schwellenwert $T = 0.8$ als definierte Größe für die Bestimmung ähnlicher Zeichenketten, angenommen wird

[11] vereinfacht heißt hier: nachdem in *Kap.4.2* anhand von Matrizen genau gezeigt wurde, wie einzelne Trigramme behandelt werden, auch wenn diese in den Zeichenketten mehrfach vorkommen, werden an dieser Stelle nur noch die nach der Bildung der Vektordifferenz verbleibenden Trigramme zur weiteren Berechnung angegeben

- beide Zeichenketten beinhalten je `9 Trigramme`
- es existieren `7 Übereinstimmungen` = `Schnittmenge` $\{bau, aue, uer, ern, mar, ark, rkt\}$
- in der `Vereinigungsmenge` sind entsprechend der Booleschen Logik die `7 Elemente` der Schnittmenge `plus` die je zwei Elemente der einzelnen Teilmengen ($\{rnm, nma\}$ sowie $\{tba, aue\}$) enthalten = `11 Elemente`

Daraus ergibt sich als `Jaccard-Maß` =
`Schnittmenge` ÷ `Vereinigungsmenge` =
`7` ÷ `11` = `0.636`

Der Wert `1` bedeutet auf der Skala zur Bewertung des Koeffizienten Gleichheit beider Zeichenketten, `0` Unähnlichkeit. Für den Vergleich von ausgewählten Einwort-Titeln im zu vergleichenden Bestand wurde ein empirisch ermittelter Schwellenwert von `0.7` festgelegt, über dem Zeichenketten als hinreichend ähnliche bestimmt wurden.
Der ermittelte Wert von `0.636` kann dem entsprechend als zu geringe Ähnlichkeit interpretiert werden.

Die Darstellung der beiden Zeichenketten als Vektoren der zu vergleichenden Trigramme zur Berechnung des euklidischen Abstands erfolgt durch folgenden Ausdruck

$\vec{A}$ = `(bau,aue,uer,ern,rnm,nma,mar,ark,rkt)`
$\vec{B}$ = `(mar,ark,rkt,ktb,tba,bau,aue,uer,ern)`

Nach Auswertung der beiden Matrizen für die Vektoren $\vec{A}$ und $\vec{B}$ zeigt sich, dass vier Elemente der Menge X zur Bildung der Differenz herangezogen werden müssen: `(rnm, nma, ktb, tba)`.
Alle Elemente kommen jeweils nur einmal vor.

$$\|\vec{D}\| = \sqrt{1^2 + 1^2 + 1^2 + 1^2} = 2$$

Eine hinreichend große Ähnlichkeit zwischen den beiden wird angenommen, wenn der euklidische Abstand den Schwellenwert T nicht übersteigt. Dieser errechnet sich im hier (in Anlehnung an Hylton 1996) gewählten Verfahren durch (siehe *Kap.4.2*)

$$T = 2.486 + 0,025 * 11 = 2.761$$

Da $2 < 2.761$ wird in diesem Verfahren eine im Weiteren wahrzunehmende Ähnlichkeit zwischen den beiden Begriffen angenommen.

Mit der Normalisierungsfunktion, die in Lohrum, Schneider & Willenborg (1999, S.6) beschrieben ist, werden die Werte des euklidischen Abstands auf eine Skala zwischen 0 und 1 projiziert, wobei die Werte 0 geringe Ähnlichkeit, 1 eine Übereinstimmung der zu vergleichenden Zeichenketten repräsentieren.
Der Wert 0.8 wird als Schwellenwert angenommen, über dem sämtliche Werte hinreichende Ähnlichkeit abbilden.

Für den Umstand, dass $\|\vec{D}\| < T$ wird zur Normalisierung des errechneten Ähnlichkeitswertes S folgende Funktion eingesetzt

$$S = \frac{4}{5} + \frac{T - D}{5 * T}$$

Für den Fall, dass $\|\vec{D}\| > T$ wird zur Normalisierung des errechneten Ähnlichkeitswertes S hingegen diese Funktion eingesetzt

$$S = \frac{4}{5} - \frac{D - T * 4}{(1 + D - T) * 5}$$

Im Fall, dass $\|\vec{D}\| = T$ wird als Ähnlichkeitswert $S = 0.8$ angenommen.

Unter Anwendung dieser Bedingungen kann für diese Beispiel ein Ähnlichkeitswert von `0.855` errechnet werden.
Da dieser über dem dort definierten Schwellenwert von `0.8` liegt, würde dieses Zeichenkettenpaar wie im Verfahren nach Hylton (1996) – aber im Gegensatz zur Anwendung des Jaccard-Koeffizienten `0.636 < 0.7` – als ähnlich gewertet werden.

4.3.2 Beispiel 2: Vergleich der beiden Begriffe „_Bauernmarkt_“ und „_Marktbauern_“

An diesem Vergleich kann gezeigt werden, dass die Berücksichtigung der Leerzeichen („Blanks“) an den Wortgrenzen zu einer deutlicheren Unterscheidung morphologisch ähnlicher Zeichenketten führt.

Die Trigramm-Bildung ergibt die Zeichenketten

- `_ba, bau, aue, uer, ern, rnm, nma, mar, ark, rkt, kt_`
- `_ma, mar, ark, rkt, ktb, tba, bau, aue, uer, ern, rn_`

Für die Berechnung des Jaccard-Maßes gilt:

- beide Zeichenketten beinhalten je `11 Trigramme`
- es existieren `7 Übereinstimmungen` = `Schnittmenge` $\{bau, aue, uer, ern, mar, ark, rkt\}$
- die `Vereinigungsmenge` beinhaltet `7 Elemente` der Schnittmenge `plus` die je vier Elemente der einzelnen Teilmengen ($\{\_ba, rnm, nma, kt\_\}$ sowie $\{\_ma, tba, aue, rn\_\}$) enthalten = `15 Elemente`

Daraus ergibt sich als `Jaccard-Maß` =
`Schnittmenge` ÷ `Vereinigungsmenge` =
`7` ÷ `15` = `0.467`

Gegenüber dem Wert von `0.636`, der bei Vernachlässigung der Leerzeichenstellen an den Wortgrenzen ermittelt wird, ist der Wert des Koeffizienten – und

die damit ausgedrückte Ähnlichkeit der Zeichenketten – bei Berücksichtigung dieser mit `0.467` deutlich geringer (und entsprechend weiter vom Schwellenwert `0.7` entfernt).

Die beiden Vektoren zur Berechnung des euklidischen Abstands sind

- $\vec{A}_{S1} = \{(\_ba, 1), (bau, 1), (aue, 1), (uer, 1), (ern, 1), (rnm, 1), (nma, 1), (mar, 1), (ark, 1), (rkt, 1), (kt\_, 1)\}$
- $\vec{A}_{S2} = \{(\_ma, 1), (mar, 1), (ark, 1), (rkt, 1), (ktb, 1), (tba, 1), (bau, 1), (aue, 1), (uer, 1), (ern, 1) (rn\_, 1)\}$

Die Differenz der beiden Vektoren ist

$$\begin{aligned}\|\vec{D}\| &= \vec{A}_{S1} - \vec{A}_{S2} \\ &= \{(\_ba, 1), (rnm, 1), (nma, 1), (kt\_, 1), (\_ma, 1), \\ &\quad (ktb, 1), (tba, 1), (rn\_, 1)\} \\ &= \sqrt{8} = 2.828\end{aligned}$$

Für den Schwellenwert T ergibt sich

$$T = 2.486 + 0,025 * 15 = 2.861$$

Da $2.828 < 2.861$ wird laut diesem Verfahren wiederum eine im Weiteren zu berücksichtigende Ähnlichkeit zwischen den beiden Begriffen angenommen.

Für das in Beispiel 1 vorgestellte Normalisierungsverfahren von Lohrum, Schneider & Willenborg (1999) ergibt sich für den errechneten euklidischen Abstand ein Ähnlichkeitswert von `0.827`.
Dieser Wert liegt über dem dort definierten Schwellenwert von `0.8`. Aus diesem Grund würden diese beiden Zeichenketten – ebenso wie im Verfahren von Hylton (1996), jedoch im Gegensatz zum errechneten Jaccard-Maß – als hinreichend ähnlich gewertet werden.

4.3.3 Beispiel 3: Vergleich der beiden Begriffe „_Bauernmarkt_“ und „_Tauernmarkt_“

Hier zeigt sich – in Ergänzung zum Beispiel 2 – deutlich, dass die Berücksichtigung der Leerzeichen („Blanks“) an den Wortgrenzen auch bei Verschiedenheit von nur einem Buchstaben am Wortanfang oder -ende zu einer deutlichen Unterscheidung morphologisch ähnlicher Zeichenketten führt.
Diese Eigenheit ist eines der wesentlichen Charakteristika in der Verrechnung von Trigrammen: kleine Differenzen am Wortanfang und -ende führen üblicherweise zu deutlichen Differenzen in der Berechnung von Ähnlichkeitskoeffizienten (vgl. dazu auch *Kap.2*).

Die Trigramm-Bildung ergibt die Zeichenketten

- `_ba, bau, aue, uer, ern, rnm, nma, mar, ark, rkt, kt_`
- `_ta, tau, aue, uer, ern, rnm, nma, mar, ark, rkt, kt_`

Für die Berechnung des Jaccard-Maßes gilt:

- beide Zeichenketten beinhalten je `11 Trigramme`
- es existieren `9 Übereinstimmungen` = `Schnittmenge` $\{aue, uer, ern, rnm, rma, mar, ark, rkt, kt\_\}$
- die `Vereinigungsmenge` beinhaltet `9 Elemente` der Schnittmenge `plus` die je zwei Elemente der einzelnen Teilmengen ($\{\_ba, bau\}$ sowie $\{\_ta, tau\}$) enthalten = `13 Elemente`

Daraus ergibt sich als `Jaccard-Maß` = `Schnittmenge` ÷ `Vereinigungsmenge` = `9` ÷ `13` = `0.692`

Die Differenz von nur einem Buchstaben am Wortanfang zwischen den Zeichenketten „Bauernmarkt“ und „Tauernmarkt“ wirkt sich unter Berücksichtigung der Leerzeichenstellen an den Wortgrenzen im sich ergebenden Koeffizienten von `0.692` markant messbar – und knapp am Schwellenwert von `0.7` – aus.
Bei Vernachlässigung der Leerzeichenstellen ergibt sich für dieses Beispiel ein Wert von `0.8`. Dieser liegt deutlich oberhalb des Schwellenwerts von `0.7`.

Die beiden Vektoren zur Berechnung des euklidischen Abstands sind

- $\vec{A}_{S1} = \{(\_ba, 1), (bau, 1), (aue, 1), (uer, 1), (ern, 1), (rnm, 1), (nma, 1), (mar, 1), (ark, 1), (rkt, 1), (kt\_, 1)\}$
- $\vec{A}_{S2} = \{(\_ta, 1), (tau, 1), (aue, 1), (uer, 1), (ern, 1), (rnm, 1), (nma, 1), (mar, 1), (ark, 1), (rkt, 1), (kt\_, 1)\}$

Die Differenz der beiden Vektoren ist

$$\begin{aligned}\|\vec{D}\| &= \vec{A}_{S1} - \vec{A}_{S2} \\ &= \{(\_ba, 1), (bau, 1), (\_ta, 1), (tau, 1)\} \\ &= \sqrt{4} = 2\end{aligned}$$

Für den Schwellenwert T ergibt sich

$$T = 2.486 + 0,025 * 13 = 2.811$$

Da der errechnete Abstand $2 < 2.811$ ist, wird laut diesem Verfahren eine zu berücksichtigende Ähnlichkeit zwischen den beiden Begriffen angenommen.
Bei Vernachlässigung der Leerzeichenstellen an den Wortgrenzen ergibt sich ein deutlich geringerer Abstand $\|\vec{D}\| = \sqrt{2} = 1.414 < T = 2.736$.

Entsprechend dem im Beispiel 1 vorgestellten Verfahren von Lohrum, Schneider & Willenborg (1999, S.6) wird mit diesem ein Wert von `0.855` für das Maß an Ähnlichkeit ermittelt. Dieser Ähnlichkeitswert liegt über

dem dort definierten Schwellenwert von `0.8`. Daher wird auch nach dieser Berechnung von einer Ähnlichkeit ausgegangen.

4.3.4 Übersicht vergleichbarer Werte

Im hier mehrfach zitierten Aufsatz von Lohrum, Schneider & Willenborg (1999) wird eine Tabelle angeführt, mit der die Ähnlichkeitskoeffizienten für den euklidischen Abstand sowie für den daraus abgeleiteten Ähnlichkeitswert (auf einer Skala von 0 bis 1) an einigen, ausgewählten Beispielen dargestellt werden.

Diese Tabelle wurde hier übernommen und um die Spaltenwerte für den Jaccard-Koeffizienten erweitert. Um den aus den Berechnungsvorgaben entwickelten Programm-Algorithmus zu testen, wurden die in *Abb.1* wiedergegebenen Werte zudem „händisch“ überprüft.[12]

Bei einem ersten Vergleich der Ergebnisse aus *Abb.1* zeigt sich, dass bei Berücksichtigung eines Schwellenwerts von `0.8` der berechnete Ähnlichkeitswert fünf mal den Schwellenwert übersteigt. Diese fünf Paare von Zeichenketten wären demnach für eine weitere Beachtung als mögliche Dubletten aufgrund Ihrer Zeichenähnlichkeiten von Relevanz.
Für vier der angeführten Paare übersteigt der berechnete Jaccard-Koeffizient den hier verwendeten Schwellenwert von `0.7`.
Somit tragen alle vier in der Tabelle durch das Jaccard-Maß angezeigten Dubletten auch einen Ähnlichkeitswert über der angegebenen Schwelle.

Aus der Gegenüberstellung dieser (sehr vereinfachten) Wertangaben erkennt man zudem, dass ein Zeichenketten-Paar zwar im Wesentlichen gleichzeitig einen hohen Ähnlichkeitswert und zumeist einen eher hohen Jaccard-Koeffizienten aufweist. Man erkennt aber auch deutlich, dass diese beiden Maßangaben kein lineares Verhältnis zueinander zeigen.

In diesem Umstand liegt z.B. begründet, dass einerseits das Zeichenketten-Paar
`blue velvet -- green velour`
das nach „menschlicher“ Interpretation wenig (morphologische) Ähnlichkeiten zueinander zeigt, einen relativ hohen Ähnlichkeitswert von `0.408` aber einen sehr geringen Jaccard-Koeffizienten von `0.118` aufweist,
andererseits das Zeichenketten-Paar
`springer verlag -- vrlg. springer`
einen eher weit vom Schwellenwert entfernten Ähnlichkeitswert von `0.486`, jedoch einen knapp unter dem Schwellenwert liegenden Jaccard-Koeffizienten von `0.667` aufweist.

Eine rein intellektuelle Interpretation dieser einfachen Gegenüberstellung würde wohl nahelegen, dass der Jaccard-Koeffizient eher Dubletten anzeigt, die auch in der menschlichen und nicht nur durch die maschinelle Wahrnehmung als solche eingeschätzt würden.

Die Interpretation dieser wenigen Beispiele ist jedoch für eine Gesamtdarstellung nicht weiterführend. Sie soll an dieser Stelle ausschließlich auf einige Eigenheiten hinweisen – und die aus *Abb.1* bekannten Werte jenen zur Ermittlung des Jaccard-Koeffizienten gegenüberstellen.

Zeichenkette 1	Zeichenkette 2	Eukl. Abstand	Ähnlich-keitswert	Jaccard-Koeff.
blue velvet	green water	4,243	0,347	0,000
blue velvet for all his clothes	a big shark swimming in green water	7,874	0,165	0,000
blue velvet	green velour	3,873	0,408	0,118
1997	1998	1,414	0,890	0,333
springer verlag	springer assoc.	3,464	0,532	0,368
springer verlag	vrlg. springer	3,606	0,486	0,667
springer verlag	springland verbund	4,123	0,388	0,261
springer verlag	verl. springer	3,000	0,735	0,471
springer verlag	springer verl.	1,732	0,878	0,786
introduction to modern information retrieval	introduction to text search algorithms in information retrieval	5,745	0,313	0,515
introduction to modern information retrieval	introduction to modern information retrieval	0,000	1,000	1,000
introduction to modern information retrieval	modern introduction to information retrieval	2,000	0,888	0,909
introduction to modern information retrieval	information retrieval	4,796	0,354	0,452
klinische psychologie teil I	klinische psychologie teil II	1,000	0,937	0,963

Abbildung 1: Erweiterte Wertetabelle in Anlehnung an Lohrum, Schneider & Willenborg (1999, S.6)

5 Ähnlichkeit auf der Ebene von Datensätzen

Bisher wurden die Berechnung und Interpretation der Zeichenketten-Ähnlichkeit auf der Ebene einzelner Felder bzw. ihrer bibliographischen Kategorien betrachtet. Um zwei oder mehrere Datensätze als ähnlich oder gleich (im Sinne einer möglichen Titel-Dublette) zu qualifizieren bedarf es jedoch einer umfangreicheren Interpretation.

Der einfachste dabei zu verfolgende Ansatz ist – entsprechend den verwendeten Vektormodellen – einen Gesamtvektor über die zum Vergleich herangezogenen und bereits normalisierten Kategorieninhalte zu berechnen.
In diesem Fall wäre der Gesamtvektor die Summe seiner Teilvektoren.
Dieser Ansatz vernachlässigt jedoch, dass bei der Interpretation der errechneten Ergebnisse eine Wertigkeit festgestellt werden kann, mit der zum Ausdruck gebracht wird, dass eine errechnete Ähnlichkeit innerhalb

[12] die daraus entstandenen Unterlagen finden sich unter folgendem Link und können als Veranschaulichung weiterer Berechnungsbeispiele verstanden werden:
`http://www.uni-klu.ac.at/~hjele/publikationen/ngramme/haendisches/index.html`

einer bestimmten Kategorie „mehr Wert“ sein kann, als die gleiche innerhalb einer anderen Kategorie.

Aus diesem Grund wird bei der Aufsummierung der Teilvektoren zumeist eine Gewichtung vorgenommen, mit der diese Wertigkeit formuliert wird. Der Gewichtungsfaktor wird mit dem entsprechenden Ähnlichkeitswert eines bestimmten Kategorieninhalts multipliziert. Die Summe aller so berechneten Werte ergibt das Maß für die Bestimmung der Ähnlichkeit der zum Vergleich herangezogenen Datensätze.

Wenn also die Kategorien *Personenname* (`P`), *Sachtitel* (`T`), *Ausgabebezeichnung* (`B`), *Erscheinungsorte* (`O`), *Verleger/in* (`V`), *Erscheinungsjahr* (`J`), *Umfangsangabe* (`U`) und *ISBN* (`I`) zur Dublettenkontrolle zweier monographischer Werke herangezogen werden und diese in gewichteter Form[13] aufsummiert werden, ergibt sich folgende Berechnung für die Gesamt-Ähnlichkeit (`G`):[14]

$$G = 0.3 * P + 0.7 * T + 0.3 * B + 0.2 * O + 0.2 * V + 0.3 * J + 0.2 * U + 0.7 * I$$

Entsprechend der bisherigen Beschreibung wird dabei angenommen, dass der errechnete Wert für die Gesamt-Ähnlichkeit (eines gewählten Datensatzes zu einem zweiten) einen Schwellenwert (T, Threshold) übersteigen muss, um als Dublette gewertet zu werden.

Reichart & Mönnich (vgl. 1994, S.204) weisen darauf hin, dass diese Annahme (nämlich das alleinige Aufaddieren positiv gewichteter Werte) unter bestimmten Umständen zu verfälschten Ergebnissen führen kann.
Als Beispiele dafür führen sie errechnete Titeldubletten an, die in den Kategorien *Personenname*, *Sachtitel*, *Verleger/in* und *Erscheinungsorte* keine, dafür aber in den Kategorien *Erscheinungsjahr*, *Umfangsangabe* und *Ausgabebezeichnung* leichte Abweichungen aufweisen (also z.B. ein Werk, das ein oder zwei Jahre später in einer höheren Auflage erschienen ist).

Nach der Analyse und dem Referieren verschiedener Verfahren gelangen Reichart & Mönnich (vgl. ebda., S.205) letztlich zum Ergebnis, dass die Zuverlässigkeit der Ergebnisse gesteigert werden kann, wenn das Berechnungsverfahren einer „Kombination von Bedingungen“ gehorcht.[15]

Dieser Empfehlung folgend wird auch in der Modellberechnung des `KOBV` eine Titeldublette erst dann als eine solche gewertet, wenn die positiv gewichtete Summe der Gesamt-Ähnlichkeit über dem zu berechnenden Gesamt-Schwellenwert und zugleich die negativ gewichtete Summe unter dem entsprechenden Schwellenwert liegt.[16]
Eine positive Gewichtung erfährt eine Kategorie, welche im Paar-Vergleich zweier Datensätze den bereits beschriebenen Schwellenwert von `0.8` übersteigt. Anderenfalls wird deren Ähnlichkeitswert negativ gewichtet.[17]

Kategorie	pos. Gewichtung	neg. Gewichtung
Personenname	30	50
Körperschaftsname	30	50
Sachtitel	70	70
Standardnummer (z.B. ISBN)	70	60
Erscheinungsjahr	30	60
Erscheinungsorte	20	30
Verleger/in	20	20
Ausgabe-bezeichnung	30	60
Umfangsangabe	20	40

Abbildung 2: Wertetabelle der positiven und negativen Gewichtungsfaktoren im Modell des `KOBV`
Quelle: Lohrum, Schneider & Willenborg (1999, S.17)

6 Vorgehensweise

Die hier beschriebenen Verfahren – nämlich die Berechnung des Jaccard-Maßes, des euklidischen Abstands sowie des Ähnlichkeitswertes des `KOBV` – wurden vergleichend eingesetzt und anschließend die Ergebnisse einander gegenübergestellt. Dabei wurden Messungen zur Performanz der Verfahren durchgeführt.
Das heißt, dass sequentiell alle Datensätze der zu ladenden Datenmenge nach beiden Verfahren auf Duplizität geprüft wurden, sowie dass die dafür benötigte Zeitdauer gemessen wurde.

---

[13] die hier herangezogenen Gewichte entsprechen jenen des `KOBV`, wie sie in Lohrum, Schneider & Willenborg (1999, S.17) genannt sind. Siehe dazu auch *Abb.2*

[14] eine Übersicht zu den (bei einer Dublettenbestimmung üblicherweise herangezogenen) bibliographischen Kategorien findet sich in Kuberek (1999, S.25 (=Anlage 1)).
Für den angelsächsischen Raum bzw. für typische bibliographische Datenbanken, deren Datenformat auf `MARC` basiert kann der Beitrag von O'Neill, Rogers & Oskins (1993) von Interesse sein, wenn im Datenbestand zugleich eine Analyse der zur Dublettenerkennung heranzuziehenden Kategorien durchgeführt werden muss

[15] ob diese Annahme auch in den hier vorgestellten Verfahren ihre Gültigkeit bzw. Tauglichkeit auch für ein Offline-Verfahren behält, muss die weitere Auswertung der empirisch ermittelten Ergebnisse noch zeigen.
Dies wird – wie in der Einleitung zu diesem Text bereits erwähnt – in einem nachzureichenden Text besprochen

[16] zur Beschreibung der eingeführten Entscheidungslogik siehe vorallem den abgebildeten Algorithmus in Lohrum, Schneider & Willenborg (1999, S.9).
Einschränkend muss an dieser Stelle angemerkt werden, dass dem zitierten Text nicht entnommen werden kann, ob der eben angesprochene Algorithmus tatsächlich zum Einsatz kommt oder ob dieser ausschließlich als eine mögliche aber nicht realisierte Variante angeführt wurde

[17] eine Vergleichstabelle zur Gewichtung bibliographischer Kategorien findet sich in Kuberek (1999, S.26 (=Anlage 2))

Um die dabei erzielten Werte besser einschätzen und entsprechend kommentieren zu können, wurden bei der Berechnung weitere Kennzahlen festgehalten:

- die Zahl der Durchläufe pro Berechnung bei der Zerlegung und Verrechnung der Trigramm-Zeichenketten (dieser Vorgang ist eher rechenintesiv und kann u.a. durch den Einsatz schnellerer Hardware-Prozessoren verbessert werden),
- die Anzahl der zu prüfenden bibliographischen Datensätze (die Performanz des Lesens und Schreibens der Datensätze ist sehr von den Ein- und Ausgabe-Optionen der Datenbank abhängig und kann durch den Einsatz größerer Arbeitsspeicher („Memory“) sowie durch schnellere Festplatten verbessert werden),
- die Messung der Zeitdauer der einzelnen Schritte vom Datenentladen bis zur fertigen Ergebnisdatenbank

Bei der Gegenüberstellung der Ergebnisse wurde vorallem auch jene Differenzmenge gebildet, die sich durch den Umstand ergibt, dass die beiden Berechnungsmethoden voneinander abweichende Dublettenmengen ergeben.
Stichprobenartig wurden dieser Differenzmenge einzelne Datensätze entnommen. Anhand dieser wurde versucht zu erklären, wodurch sich im Einzelnen die Unterschiede in den berechneten Ergebnismengen zeigen.
Die stichprobenartige Überprüfung einiger weniger Ergebnisse kann innerhalb dieser Arbeit jedoch bloß als eine Überprüfung angesehen werden, die vorwiegend dazu dient, die eigene Intuition bezüglich möglicher passender oder fehlerhafter Ergebnisse zu schärfen. Eine vollständige und umfassende Auswertung der erzielten Ergebnismenge muss nachfolgen.

Ein wesentlicher Schwerpunkt der nachfolgend zu leistenden Interpretation wird natürlich einer der Ausgangspunkte dieses Vorhabens sein:
Nämlich vorrangig jene Titel bestimmen zu können, deren Aufnahmen voneinander so deutlich verschieden sind, dass sie als Dublette nicht in Frage kommen und damit in weiterer Folge wie neu einzubringende Datensätze geladen werden können.

6.1 Auswahl der Datensätze

Nachdem die Datensätze zur anstehenden Dublettenprüfung übernommen waren, wurden diese in strukturierter Form in eine relationale Datenbank geladen, deren Spaltenkonfiguration den zu prüfenden Kategorien entsprachen.
Anschließend wurden sämtliche Datensätze dupliziert und in die in *Kap.3* beschriebene normierte Form vor der weiteren Verarbeitung übergeführt.

Die Speicherung innerhalb einer Datenbank ist für den weiteren Vorgang weder wesentlich noch notwendig – jedoch ist die zu leistende Programmierung der Anwendungen damit deutlich vereinfacht:
Über den Index der Datenbank lässt sich relativ einfach ermitteln, welche weiteren Datensätze einen bestimmten Eintrag innerhalb eines Datensatzes tragen und sich dementsprechend als potentielle Treffer für eine Dublettenkontrolle qualifizieren.

Gesucht wurde im Index der Datenbank ausschließlich nach Begriffen, die nicht als Stoppwörter gekennzeichnet waren, sowie nach einem einfachen Algorithmus für jeden Datensatz mehrfach:
Das Auffinden vergleichbarer Datensätze wurde immer durch mindestens zwei Suchvorgänge voneinander verschiedener Begriffe aus dem Personennamen- und Sachtitelfeld angestrebt. Führte die Suche nach dem ersten Begriff zu keinem Treffer und lag ein weiterer suchbarer Begriff aus dem Titel vor, so wurde eine Suche auch nach diesem durchgeführt.
Im Extremfall konnte es also dazu kommen, dass bestimmte Titeldatensätze so oft gesucht wurden, bis die Anzahl ihrer Begriffe erreicht war.
Begriffe, die zu sehr großen Treffermengen führten, mussten aus der weiteren Verrechnung nicht ausgeschieden werden. Diese wurden jedoch als potentielle Stoppwörter markiert und entsprechend geprüft.[18] Bei einem Online-Verfahren zur Dublettenermittlung wäre dies aus zeitökonomischen Gesichtspunkten evtl. notwendig.

Nach erfolgter Dublettenberechnung wurden in der Datenbank zu jedem der zu prüfenden Datensätze die Satznummer der als ähnlich berechneten Datensätze hinzugefügt. Diese mussten anschließend intellektuell kontrolliert werden – während jene, die keine hinreichende Ähnlichkeit zu den bestehenden Titeldatensätzen aufweisten und formal korrekt vorlagen, übernommen wurden.

6.2 Messungen zur Performanz

6.2.1 Arbeitsmittel und Vorgehensweise

Sämtliche hier dokumentierte Ergebnisse sind auf handelsüblicher PC-Hardware erbracht worden. Zur besseren Einschätzung und Erstellung von Vergleichswerten sind folgende technische Angaben zur Hard- und Software zu berücksichtigen:
`CPU AMD Athlon64 3800+ Sockel AM2,`

[18] „entsprechend geprüft“ meint in diesem Fall, dass für jeden als potentielles Stoppwort markierten Begriff die Abfragesituation geklärt wurde, die entsteht, wenn nach diesem Begriff nicht weiter gesucht werden kann. Aufgrund der großen Rechenleistung moderner PCs stellte diese Klärung im Wesentlichen keinen besonders nennenswerten Umstand dar

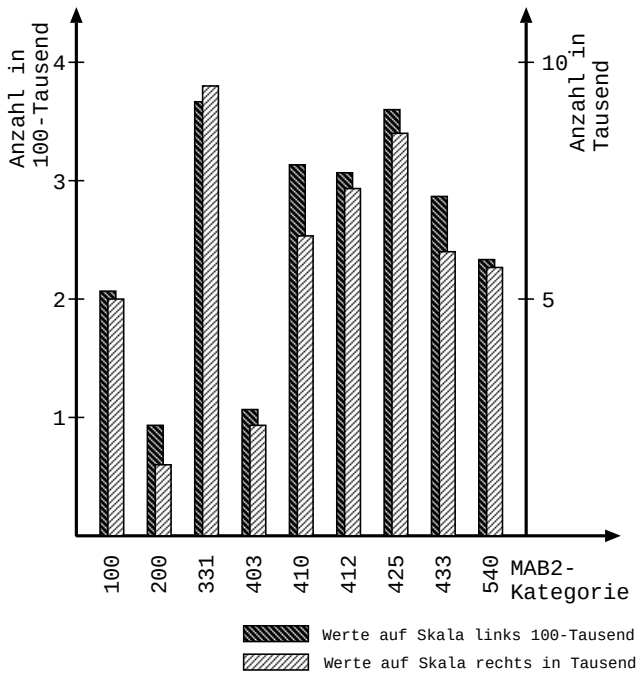

Abbildung 3: Verteilung jener Kategorien, die potentiell zur Dublettenerkennung herangezogen werden

`2 GB Hauptspeicher,`
`Festplatte Western Digital SATA 80 GB,`
`7500rpm, 8 MB Cache,`
`Betriebssystem Linux Ubuntu Dapper (32bit),`
`Perl v.5.8.7, Datenbank MySQL v.5.0.22`

Um die einzelnen Schritte vom Entladen der Quell-Datenbanken bis hin zur Feststellung der bibliographischen Dubletten möglichst einfach und modular zu halten wurde beschlossen, dass die einzelnen Schritte im Wesentlichen nach folgendem Schema ablaufen:
Die entsprechenden und zur jeweiligen Verrechnung heranzuziehenden Datensätze werden in der Datenbank durch Anwendung der spezifischen Routinen selektiert, entsprechend den jeweiligen Vorgaben aufbereitet und letztendlich in eine neue Tabelle geschrieben – oder aber in eine flache Textdatei gespeichert, die im nächsten Schritt wieder geladen wird.
Diese, aus rein pragmatischen Gründen getroffene Entscheidung, so vorzugehen, ist wohl nicht jene, die in jedem Fall als besonders schnell oder gar als elegant anzusehen ist. Dafür ist gewährleistet, dass die Datenbank immer nur jene Daten vorhalten muss, die auch tatsächlich gebraucht werden und das Verfahren in jedem Einzelschritt in sehr einfacher Weise wiederholt oder an jedem Punkt des Gesamtverfahrens fortgesetzt werden kann.[19]
Aus diesem Grund werden bei der Beschreibung der einzelnen Schritte sowohl die sich ergebenden Datenmengen als auch die Lade- und Entladezeiten der Datenbank angegeben.

6.2.2 Das Entladen der Daten aus der Quelldatenbank (`1h versus 48h, 8 GB Daten`)

Ein wesentlicher Schritt im Gesamtverfahren ist die Entscheidung darüber, welche Kategorieninhalte zur Berechnung der Datensatzähnlichkeit herangezogen werden.
Aufgrund m.u. verschiedener, zum Einsatz kommender Katalogisierungsregeln innerhalb der Vergleichsmengen wurde an dieser Stelle die Entscheidung getroffen, eine Vielzahl an inhaltsähnlichen Kategorien der Quelldatenbank zu entnehmen und erst durch eine nachgereihte und möglichst flexibel gehaltene Entscheidungslogik diese, entsprechend der vorgefundenen Datenlage, als Basis zur Entscheidung dafür heranzuziehen, welche Inhalte letztlich verwendet werden.

Die Methode, mit der das Entladen der zur Berechnung notwendigen Daten geschieht, hat gleichzeitig auch direkten und wesentlichen Einfluss auf die Performanz dieses Schrittes.
Das Extrahieren der o.a. Kategorien aus einem sog. „Full-Table-Export“ ist jene Methode, die im Wesentlichen sehr rasch durchzuführen ist: Sind die Daten einmal entladen, dauert das Selektieren der darin enthaltenen Kategorien nach dem oben beschriebenen Schema weniger als `eine Stunde`.[20] Dagegen dauert das sequentielle Entladen der gesamten Titeldaten im Online-Verfahren mittels der üblichen Datenbankschnittstellen bei gleicher Konstellation knapp über `48 Stunden`.
Bei der Anwendung der einen oder anderen Methode zum Entladen der Titelinformationen ist die Entscheidung zwischen Aktualität der Daten versus ihrer schnellen Verfügbarkeit zu treffen. Möglicherweise wird diese Entscheidung erst dann zu fällen sein, wenn hinreichende Erfahrungen im Umgang mit der Aufbereitung der Titeldaten vorliegen und Klarheit darüber herrscht, welche Kategorien konkret verrechnet werden sollen und sich zudem kaum mehr „Überraschungen“ aus der Datenlage des Quellsystems ergeben.

6.2.3 Umsetzung einer nachgereihten Entscheidungslogik (`8m21s, 72 MB Daten`)

Insgesamt wurden die Inhalte aus 28 bibliographischen Kategorien sowie die eindeutige Datensatz-ID den Quelldatenbanken entnommen und diese entsprechend ihrem Vorkommen jenen `12` Kategorien zugeordnet, die zur weiteren Berechnung herangezogen wurden. Die Vor-

[19] an dieser Stelle muss besonders auf den Umstand hingewiesen werden, dass die Berücksichtigung der entstehenden Datenmengen im hier beschriebenen Verfahren nicht unerheblich ist. Durch die Berechnung von Ähnlichkeiten aufgrund einer Stichwort-Auswahl entstehen Ergebnismengen, deren Umfang eine typische PC-Hardware leicht überfordert

[20] die Angabe der hier genannten Zahlen bezieht sich auf die Menge von `400.000` Datensätzen, die einer von vier Spalten aus einer einzigen Tabelle innerhalb eines relationalen Datenbanksystems entnommen sind

gehensweise dabei war nach folgendem Muster definiert:
KAT100, 100b, 100c, 100f, 359 → KAT100
Die in der Aufzählung vom Pfeil links stehenden Kategorien wurden in der angegebenen Reihenfolge für jeden Datensatz selektiert und jene, die (dieser Reihe folgend) „als erste“ mit Inhalten gefüllt war, wurde für die Berechnung letztlich herangezogen.

Für die weiteren Kategorien galt das folgende Schema:
KAT200, 200b, 200c → KAT200
KAT403 → KAT403
KAT410, 410a → KAT410
KAT412, 412a → KAT412
KAT425a, 425b, 425c → KAT425
KAT433, 433a, 433b → KAT433
KAT540a, 540b, 540 → KAT540

Die Kategorien für den Sachtitel 331 und 335 wurden zu einem Textstring zusammengefasst.
Zusätzlich wurden für die Auswertung der Reihentitel die Kategorien 453m, 453r und 455 exportiert und innerhalb der entsprechenden Datenbankschemata gespeichert. Mit der Information aus diesen Kategorien konnten letztlich die Bände bzw. Stücktitel mit den übergeordneten Reihentiteln zusammengeführt – und in weiterer Folge wie monographische Titel behandelt – werden.

Somit waren (in Begriffen relationaler Datenbanken gesprochen) 29 Datenbank-Spalten mit exakt 437.533 Datensätzen nach der o.a. Logik zu verarbeiten. In einer flachen Textdatei liegt die entladene Datenmenge in einem Umfang von 72 MB vor. Das Laden dieser Datei in eine Tabelle der Datenbank nimmt 1m44s in Anspruch. Diese Tabelle wurde von jenen bibliographischen Angaben bereinigt, die in der Datenbank als gelöscht (40.688 Datensätze, Löschdauer 1m02s) bzw. als „provisorische“ Aufnahmen (4.299 Datensätze, Löschdauer 5s) markiert waren.

Die Anwendung der eben beschriebenen Entscheidungslogik zur Verwendung einzelner Kategorieninhalte brachte eine Verarbeitungsdauer von 5m30s bei einer verbleibenden Anzahl von 392.616 Datensätzen mit sich und reduzierte die Datenmenge von eingangs 72 MB auf im Weiteren 53 MB. Das Laden dieser Menge in die Zieltabelle nahm 32s in Anspruch.

6.2.4 Generieren der Stoppworttabelle (4m77s, 699 Begriffe)

Wie in *Kap.6.1* bereits kurz angeführt beruht das Verfahren des Datensatzretrievals auf einer Stoppworttabelle, deren Erstellung im Wesentlichen in zwei Schritten geschah:

- in einem ersten Schritt wurden sämtliche Datensätze nach jenen Begriffen durchsucht, die als Stoppwörter durch die Zeichen „>>“ und „<<“ markiert waren. Das Sammeln dieser brachte 684 unterschiedliche Begriffe in die Stoppworttabelle ein und nahm 4m27s in Anspruch
- im zweiten Schritt wurden jene Begriffe in die Stoppworttabelle aufgenommen, deren Suche zu einem Ergebnis mit mehr als 2.000 Treffern (Titeldatensätzen) führte. Davon waren 15 weitere Begriffe betroffen, die einzeln überprüft und letztlich der Stoppworttabelle hinzugefügt wurden

Daneben wurden jene Begriffe mit in die Tabelle aufgenommen, die sich zwar weder im ersten noch im zweiten Schritt als Stoppwörter qualifizierten, jedoch in der Stichwortsuche zu deutlich mehr als 1.000 Treffern führten und als solche in anderen bibliographischen Datenbanken (wie z.B. im Bibliothekssystem *allegroC*) als Stoppwörter geführt werden.

6.2.5 Normalisieren / Normieren der Daten (46m47s, 392.616 Datensätze, 53 MB)

Dieser als „Normierung“ in *Kap.3* beschriebene Vorgang zählt neben der eigentlichen Berechnung zu den zeitlich und programmtechnisch aufwändigsten. Allein die Umsetzung der vorkommenden Zeichen in ihre Grundformen bedarf eines langwierigen Prozesses, der erst dann als beendet angesehen werden kann, wenn alle für die weitere Berechnung nicht zugelassenen Zeichen ersetzt sind.
Für die Normierung der zu verarbeitenden 392.616 Sätze wurden 46m47s benötigt. Die Überprüfung der Ergebnisse erfolgte mit einer eigenen Routine, mit der sämtliche Zeichen, die sich außerhalb der Menge der zugelassenen befanden, gesammelt und für die weitere Umsetzung dokumentiert wurden. Diese Überprüfung nahm 6m2s in Anspruch.

Die daraus entstandenen Daten hatten einen Umfang von 43 MB in einer flachen Textstruktur (gegenüber den 53 MB vor der Normierung.

7 Ergebnisdarstellung: Berechnung der Ähnlichkeiten (258h, 29.414.802 Datensätze, 4,3 GB)

Die umfassende Beschäftigung mit diesem Thema hat zum Ergebnis vor allem auch jenes, dass sich die Auswertung der Ergebnisse genauso umfang- und detailreich zeigt wie die Erbringung dieser. Die Ergebnisse zur Dublettenerkennung mittels N-Grammen wie sie in der angeführten Literatur vorliegen, lassen allein nicht dar-

auf schließen, welches der hier besprochenen Verfahren in der praktischen Anwendung und für die empirische Überprüfung geeignet oder gar am besten geeignet ist. Zudem zeigte sich mit zunehmendem Fortschritt immer deutlicher, dass ein Vergleich der publizierten Ergebnisse mit den Ergebnissen aus der hier verrechneten Datenmenge und Datenqualität nicht aussagekräftig sein kann: Die bereits publizierten Mess- und Erfahrungswerte beruhen überwiegend auf einer deutlich geringeren Menge an bibliographischen Daten[21], die zudem nur in den seltensten Fällen in großem Umfang einem „Echtsystem“[22] entnommen sind.
Aus diesen Gründen wurden in diesem Text allein die zum Einsatz kommenden Verfahren und Methoden sowie deren Performanz (vgl. auch *Fußnote4*) beschrieben.

Als ein wesentliches Ergebnis aus dem Erbringen der diese Verfahren beschreibenden Kennzahlen ist letztlich die Angabe der Berechnungsdauer sowie des dabei zu erbringenden Aufwands zur Nachstellung der empirischen Überprüfung anzusehen.

Die Berechnung im Batch-Verfahren, bei der die Menge von `3.100` bibliographischen Datensätzen (=die Anfragemenge) gegen eine größere Menge von `392.616` Datensätzen (=die Abfragemenge) mittels N-Grammen auf Dubletten geprüft wurde ergab, dass im Schnitt pro Anfrage `90` Abfrageergebnisse erzielt wurden, die durch den Dublettenvergleich geprüft werden mussten. Das ergibt letztlich für den Gesamtdurchlauf `279.000` Ergebnisdatensätze, die pro Datensatz `21` Einzelberechnungen beinhalten.[23]
Exportiert in eine flache Textdatei entspricht die Ergebnismenge einer Größe von `40.8 MB`. Die Zeitdauer, die für die Berechnung dieser Menge in Anspruch genommen wurde beträgt `2.4 Stunden`.

Um diese Ergebnisse im Weiteren besser einschätzen zu können und um für die noch zu erbringende Ergebnisauswertung eine deutlich umfangreichere Menge an Berechnungswerten vorliegen zu haben, wurde die Abfragemenge (die `392.616` Datensätze) zudem auch gegen sich selbst abgefragt.
Dies hatte zum Ziel, die in der Abfragemenge enthaltenen Dubletten zu identifizieren und letztlich zu einer dublettenfreien Menge zu gelangen. Diese könnte in weiterer Folge als Referenzmenge dienen, um einerseits weitere Daten gegen diese Menge zu prüfen. Andererseits ist eine umfangreiche dublettenfreie Menge an bibliographischen Datensätzen nützlich, um weitere (und möglicherweise notwendige) Veränderungen an den Berechnungsparametern günstig überprüfen zu können.

Bei der Berechnung der Menge von `392.616` Abfrage- und gleichzeitig gleich vielen Anfragedatensätzen ergaben sich im Schnitt `75` Abfrageergebnisse pro Datensatz. Dies wiederum ergab insgesamt `29,414.802` Ergebnisdatensätze mit den o.a. beinhalteten `21` Einzelberechnungen. Die entstandene Datenmenge entsprach `4.3 GB` in einer flachen Textdatei. Die Berechnungdauer für den Gesamtdurchgang der Dublettenberechnung betrug zusammengezählt `258 Stunden`, die auf mehrere Rechnern verteilt erbracht wurde. Bei einer Verteilung auf vier voneinander unabhängige Recheneinheiten konnte die Rechenleistung in `2.5 Tagen` erbracht werden.

[21] von zumeist einigen tausend bibliographischen Datensätzen
[22] darunter verstehe ich ein Bibliothekssystem, das sich seit vielen Jahren im Produktionseinsatz befindet
[23] die Einzelberechnungen, die hier durchgeführt wurden, setzen sich aus den drei oben genannten (Jaccard-Maß, euklidischer Abstand und der Ähnlichkeitswert des KOBV) sowie deren gewichtete und nicht gewichtete Funktionen zusammen

8 Literaturverzeichnis

8.1 Gedruckte Quellen

Cousins, Shirley Anne (1998): Duplicate detection and record consolidation in large bibliographic databases: the COPAC database experience. In: Journal of Information Science, Vol. 24, No. 4, S.231-240

Ferber, Reginald (2003): Information Retrieval. Suchmodelle und Data-Mining-Verfahren für Textsammlungen und das Web. dpunkt.verlag, Heidelberg

Goyal, Pankaj (1987): Duplicate Record Identification in Bibliographic Databases. In: Information Systems, Vol. 12, No. 3, S.239-242

Hylton, Jeremy A. (1996): Identifying and Merging Related Bibliographic Records. Master thesis, submitted to the Department of Electrical Engineering and Computer Science. M.I.T. Laboratory for Computer Science Technical Report 678
online:
`http://ltt-www.lcs.mit.edu/ltt-www/People/jeremy/thesis/MIT-LCS-TR-678.ps`

Hickey, Thomas B. (1979): Automatic Detection of Duplicate Monographic Records. In: Journal of Library Automation, Vol. 12, No. 2, S.125-142

Jones, William P. & Furnas, George W. (1987): Pictures of Relevance: A Geometric Analysis of Similarity Measures. In: Journal of the American Society for Information Science, Vol. 38, No. 6, S.420-442

Kirriemuir, John W. & Willet, Peter (1995): Identification of duplicate and near-duplicate full-text records in database search-outputs using hierarchic cluster analysis. In: Program, Vol. 29, No. 3, S.241-256

Kende, Jiří & Uhlig, Steffen (1995): Dublettenermittlung bei der Zusammenführung von Bibliotheken: (Nicht nur) ein statistisches Verfahren. In: Bibliothek, Forschung und Praxis, Jahrgang 19, Nr. 3, S.411-419

Kramer, André (2004): Falsche Fuffziger. Textplagiate per Software auf der Spur. In: c't. Zeitschrift für Computertechnik, Heft 21, S.176-181

Kuberek, Monika (1999): Dublettenbehandlung (Match- und Merge-Verfahren) in der KOBV-Suchmaschine – Grundlagen. Konrad-Zuse-Zentrum für Informationstechnik in Berlin (ZIB). Preprint SC 99-16
online:
`http://www.zib.de/Publications/Reports/SC-99-16.pdf`

Lohrum, Stefan; Schneider, Wolfram & Willenborg, Josef (1999): De-duplication in KOBV. Konrad-Zuse-Zentrum für Informationstechnik in Berlin (ZIB). Preprint SC 99-05
online:
`http://www.zib.de/Publications/Reports/SC-99-05.pdf`

Mandreoli, Federica; Martoglia, Riccardo & Tiberio, Paolo (2004): A document comparison scheme for secure duplicate detection. In: International Journal on Digital Libraries, Vol. 4, S.223-244

O'Neill, Edward T.; Rogers, Sally A. & Oskins, W. Michael (1993): Characteristics of Duplicate Records in OCLC's Online Union Catalog. In: Library Rsources & Technical Services, Vol. 37, No. 1, S.59-71

Reichart, Markus & Mönnich, Michael W. (1994): Dublettenkontrolle in bibliographischen Datenbanken. In: Bibliothek, Forschung und Praxis, Jahrgang 18, Nr. 2, S.193-216

Rusch, Beate (1999): Normierungen von Zeichenfolgen als erster Schritt des Match. Zur Dublettenbehandlung im Kooperativen Bibliotheksverbund Berlin-Brandenburg. Konrad-Zuse-Zentrum für Informationstechnik in Berlin (ZIB). Preprint SC 99-13
online:
`http://www.zib.de/Publications/Reports/SC-99-13.pdf`

Salton, Gerard (1968): Automatic Information Organization and Retrieval. (=McGraw-Hill Computer Science Series). McGraw-Hill Book Company, New York

Salton, Gerard & McGill, Michael J. (1987): Information Retrieval – Grundlegendes für Informationswissenschaftler. (=McGraw-Hill Texte). McGraw-Hill Book Company, Hamburg

Schneider, Wolfram (1999): Ein verteiltes Bibliotheks-Informationssystem auf Basis des Z39.50 Protokolls. Diplomarbeit, Technische Universität Berlin
online:
`http://wolfram.schneider.org/lv/diplom/diplom.pdf`

Zamora, E. M.; Pollock, J. J. & Zamora, Antonio (1981): The Use of Trigram Analysis for Spelling Error Detection. In: Information Processing & Management, Vol. 17, No. 6, S.305-316

Zengping Tian, Hongjun Lu, Wenyun Ji, Aoying Zhou & Zhong Tian (2002): An n-gram-based approach for detecting approximately duplicate database records. In: International Journal on Digital Libraries, Vol. 3, No. 4, S.325-331

8.2 Online-Quellen

`http://www.tillmann-wegst.de` : Wegst, Tillmann: Ähnlichkeitsbestimmung bei Zeichenketten

`http://www.uni-klu.ac.at/~hjele/publikationen/ngramme/haendisches/index.html` : Händische Ausfertigung der Berechnungen entsprechend den Werten aus *Abb.1*

`http://www.uni-klu.ac.at/~hjele/publikationen/ngramme/routinen/index.html` : Erstellte Programmroutinen zur Berechnung

Dr. Harald Jele ist Leiter der Abteilung EDV-Administration und -Entwicklung der Universitätsbibliothek Klagenfurt

Adresse:
Universität Klagenfurt
Universitätsstraße 65-67
9020 Klagenfurt, Österreich
Fax: 0043-463-2700-999599
E-Mail:harald.jele@uni-klu.ac.at