Visualisierungstechniken zum Information Retrieval für Textdokumente in der Biologie

Ralf Mikut[1], **Urban Liebel**[2]

[1]Forschungszentrum Karlsruhe GmbH, Institut für Angewandte Informatik,
[2]Forschungszentrum Karlsruhe GmbH, Institut für Toxikologie und Genetik,
E-Mail: ralf.mikut@kit.edu, urban.liebel@kit.edu

1 Motivation

Der aktuelle wissenschaftliche Stand in der Biologie ist in einer Vielzahl von Zeitschriftenartikeln (z.T. als Abstract oder Volltext verfügbar), Webseiten, Bildern und Datensatzbeschreibungen verstreut. Dabei kommen täglich neue Beiträge dazu, so dass die Verfolgung dieses Standes bereits bei kleinen Fachgebieten einen großen Aufwand erfordert.

Ein Beispiel hier ist das aktuelle Fachgebiet von Toxizitätsuntersuchungen mit Hilfe von Modellorganismen wie Zebrabärblingen. Die Suche nach den Begriffen ”zebrafish” und ”toxicity” mit der Metasuchmaschine Harvester (*http://harvester.fzk.de*) am 18. Juni 2009 ergab 104.000 Einträge bei Google, 75.800 Einträge bei Bing, 107.000 Einträge bei Clusty, 2.711 Einträge in der auf wissenschaftliche Seiten spezialisierten Suchmaschine Sciencenet (*http://sciencenet.fzk.de/*), 674 Einträge in Abstracts wissenschaftlicher Veröffentlichungen auf Medline (*http://medline.de*) und 16.200 Einträge auf *http://scholar.google.com*.

Eine rein schlagwortbasierte Suche stößt hier auf die bekannten Probleme:

1. Die Qualität der Suchergebnisse hängt stark von einer geschickten Komposition der Suchbegriffe ab. Oft werden viele irrelevante Dokumente gefunden, die die jeweiligen Schlagwörter enthalten. Hingegen werden relevante Dokumente selbst bei geringfügig abweichenden Begriffen (z.B. Synonymen) nicht gefunden. Semantische Suchanfragen (z.B. ”Finde Dokumente, die Methoden der Bioinformatik zur Modellierung der Toxizität bei Zebrabärblingen einsetzen”) sind nicht möglich.
2. Die Relevanzkriterien für das Sortieren der gefundenen Dokumente sind ein Kompromiss zwischen den Bedürfnissen aller Nutzer und des Suchmaschinenbetreibers (z.B. Werberelevanz usw.). Individuelle Kriterien wie z.B. die ausschließliche Suche nach wissenschaftlichen oder ”biologischen” Dokumenten können bei allgemeinen Suchmaschinen nicht eingestellt werden, weil die Relevanzkriterien nicht offengelegt sind. Folglich erscheinen viele für einen bestimmten Nutzer relevante Dokumente erst in hinteren Positionen der sortierten Liste.
3. Die Präsentation der Ergebnisse erfolgt in Form langer Listen mit Verlinkungen zu den gefundenen Dokumenten. Die Listen sind unübersichtlich und die Links gerade bei Volltexten wissenschaftlicher Veröffentlichungen nur teilweise zugänglich. Die Suche nach ähnlichen Dokumenten ist bestenfalls für Dokumente möglich, die in der gleichen Datenbank enthalten sind. Somit ist ein Vergleich eigener nicht in der Datenbank enthaltener Dokumente gegen Dokumente in der Datenbank nicht möglich, weil die

Formate für Dokumente und die Kriterien für Ähnlichkeit nicht offengelegt sind. Die schnelle Identifikation von Dokumenten, die seit der letzten Suche mit der gleichen Suchanfrage neu hinzu gekommen sind, wird nicht oder nur bedingt (z.B. in Form von Erscheinungsjahren in *scholar.google.com*) unterstützt. Somit fällt das Auffinden von Neuerscheinungen schwer.

4. Eine Antwort auf inhaltliche Fragen wie *Haben Erkrankte mit Tumor Typ B eine erhöhte Expression von Gen A?* und eine Wissensextraktion in Form von Regeln *Wenn ein Patient Tumor Typ B hat, ist Gen A immer stärker exprimiert* ist ein ungelöstes Problem.

Ein Weg zur Lösung des ersten und zweiten Problems ist die Suche nach semantischen Informationen in einem Dokument anstelle einer reinen Schlagwortsuche. Dazu gibt es zwei wesentliche Wege:

Ein erster Weg ist das manuelle Ergänzen maschinenlesbarer semantischer Informationen für Webdokumente (siehe z.B. Ansätze zum sogenannten Semantic Web [1] oder zu semantischen Tags in Wikipedia [2]). Allerdings erfordert diese Vorgehensweise einen großen manuellen Aufwand bei der Erstellung der Dokumente. Bislang unterstützen nur wenige Dokumente diesen Standard.

Ein zweiter Weg basiert auf einer vollautomatischen Kategorisierung von Dokumenten mit Hilfe einer geeigneten Ontologie. Hierbei werden Dokumente meist durch Vektoren repräsentiert, die die Häufigkeit der Wörter im Dokument enthalten (Vektorraum-Modell bzw. "Bag-of-words"-Modell [3]). Alternative Modelle verwenden n-Gramme (n-Gramm: Gruppe von n aufeinander folgenden Zeichen) anstelle von Wörtern [4, 5]. Die Ähnlichkeit zweier Dokumente oder zwischen Dokument und Kategorien wird aus den Ähnlichkeiten der Wort- oder n-Gramm-Häufigkeiten bestimmt. Dazu werden u.a. Support-Vektor-Maschinen [6], Clustering [7,8] oder Selbstorganisierende Karten [9] verwendet. Alle genannten Ansätze haben Probleme bei der Bewertung unterschiedlicher Wörter mit ähnlicher Bedeutung. Zur Lösung dieses Problems wird seit einigen Jahren Wikipedia als Quelle für kategorisierte Dokumente mit verwandtem Vokabular eingesetzt. Außerdem dient Wikipedia zunehmend als Thesaurus, Quelle für Ontologien und für mehrsprachige Suchen (siehe [10] für eine umfassende Übersicht).

Die Dimension der benötigten Häufigkeitsvektoren ist sehr groß (alle vorkommenden Wörter oder n-Gramme). Zudem ändert sich die Größe, wenn neue Wörter hinzukommen. Deswegen werden Methoden zur Dimensionsreduzierung genutzt (z.B. Singulärwertzerlegungen, Cluster mit ähnlichen Wörtern oder zufällige Projektionen der Wörter [11] bzw. n-Gramme [4,5,12] mit Hash-Tabellen). Bei Hash-Tabellen wird jedes Wort bzw. n-Gramm mit einer Transformation auf eine eindeutige Adresse abgebildet. Wenn die Anzahl der möglichen Adressen gegenüber der Anzahl der möglichen Wörter groß ist, reduziert sich die Anzahl der Kollisionen (verschiedene Wörter mit der gleichen Hash-Adresse) auf ein akzeptables Maß. Hash-Tabellen sparen Speicherplatz, generieren Dokumentrepräsentationen gleicher Länge und erleichtern den schnellen Zugriff in großen Datenbanken. Eine weitere Strategie ist die Entwicklung kleiner Hash-Tabellen, bei denen korrelierte Wörter und n-Gramme auf das gleiche Hash-Element abgebildet werden. Dazu eignen sich die Hauptkomponentenanalyse oder Multidimensional Scaling [13]. In [14] wird eine Hierarchie von Hash-Tabellen für ganze Dokumente eingesetzt, die in einem 20 oder 30 Bit langen Binärcode mit ähnlichen Hash-Codes für ähnliche Dokumente resultiert.

Die Option zur Suche nach Kategorien eröffnet wohl das größte Potenzial für semantische Suchanfragen. Ein Beispiel ist die Suche nach medizinischen Publikationen auf der Webseite *www.gopubmed.org* (Transinsight GmbH, Dresden, Germany). Hier wurden vorhandene Ontologien (GO: Gene Ontology, MeSH: Medical Subject Headings, UniProt: Universal Protein Resource) zum Kategorisieren von Dokumenten verwendet. Die Datenbank wird von der U.S. National Library of Medicine und dem National Institutes of Health (*http://www.ncbi.nlm.nih.gov/pubmed/*) bereitgestellt und umfasst Abstracts von Zeitschriftenartikeln. Die gefundenen Dokumente können nach Kategorien, Autoren, Orten, Schlagwörtern, Erscheinungsjahren usw. eingeteilt werden, wobei zusätzliche Statistiken und Auswertungen (wie z.B. Kooperationsnetze von Autoren usw.) zur Verfügung gestellt werden [15, 16]. Eine ähnliche Variante mit etwas reduziertem Leistungsumfang, aber ebenfalls mit Ontologien, existiert auch für Webseiten (GoWeb).

Zur Lösung des dritten Problems bietet sich eine semantisch orientierte Visualisierung einer größeren Sammlung von Dokumenten anstelle einer Auflistung an. Hierzu wurden unter anderem Graphen mit semantisch ähnlichen Dokumenten in unterschiedlichen Knoten [17], zweidimensionale Karten mit unterlegten Schlagwörtern zur Orientierung [18] oder Scatterplots mit farbcodierten Kategorien mit einer stochastischen Einbettung von Nachbarn [14] vorgeschlagen.

Das vierte Problem ist nur langfristig lösbar, weil es die höchsten Ansprüche an die semantische Analyse der Dokumente stellt. Erste Ansätze sind domänenspezifisch, z.B. zur Extraktion von Genen, Protein-Protein-Interaktionen und Protein-Funktionen in der molekularen Biologie (siehe [19, 20] für eine Übersicht). Zu diesen Themen finden regelmäßig Wettbewerbe wie BioCreAtIve [21, 22] statt, um die vorgeschlagenen Verfahren zu vergleichen.

Der Aufbau freier Suchmaschinen (wie z.B. YaCy, *http://yacy.net/*) mit einer offenen Indizierung eröffnet mittelfristig Perspektiven, Suchtechniken zu verbessern und insbesondere die Ergebnispräsentation stärker auf Biologen als Nutzergruppe zuzuschneiden. Allerdings sind bislang zwei Kernprobleme nicht gelöst:

- Es fehlt ein generell akzeptiertes Format zur Repräsentation von Dokumenten. Dieses Format sollte eine Kategorisierung ermöglichen, offen, recheneffizient, speicherplatzsparend, sprachunabhängig, plattformunabhängig, nachträglich und vollautomatisch berechenbar für existierende Dokumente sowie einsetzbar für beliebige Textdokumente inkl. Webseiten, Datensatzbeschreibungen, Volltexte und Abstracts von Artikeln (auch für nicht frei verfügbare Volltexte aus kommerziellen Datenbanken) sein.
- Es fehlt eine generell akzeptierte, einfach verständliche Technik zur Visualisierung der Semantik eines Dokuments.

Das Ziel dieses Beitrags besteht darin, neue Ansätze zur Lösung beider Probleme vorzuschlagen und diese Ansätze in ein Konzept einzubetten, das viele der oben diskutierten bereits bekannten Strategieelemente enthält (Abschnitt 2). Das Konzept basiert auf einem zweidimensionalen Hash-Ansatz, der Dokumente einheitlich als Bilder repräsentiert (Abschnitt 2.1). Als Kandidat für eine umfassende und regelmäßig aktualisierte Ontologie werden Wikipedia-Kategorien vorgeschlagen (Abschnitt 2.2). Anschließend werden Algorithmen zur Berechnung von repräsentativen Bildern für Kategorien (Abschnitt 2.3) und passende Maße für die Ähnlichkeit zwischen Dokumenten und Kategorien (Abschnitt 2.4) präsentiert, die als Basis für eine semantische Visualisierung dienen. Abschließend werden Implementierungsaspekte diskutiert (Abschnitt 2.5). Die vorgeschlagenen Methoden werden dann auf biologische Dokumente angewendet (Abschnitt 3).

2 Methoden

2.1 Einheitliche Repräsentation von Dokumenten als Hash-Bilder

In einem ersten Schritt werden alle Wörter ($n = 1, \ldots, N_w$) eines Dokuments D_k extrahiert. Dazu werden bestimmte Zeichen als Wortgrenzen interpretiert (Leerzeichen, Punkte, Frage- und Ausrufezeichen, Punkte, Kommas, Semikolons, Klammern, Zeilenende). In einem Wort werden nur die Zeichen "a-z" (ASCII 97-122), "0-9" (48-57), "-" (45) und "_" (95) akzeptiert. ASCII Zeichen für Großbuchstaben "A-Z" werden in Kleinbuchstaben "a-z" umgewandelt. Alle anderen Zeichen (darunter die deutschen Umlaute) werden durch '_' (95) ersetzt. Diese Ersetzung erleichtert den Umgang mit verschiedenen Codierungen für ASCII (z.B. Codierung 25, 188 oder 255 für 'ß'). Somit wird das Wort "Zebrabärbling" in "Zebrab_rbling" abgebildet. Lange Wörter werden auf eine Länge von L_{max} Zeichen begrenzt (hier: $L_{max} = 50$). Wörter mit nur einem Zeichen werden ignoriert.

Die Häufigkeiten der Wörter in einem Dokument werden in einer Hash-Tabelle $\mathbf{F}$ mit der Dimension $H \times H$ (hier: $H = 512$) abgelegt. Für jedes Wort in einem Dokument wird $F_{c_1[n],c_2[n]}$ um Eins inkrementiert. Die Hash-Adresse des Worts $c_d[n]$ wird durch eine Multiplikation des ASCII Codes $w_l[n]$ für das l. Zeichen im n. Wort (Wortlänge: L Zeichen) mit einem oder mehreren ($d = 1, \ldots, D$, hier $D = 2$) Hash-Vektoren $\mathbf{h}_d$ berechnet:

$$c_d[n] = 1 + \mathrm{rem}\left(\sum_{l=1}^{L} w_l[n] \cdot h_{ld}, H\right). \tag{1}$$

Der Operator rem berechnet den Rest einer Ganzzahldivision. Die verwendeten Hash-Parameter h_{ld} wurden mit einem Zufallsgenerator ermittelt und sind in Tabelle 1 angegeben. Eine einheitliche Verwendung dieser Parameter sorgt für eine einheitliche Repräsentation der Dokumente.

Beispielsweise wird das Wort "Cell" zunächst in Kleinschreibung umgewandelt: "cell". Der resultierende ASCII-Code lautet $\mathbf{w} = (99, 101, 108, 108)$. Die Abbildung in die Hash-Tabelle mit $H = 512, D = 2$ ergibt

$$\begin{aligned} c_1 &= 1 + \mathrm{rem}(486 \cdot 99 + 311 \cdot 101 + 456 \cdot 108 + 234 \cdot 108, 512) = 446 \\ c_2 &= 1 + \mathrm{rem}(118 \cdot 99 + 249 \cdot 101 + 390 \cdot 108 + 9 \cdot 108, 512) = 52. \end{aligned}$$

Für jedes Auftreten von "cell" oder "Cell" wird der Wert von $F_{446,52}$ um Eins erhöht. Die gewählte Hash-Tabelle besteht aus $H^2 = 262144$ Elementen. Diese Größe reduziert das Risiko zufälliger Kollisionen, allerdings ist mit einer bestimmten Anzahl von Kollisionen zu rechnen (Beispiel: Hash-Element $F_{51,383}$ für die Wörter "brain" und "Canada").

Diese Hash-Strategie ist auf textbasierte Dokumente in beliebigen Sprachen und Dateiformaten (z.B. nur Text, extrahierter Text aus PDF- oder Word-Dokumenten, HTML usw.) anwendbar. Sogenannte Stoppwörter (häufige Wörter wie Artikel, verbleibende Steuerelemente wie HTML-Tags) werden hier noch nicht unterdrückt. Diese Aufgabe wird später durch die Ähnlichkeitsmaße in Abschnitt 2.4 mit gelöst. Eine Extraktion des Wortstamms durch das Abschneiden von Endungen wie "-en" (deutsch) bzw. "-ing", "-s" (englisch, siehe z.B. [23]) ist zwar prinzipiell möglich, wird aber wegen der angestrebten Sprachunabhängigkeit und der Reduzierung der Auswirkungen von Hash-Kollisionen bewusst nicht durchgeführt.

Position eines Zeichens l	Hash-Parameter h_{l1}	Hash-Parameter h_{l2}	Position eines Zeichens l	Hash-Parameter h_{l1}	Hash-Parameter h_{l2}
1	486	118	26	99	349
2	311	249	27	155	277
3	456	390	28	77	357
4	234	9	29	194	440
5	421	228	30	437	304
6	315	405	31	254	461
7	472	378	32	421	330
8	90	208	33	419	338
9	479	469	34	175	148
10	210	458	35	175	273
11	30	181	36	372	158
12	416	5	37	429	291
13	71	104	38	190	360
14	102	309	39	280	228
15	139	102	40	356	318
16	8	382	41	407	490
17	228	477	42	268	451
18	239	214	43	89	502
19	433	269	44	139	129
20	104	344	45	448	378
21	429	10	46	70	6
22	349	194	47	458	102
23	426	257	48	153	339
24	363	220	49	146	240
25	156	97	50	33	506

Tabelle 1: Parameter für die Berechnung der Hash-Werte in (1)

In einem nächsten Schritt wird für jedes Dokument aus der Hash-Tabelle der Häufigkeiten $\mathbf{F}$ ein korrespondierendes Bild generiert, das als Grauwert-Matrix $\mathbf{I}$ abgespeichert wird. Dazu wird jede Hash-Adresse als Pixel-Code interpretiert. Die Helligkeit des Pixels der Position (x, y) wird mittels

$$I_{xy} = \mathrm{ceil}\left(P_{max} \cdot \frac{\log\left(1 + F_{xy}\right)}{\max_{x,y} \log\left(1 + F_{xy}\right)}\right) \tag{2}$$

berechnet. Der Operator ceil rundet den Wert auf, um für alle vorkommenden Wörter einen Mindestwert von Eins zu garantieren. Der Logarithmus wurde gewählt, um die starken Unterschiede der Worthäufigkeiten zwischen 1 und einigen Tausend geeignet abzubilden. Der maximale Grauwert wird für jedes Dokument auf einen einheitlichen Wert P_{max} (hier: $P_{max} = 255$ als Codierung für weiß) gesetzt. Daraus ergibt sich eine 8-Bit-Codierung, siehe Beispiel in Abschnitt 3.1. Wenn für die korrespondierende Worthäufigkeit $F_{xy} = 0$ gilt, erhält das Pixel einen Grauwert $I_{xy} = 0$ und erscheint schwarz. Die Anzahl der nicht schwarzen Pixel ist eine untere Abschätzung für die Anzahl verschiedener Wörter in einem Dokument, eventuelle Unterschiede resultieren aus Hash-Kollisionen.

Die resultierenden Bilder werden im weitverbreiteten Portable Network Graphics (PNG) Format abgespeichert. PNG ist ein Bitmap-Format mit einer verlustfreien Kompression. Bei den verwendeten 8-Bit-Grauwerten sind die Bilder relativ speicherplatzspa-

rend. Beispielsweise wird ein Verzeichnis mit kurzen Abstracts (Größe der Abstracts: 1.30 ± 0.47 kByte) auf PNG-Bilder der Größe 0.70 ± 0.10 kByte abgebildet. Ein Verzeichnis mit PDF-Dokumenten (Größe 1.04 ± 2.52 MByte) wird zunächst in extrahierte Textdokumente der Größe 75.8 ± 143.6 kByte und dann in PNG-Bilder der Größe 4.92 ± 3.40 kByte konvertiert. Das größte Dokument (ein komplettes Buch mit 258 Seiten, PDF-Größe 7.97 MByte) hat als PNG-Bild eine Größe von 30 kByte.

Eine Rückübersetzung von Pixeln in Wörter ist für die Analyse von dominanten Wörtern in Bildern interessant. Für diese Aufgabe wird bei der Umwandlung von Dokumenten in Bilder ein Wörterbuch mit korrespondierenden Wörtern zu Pixel-Adressen angelegt. Das Wörterbuch hat allerdings einige Mehrfachzuordnungen durch die angesprochenen Hash-Kollisionen.

2.2 Definition von Kategorien

Kategorien werden im Folgenden über eine Menge an zugehörigen Bildern definiert, die Dokumente oder Subkategorien repräsentieren. Dazu bieten sich zwei Varianten an:

- Erstellung von *nutzerspezifischen* Kategorien (z.B. auf Basis einer Verzeichnisstruktur mit enthaltenen Dokumenten, wobei die Verzeichnisse als Kategorien dienen),
- Verwendung von *standardisierten* Kategorien (z.B. auf Basis der Kategorien von Wikipedia, siehe [24]).

Wikipedia enthält eine große Sammlung kategorisierter Dokumente in verschiedenen Sprachen und deckt einen beträchtlichen Teil des vorhandenen Wissens in der Welt ab. Wikipedia wird ständig aktualisiert und erweitert, was eine regelmäßige Aktualisierung der Kategorien ermöglicht (z.B. in Form jährlicher Aktualisierungen).

Die hier verwendete Kopie der englischen Wikipedia vom November 2008 umfasst ca. 2600000 Dokumente und 384544 als gerichteter Graph organisierte Kategorien. Auf der obersten Ebene existieren 25 Kategorien (engl. *main topic classifications*): *mathematics, people, science, music, law, history, geography, culture, agriculture, politics, social sciences, nature, technology, education, computing, health, business, belief, humanities, chronology, visual arts, crafts, environment, arts, language*. Der Graph ist nicht streng hierarchisch, z.B. ist *applied sciences* sowohl auf der obersten Ebene als auch als Kind von *science* auf der 2. Ebene vertreten. In solchen Fällen wurde die tiefere Ebene verwendet. Die Kategorien ändern sich von Zeit zu Zeit. Beispielsweise existierten im August 2009 nur noch 22 Kategorien auf der obersten Ebene (neu: *culture*, auf untere Ebenen verschoben: *music, social sciences, visual arts, crafts*). Die aktuellste Version ist unter *http://en.wikipedia.org/wiki/Kategorie:Main_topic_classifications* verfügbar.

Für die folgenden biologischen Anwendungen wurde ein Kategorie-Baum mit 346 Kategorien eingesetzt (Basis: November 2008). Er enthält alle 25 Kategorien der obersten Ebene und ausgewählte Subkategorien bis zur 5. Ebene. Die 2. Ebene umfasst 21 Subkategorien von *science* (z.B. *applied science, computational science, cybernetics*), die 3. Ebene 21 Subkategorien von *applied science* (z.B. *management*) und 5 Subkategorien von *natural science* (z.B. *biology*). Auf Ebene 4 gibt es 38 Subkategorien von *biology* (z.B. *genetics*) und auf Ebene 5 236 weitere Subkategorien der Ebene 4 (z.B. *viral life cycle*). Einige allgemeine Kategorien wurden ignoriert, z.B. Kategorien mit Strings wie ”Articles with”, ”All pages needing” usw. Für die Ebene 0 wird eine zusätzliche Basiskategorie *total* eingeführt, die alle Kategorien der obersten Ebene zusammenfasst.

2.3 Repräsentation von Kategorien als Hash-Bilder

Im Rahmen des vorgeschlagenen Konzepts werden Kategorien ebenfalls als Bild repräsentiert. Die Kategorie-Bilder werden durch ein gewichtetes Mittel aller zugehörigen Bilder für Dokumente und Subkategorien ermittelt:

$$\mathbf{I}(\mathbb{C}_c) = \text{round}\left(\frac{\sum\limits_{k \text{ mit } \mathbf{I}_k \in \mathbb{C}_c} w_k \cdot \mathbf{I}_k}{\sum\limits_{k \text{ mit } \mathbf{I}_k \in \mathbb{C}_c} w_k}\right). \tag{3}$$

$I_k \in \mathbb{C}_c$ bedeutet, dass das Bild $\mathbf{I}_k$ zur Kategorie $\mathbb{C}_c$ gehört. $\mathbf{I}_k$ repräsentiert entweder ein zugehöriges Textdokument D_k (z.B. *cell_biology.txt* für die Kategorie *cell biology*) oder eine Subkategorie $\mathbb{C}_k$ (z.B. *cell signaling* als Subkategorie von *cell biology*). Der Operator round rundet alle Pixelwerte. Anstelle des Rundungsoperators, der u.U. seltene Pixel mit Werten zwischen Null und Eins auf Null abrundet, kann auch der immer aufrundende Operator ceil verwendet werden, der seltene Pixel erhält.

w_k ist ein Wichtungsfaktor für das k. Bild. Solche Wichtungsfaktoren bevorzugen Bilder mit mehr Information, z.B. auf der Basis längerer Dokumente oder mit vielen Subkategorien. Ein guter Indikator ist der kleinste Grauwert mit einem Wert größer Null in einem Bild $P_{min}(\mathbf{I}_k)$. Er fällt mit der Länge des Dokuments (einmal vorkommende Wörter mit kleinen relativen Häufigkeiten) und mit der Anzahl von Dokumenten in einer Subkategorie (seltene Wörter, die nur in einer Subkategorie vorkommen) durch Anwenden von (3). Deshalb wird der folgende Wichtungsfaktor genutzt:

$$w_k = 1/P_{min}(\mathbf{I}_k). \tag{4}$$

Dennoch enthalten einige Kategorien nur wenige Dokumente und Subkategorien. Die resultierenden Bilder haben dann relativ große Grauwerte, was bei der späteren Ähnlichkeitsberechnung zu unerwünschten Präferenzen zu diesen schlecht abgedeckten Kategorien führen kann. Deshalb werden nach der Kategorie-Berechnung die Bilder mit Grauwerten $\{0, P_{min}, \ldots, P_{max}\}$ auf Grauwerte von $\{0, 1, \ldots, P_{max}\}$ normalisiert (hier: $P_{max} = 255$). Das erfolgt durch die Operation:

$$\mathbf{I}_{norm} = \text{round}\left(\frac{P_{max}}{(P_{max})^{\alpha_{norm}}} \cdot \mathbf{I}^{\alpha_{norm}}\right) \text{ mit } \alpha_{norm} = \frac{\log(P_{max})}{\log(P_{max}) - \log(P_{min})}. \tag{5}$$

2.4 Ähnlichkeit zwischen Dokumenten und Kategorien

Das nächste Ziel ist die Bestimmung paarweiser Ähnlichkeiten zwischen *Bildern*, die einzelne Dokumente und Kategorien repräsentieren. Die Hauptanwendung ist die Kategorisierung von Dokumenten.

Klassische Ähnlichkeitsmaße wie Worthäufigkeit in einem Dokument vs. Inverse Häufigkeit des Vorkommens von Wörtern in allen Dokumenten TF-IDF (engl. term frequency - inverse document frequency) [7] sind nicht direkt nutzbar, weil in (2) bereits logarithmische Verhältnisse relativer Häufigkeiten berechnet werden und die benötigten Werte für TF folglich nicht mehr verfügbar sind. Außerdem steht die Anzahl der Dokumente in der Datenbank, in denen ein Wort wenigstens einmal vorkommt (IDF), nicht zur Verfügung. Deshalb werden beide Maße durch die verwandten Maße der Grauwerte in einem

Dokument bzw. einer Kategorie (statt TF) und den inversen Grauwert für das Bild der Basiskategorie $\mathbf{I}(\mathbb{C}_0)$ (statt IDF) ersetzt. Dieses Bild stellt einen mittleren Grauwert für alle Kategorien dar und hängt stets von der Sprache und den existierenden Kategorien ab. Daraus entsteht eine Wichtungsmatrix $\mathbf{W}$ mit Elementen

$$W_{xy} = \frac{1}{1 + I_{xy}(\mathbb{C}_0)}, \tag{6}$$

die seltene Wörter mit niedrigen Grauwerten bevorzugt. Alternative Wichtungsstrategien (z.B. Verhältnisse zwischen verschiedenen Kategorien zum Identifizieren charakteristischer Pixel) wurden bislang nicht umfassend untersucht.

Ein Maß für die Ähnlichkeit zwischen zwei Bildern sind hohe Grauwerte für gleiche Pixel. Dabei sind nur die Pixel bedeutsam, die heller als im Bild der Basiskategorie und die in der Basiskategorie möglichst selten sind. Diese beiden Eigenschaften werden durch Subtrahieren des Bildes der Basiskategorie und Wichten mit (6) umgesetzt:

$$I^*_{xy} = \max\left(0, \mathrm{round}\left(\left(I_{xy,Norm} - I_{xy}(\mathbb{C}_0)\right) \cdot W_{xy}\right)\right). \tag{7}$$

Die Operation wird für jedes Dokument- und jedes Kategorie-Bild einmal ausgeführt, wobei die mit (5) normalisierten Bilder verwendet werden. Dabei ist zu beachten, dass wegen des Abspeicherns als PNG wieder ganzzahlige Grauwerte entstehen. Der Rundungsoperator setzt eine relativ große Anzahl von Pixeln mit kleinen Grauwerten auf einen Grauwert Null, sofern sie im Bild der Basiskategorie keinen Wert von Null haben, was zu $w_{ik} < 1$ führt. Der resultierende Effekt wirkt wie eine Merkmalsreduktion.

Wenn dieser Effekt nicht erwünscht ist (z.B. bei sehr kurzen Dokumenten, bei denen auch einzelne Wörter wichtig sind), kann anstelle von (7) für die Dokument-Bilder auf die Normierung mit (5) verzichtet werden:

$$I^*_{xy} = \max\left(0, \mathrm{round}\left(\left(I_{xy} - I_{xy}(\mathbb{C}_0)\right) \cdot W_{xy}\right)\right). \tag{8}$$

Außerdem beeinflussen zufällige Schwankungen der Worthäufigkeiten und Hash-Kollisionen die Ähnlichkeiten. Einen erheblichen Beitrag zu Hash-Kollisionen liefert die relativ große Anzahl seltener Wörter in einem Dokument (z.B. Namen, Orte, E-Mail-Adressen, spezielle mathematische Symbole, Wortfragmente durch PDF zu Text-Umwandlungen, nicht erkannte Hypertext-Tags usw.). Das Risiko ist besonders dann hoch, wenn eine Kategorie bezogen auf die Hash-Größe viele nicht schwarze Pixel aufweist. Das gesamte Risiko wird mit einem Maß für zufällige Pixel-Ähnlichkeiten abgeschätzt:

$$S_{rand,c} = \alpha_{Heur,abs} + \frac{\alpha_{Heur,rel}}{H^2} \sum_{x=1}^{H} \sum_{y=1}^{H} I^*_{xy}(\mathbb{C}_c). \tag{9}$$

Der Wert $\alpha_{Heur,abs}$ muss an zufällige Schwankungen der Worthäufigkeiten angepasst werden. Der zweite Term schätzt den Beitrag der Hash-Kollisionen durch den Parameter $\alpha_{Heur,rel}$, die Hash-Größe H und die Summe der Grauwerte in einem Kategorie-Bild. Im Folgenden werden für alle Kategorien die einheitlichen Einstellungen $\alpha_{Heur,abs} = 0.01, \alpha_{Heur,rel} = 5$ verwendet.

Das Maß für zufällige Pixel-Ähnlichkeiten in $S_{rand,c}$ in (9) muss nur einmal pro Kategorie berechnet werden. Es schätzt den Beitrag *pro Pixel* in einem Dokument ab, folglich wird

das korrespondierende Maß für zufällige Beiträge *pro Dokument* noch mit der Anzahl der nicht schwarzen Pixel pro Dokument $N_{Pix}(D_k)$ multipliziert:

$$S_{rand,kc} = \alpha_{min} + S_{rand,c} \cdot N_{Pix}(D_k). \tag{10}$$

Der Wert α_{min} (hier: $\alpha_{min} = 2$) trägt dazu bei, Probleme mit extrem kurzen Dokumenten zu vermeiden. Im nächsten Schritt misst

$$S_{cooc,kc} = \sum_{x=1}^{H} \sum_{y=1}^{H} \min \left(I^*_{xy}(D_k), I^*_{xy}(\mathbb{C}_c), S_{rand,kc} \right) \tag{11}$$

die Pixel-Ähnlichkeit zwischen den Bildern des Dokuments D_k und der Kategorie $\mathbb{C}_c$. Alle Pixel, die in beiden Bildern nicht schwarz sind, tragen zur Ähnlichkeit bei. Der Minimum-Operator wählt den niedrigeren Grauwert beider Bilder. Der Term $S_{rand,kc}$ beschränkt den Beitrag eines Pixels auf den Wert der zufälligen Pixel-Ähnlichkeit des Dokuments, um eine Dominanz durch einzelne extrem helle Pixel zu vermeiden, die u.U. durch Hash-Kollisionen entstehen. Die Berechnung von (11) erfordert nur Integer- und Extrema-Operationen und kann auf spezielle Grafikhardware ausgelagert werden.

Aus der Pixel-Ähnlichkeit in (11) und der zufälligen Pixel-Ähnlichkeit des Dokuments in (10) berechnet sich nun die Ähnlichkeit zwischen Dokument und Kategorie:

$$S_{kc} = \max \left(0, \frac{S_{cooc,kc}}{S_{rand,kc}} - 1 \right). \tag{12}$$

Ein Wert von Null bedeutet keine Ähnlichkeit, steigende Werte zeigen eine zunehmende Ähnlichkeit bezüglich des verwendeten Vokabulars an. Die Genauigkeit verbessert sich mit zunehmender Dokumentlänge und mit der Spezifik des Vokabulars einer Kategorie. Verwandte Kategorien führen zu vergleichbaren Ähnlichkeiten der Dokumente.

Eine scharfe Kategorisierung entscheidet zugunsten der ähnlichsten Kategorie:

$$\hat{c}_{opt}(\mathbf{I}_k) = \mathrm{argmax}_c \, S_{kc}. \tag{13}$$

Für eine hierarchische Kategorisierung eines Dokuments muss die Strategie nur leicht modifiziert werden:

1. Für jede Kategorie $\mathbb{C}_c$ werden gezielt die charakteristischen Pixel gesucht, die häufiger als in ihren direkten Oberkategorien und deren Oberkategorien sind. Dazu wird das Bild der Basiskategorie $\mathbf{I}(\mathbb{C}_0)$ in (6) und (7) durch ein spezifisches Bild $\mathbf{I}(\mathbb{C}_{0c})$ für jede Kategorie $\mathbb{C}_c$ ersetzt. Dieses Bild entsteht durch eine Maximum-Verknüpfung aller Bilder von direkten oder indirekten Oberkategorien mit

   $$I_{xy}(\mathbb{C}_{0c}) = \max_{i \in \mathrm{Par}(c)} I_{xy}(\mathbb{C}_i). \tag{14}$$

   Der Menge $\mathrm{Par}(c)$ umfasst alle Oberkategorien der c-ten Kategorie auf den höheren Hierarchieebenen, inkl. der Basiskategorie auf der nullten Ebene. Falls nur eine Ebene existiert, gilt folglich $\mathbf{I}(\mathbb{C}_{c0}) = \mathbf{I}(\mathbb{C}_0)$.
2. Start mit der höchsten Kategorie-Ebene (Ebenen-Zähler auf $L_{hierarchy} = 1$ setzen).
3. Ähnlichkeiten für alle Kategorien der aktuellen Ebene $L_{hierarchy}$ berechnen und beste Kategorie mit (13) bestimmen.

4. Alle Subkategorien (mit $L_{hierarchy} + 1$) für die beste Kategorie und alle weiteren Kategorien aus Schritt 3 mit vielversprechenden Ähnlichkeiten

$$\frac{S_{cooc,kc}}{S_{rand,kc}} > S_{thres} \tag{15}$$

   bestimmen (z.B. das hier gewählt wenig selektive Niveau $S_{thres} = 0.5$, um nicht vorzeitig möglicherweise interessante Kategorien auf tieferen Ebenen auszuschließen).
5. Stopp wenn in Schritt 4 keine Subkategorien mehr gefunden werden, sonst Ebenen-Zähler erhöhen ($L_{hierarchy} = L_{hierarchy} + 1$) und mit Schritt 3 fortsetzen.

Diese Strategie liefert auf jeder Ebene die beste Kategorie. Zudem kann die Zahl der Ähnlichkeitsberechnungen reduziert werden, weil nur Ähnlichkeiten für vielversprechende Subkategorien ausgewertet werden. Bei einer Anzahl von C nicht hierarchisch angeordneten Kategorien müssen hingegen C Ähnlichkeiten bestimmt werden.

2.5 Implementierungsaspekte

Alle Algorithmen wurden in MATLAB (The MathWorks, Inc.) implementiert und in ein Text-Mining-Erweiterungspaket der MATLAB-Toolbox Gait-CAD [25] integriert. Gait-CAD kann unter *http://sourceforge.net/projects/gait-cad/* frei bezogen werden, die vorläufige Version des Erweiterungspakets ist per Mailanfrage an die Autoren erhältlich. Die Konvertierung von PDF in Textdateien wurde mit der Open-Source-Toolbox PDFTEXT (Version 3.02, Glyph & Cog, LLC) durchgeführt.

Die beschriebenen Algorithmen eignen sich gut für eine Parallelisierung und eine dateiorientierte Handhabung. Das ist eine notwendige Voraussetzung für die Beherrschung großer Datenbanken mit Millionen Dokumenten wie Wikipedia und biologische Suchmaschinen.

3 Experimentelle Ergebnisse

3.1 Illustratives Beispiel

Im folgenden illustrativen Beispiel wird die Hash-Repräsentation gemäß Abschnitt 2.1 und die Kategorisierung für ein Dokument beschrieben (Titel und Abstract von [26]):

Zebrafish embryos as models for embryotoxic and teratological effects of chemicals: The experimental virtues of the zebrafish embryo such as small size, development outside of the mother, cheap maintenance of the adult made the zebrafish an excellent model for phenotypic genetic and more recently also chemical screens. The availability of a genome sequence and several thousand mutants and transgenic lines together with gene arrays and a broad spectrum of techniques to manipulate gene functions add further to the experimental strength of this model. Pioneering studies suggest that chemicals can have in many cases very similar toxicological and teratological effects in zebrafish embryos and humans. In certain areas such as cardiotoxicity, the zebrafish appears to outplay the traditional rodent models of toxicity testing. Several pilot projects used zebrafish embryos to identify new chemical entities with specific biological functions. In combination with

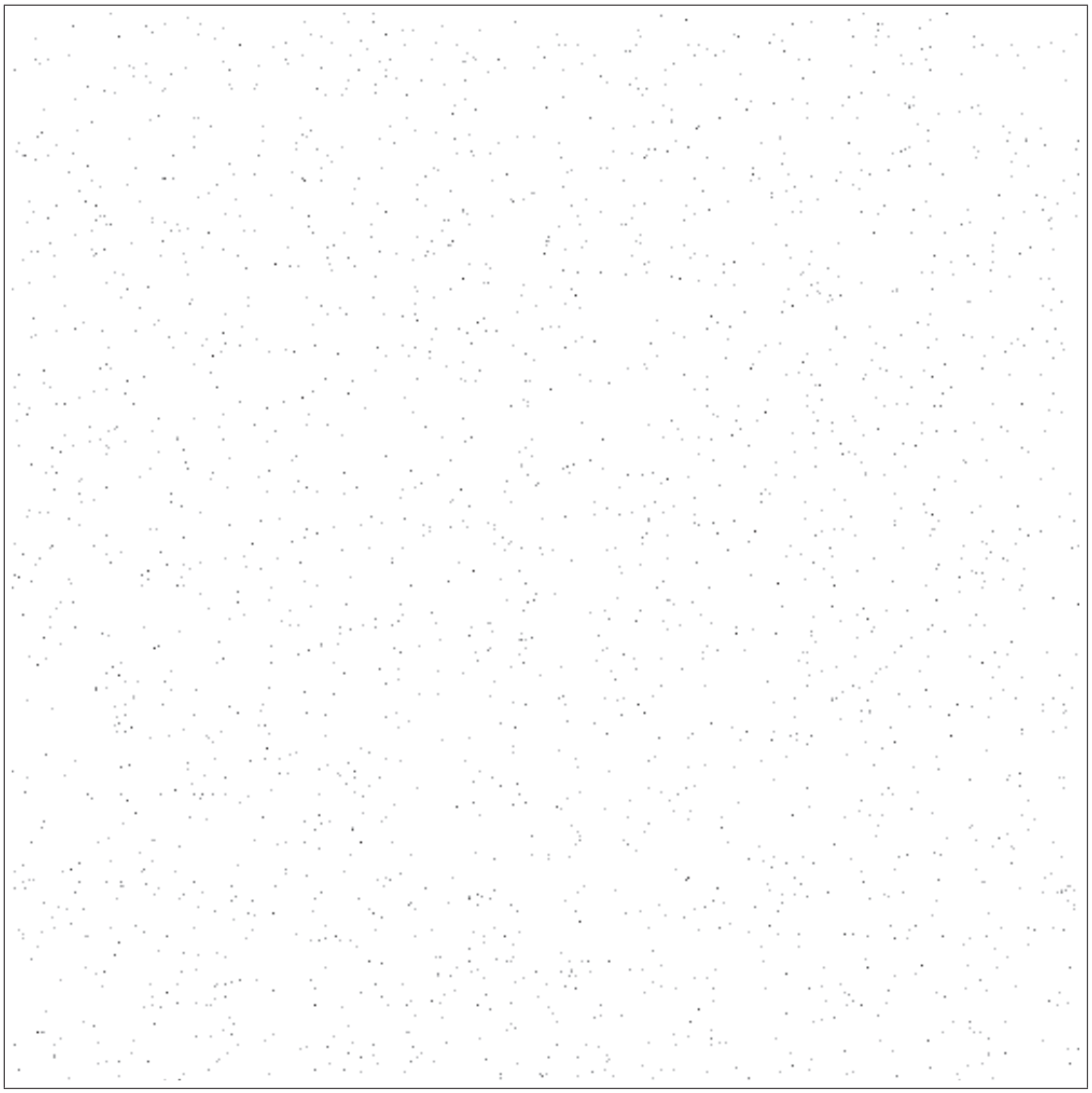

Bild 1: Resultierendes PNG-Bild für den Volltext von [26]. Die Koordinate $(x = 1, y = 1)$ befindet sich oben links, die Koordinate $(x = 512, y = 512)$ unten rechts. Zur Verbesserung der Sichtbarkeit in der Druckversion wurden die Grauwerte invertiert und mit einer Gammakorrektur bearbeitet ($\gamma = 0.5$). Wegen dieser Invertierung codieren besonders dunkle Pixel häufige Wörter.

the establishment of transgenic sensor lines and the further development of existing and new automated imaging systems, the zebrafish embryos could therefore be used as cost-effective and ethically acceptable animal models for drug screening as well as toxicity testing.

Der Abstract enthält 104 verschiedene Wörter. Das häufigste Wort ist ”the” (12x, wird auf Grauwert 255 abgebildet), gefolgt von ”and’ und ”of” (10x, Grauwert 239) sowie ”zebrafish” (7x, Grauwert 207), siehe Tabelle 2. Alle einmal vorkommenden Wörter werden auf einen Grauwert von 69 abgebildet. Die unerwünschten allgemeinen Wörter werden durch Subtrahieren und Wichten mit dem Bild der Wikipedia-Basiskategorie gemäß (8) unterdrückt. Die Ergebnisse für den Volltext des gleichen Artikels sind ähnlich (Tabelle 2). Das PNG-Bild für den Volltext ist in Bild 1 dargestellt.

Die jeweils fünf besten Wikipedia-Kategorien für Abstract und Volltext in Tabelle 3 zei-

	Originalbild	Verarbeitetes Bild mit (8)
Abstract	104 nicht schwarze Pixel mittlerer Grauwert 88 1. x=121 y=127 Gray=255 the 2. x=489 y= 13 Gray=239 and 3. x=165 y= 97 Gray=239 of 4. x=185 y=140 Gray=207 zebrafish 5. x=476 y=146 Gray=194 as	83 nicht schwarze Pixel mittlerer Grauwert 50 1. x=185 y=140 Gray=207 zebrafish 2. x=136 y=271 Gray=161 embryos 3. x= 80 y= 74 Gray=110 chemicals 4. x=128 y=217 Gray=110 gene 5. x=120 y=347 Gray=110 transgenic
Volltext	2238 nicht schwarze Pixel mittlerer Grauwert 46 1. x=121 y=127 Gray=255 the 2. x=165 y= 97 Gray=251 of 3. x=249 y=357 Gray=232 in 4. x=489 y= 13 Gray=225 and 5. x=185 y=140 Gray=220 zebrafish	2121 nicht schwarze Pixel mittlerer Grauwert 34 1. x=185 y=140 Gray=220 zebrafish 2. x=136 y=271 Gray=176 embryos 3. x=128 y=217 Gray=159 gene 4. x=415 y=325 Gray=157 genes 5. x=305 y= 69 Gray=151 hpf . . . 7. x= 80 y= 74 Gray=137 chemicals 15. x=120 y=347 Gray=122 transgenic

Tabelle 2: Pixel-Statistik für den Abstract und Volltext von [26]: Anzahl nicht schwarzer Pixel im Dokument, mittlerer Grauwert der nicht schwarzen Pixel, x-, y-Koordinaten und Grauwert der fünf hellsten Pixel sowie Liste der korrespondierenden Wörter aus der Rückübersetzung. Alle Statistiken sind jeweils für das originale PNG-Bild aus Abschnitt 2.1 und das verarbeitete Bild nach Subtraktion des Bildes der Basiskategorie mit nachfolgender Wichtung gemäß (8) angegeben. Die Abkürzung ”hpf” bedeutet Stunden nach der Befruchtung (engl. hours post fertilization).

gen eine klare Präferenz zu biologischen Kategorien und eine große Ähnlichkeit der Kategorisierung. Die Differenzen zwischen Abstract und Volltext resultieren aus den Nuancen der Worthäufigkeit. Einerseits sind Wörter für Modellorganismen wie ”zebrafish”, ”rodent”, ”animal”, ”model” und ”models” im Abstract überproportional vertreten, wodurch die entsprechende Kategorie *model organisms* besonders stark ist. Andererseits ist charakteristisches biologisches Vokabular im Volltext häufiger.

Unter Nutzung der hierarchischen Kategorien wird der Abstract den Kategorien *science* (Ebene 1), *scientific method* (Ebene 2), *pharmaceutical sciences* (Ebene 3), *biotechnology* (Ebene 4) und *model organisms* (Ebene 5) zugeordnet. Die entsprechenden Einteilungen für den Volltext lauten *health* (Ebene 1), *scientific method* (Ebene 2), *pharmaceutical sciences* (Ebene 3), *biotechnology* (Ebene 4) und *biochemistry methods* (Ebene 5). Diese Einteilung zeigt, dass auf den unteren Ebenen nicht nur Subkategorien der besten oberen Ebene ausgewählt werden. Beispielsweise ist *scientific method* keine Subkategorie von *health*. Der vorgeschlagene Algorithmus stellt sicher, dass solche Subkategorien auch gefunden werden, indem er mittels (15) die Subkategorien aller aussichtsreichen Kategorien untersucht.

3.2 Datensatz mit Abstracts von *gopubmed.org*

Im nächsten Experiment wurde ein Datensatz mit 1000 Abstracts von *gopubmed.org* zusammengestellt. Dieser Datensatz enthält die aktuellsten 100 Abstracts für die folgenden zehn Schlagwörter, die als nutzerspezifische Kategorien verwendet werden: *antimicrobial peptides*, *biofilm formation*, *bioinformatics*, *cell signalling*, *confocal microscopy*, *data*

Kategorie	Abstract		Volltext	
	Rang	Ähnlichkeit S_{kc}	Rang	Ähnlichkeit S_{kc}
model organisms	1	4.38	12	1.41
biotechnology	2	2.96	4	2.11
scientific method	3	2.56	3	2.21
science	4	2.39	19	1.28
pharmaceutical sciences	5	2.38	1	2.43
genetics	10	2.07	5	1.91
biology	31	0.85	2	2.38

Tabelle 3: Ähnlichkeiten und Rangordnung der besten Wikipedia-Kategorien für den Abstract und Volltext von [26]

mining, *high throughput microscopy*, *neuroprostheses*, *neurotoxins* und *zebrafish toxicity*. Der Datensatz wurde in einen Lern- und einen Testdatensatz mit jeweils 50 Dokumenten pro Kategorie geteilt. 11 Artikel werden mehrfach für verschiedene Kategorien gefunden.

In einem ersten Versuch wurden aus den Abstracts des Trainingsdatensatzes repräsentative Bilder für die zehn nutzerspezifischen Kategorien gemäß Abschnitt 2.1 und 2.2 gelernt. Für die Abstracts des Testdatensatzes wurde das jeweils ähnlichste Kategorien-Bild bestimmt. Das Ergebnis war eine Klassifikationsgüte von 75.6 %, wobei es Zuordnungsprobleme zwischen den Kategorien *data mining* vs. *bioinformatics* sowie *confocal microscopy* vs. viele andere Kategorien wie *biofilm formation*, *cell signalling*, und *high throughput microscopy* gab. Die Ursache sind semantische Ähnlichkeiten zwischen ähnlichen Methoden sowie Methoden zur Lösung bestimmter Probleme wie konfokale Mikroskopie als Methode zur Analyse von Zellsignalen und der Entstehung von Biofilmen.

In einem zweiten Versuch wurden die Ähnlichkeiten zu ausgewählten Wikipedia-Kategorien in zweidimensionalen Scatterplots visualisiert (Bild 2). Das linke Teilbild zeigt die Ähnlichkeiten zu den Wikipedia-Kategorien *bacteriology* und *optics*, das rechte zu den Kategorien *model organisms* und *bioinformatics*. Auffällig sind insbesondere die hohen Werte für die Wikipedia-Kategorie *microscopy* für viele Dokumente der nutzerspezifischen Kategorie *confocal microscopy* sowie die hohen Werte für die Wikipedia-Kategorie *bioinformatics* für die nutzerspezifischen Kategorien *bioinformatics* und *data mining*. In beide Bilder wurden beispielhaft die Positionen des in Abschnitt 3.1 umfassend diskutierten Abstracts und Volltextes von [26] sowie des Abstracts von [27] eingezeichnet.

Interdisziplinäre Arbeiten können durch hohe Ähnlichkeiten zu mehreren semantisch verschiedenen Kategorien gefunden werden. Beispielsweise untersucht [27] Interaktionen zwischen sieben verschiedenen Spezies in Biofilmen mit konfokalen Mikroskopen. Das Dokument ist in zwei nutzerspezifischen Kategorien (*biofilm formation* und *confocal microscopy*) enthalten. Der Abstract zeigt relevante Ähnlichkeiten zu den Kategorien *bacteriology* und *optics* (als geeignete Oberkategorie für Mikroskopie) (siehe Bild 2). Außerdem kann aus den Ähnlichkeiten S_{kc} quantitativ abgeschätzt werden, dass hier beide Themen etwa in gleichem Umfang behandelt werden. Viele andere Dokumente im Datensatz haben hingegen nur Ähnlichkeit zu *optics* oder *bacteriology*. Für eine Suchanfrage nach allen bakteriologischen Untersuchungen mit mikroskopischen Methoden können in Frage kommende Dokumente mit der Maus bereichsweise selektiert und später weiter ausgewertet werden. Solche grafischen Navigationsmethoden eröffnen somit einen intuitiveren Zugang zu semantischen Suchanfragen.

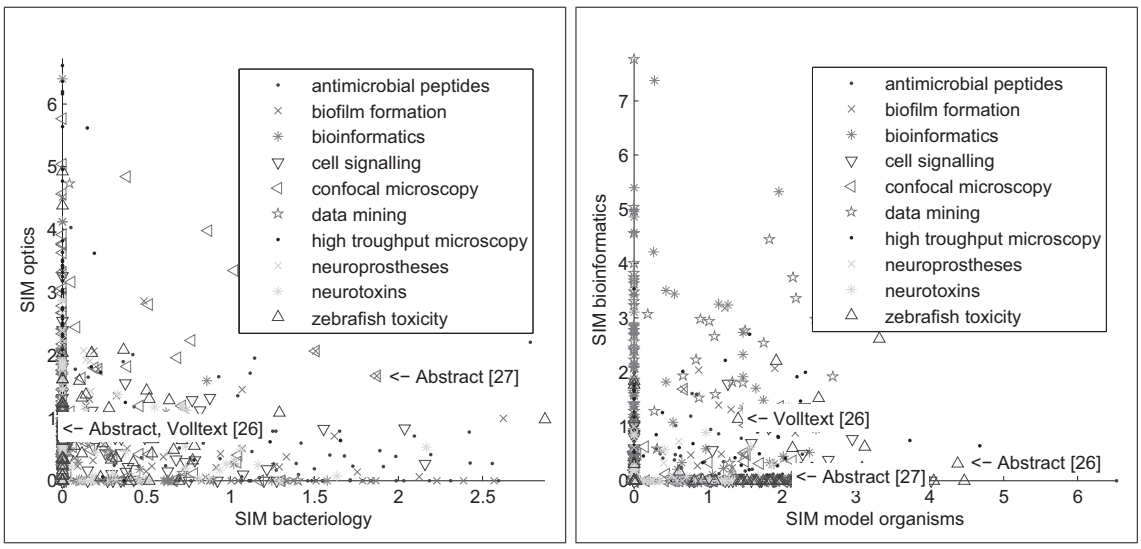

Bild 2: Zweidimensionale Scatterplots der Ähnlichkeiten S_{kc} (SIM) zu verschiedenen Wikipedia-Kategorien (jeweils dargestellt für alle Dokumente des Lern- und Testdatensatz sowie zusätzlich für den Volltext von [26]). Die zehn nutzerspezifischen Kategorien sind farb- und symbolcodiert.

4 Diskussion und Zusammenfassung

Die vorgestellte Methode umfasst einen ersten Schritt in Richtung einer verbesserten Repräsentation und Visualisierung von Dokumenten zur inhaltsbasierten Suche nach textbasierten Dokumenten in einer großen Datenbasis (engl. Information Retrieval). Die Hauptidee besteht in der *einheitlichen* Repräsentation von Dokumenten in einem zweidimensionalen Hash-Index, der Häufigkeiten von Wörtern in einem Dokument mit einer stark reduzierten 8-Bit-Auflösung als Bild codiert. Somit werden alle enthaltenen Wörter unabhängig von der Sprache und der Bekanntheit des Worts auf genau ein Hash-Element abgebildet. Die Hash-Darstellung nimmt einen bestimmten Anteil von Mehrfachzuordnungen von Wörtern zu Pixeln sowie eine reduzierte Genauigkeit bei der Repräsentation der Worthäufigkeiten in Kauf. Diese Darstellung hat den großen Vorteil, kompatibel zum gängigen PNG-Format für Rastergrafiken mit verlustfreier Bildkompression zu sein. Alle Textdokumente und daraus berechnete zusammenfassende Formate für Kategorien werden somit speichersparend als Grauwert-Bild abgelegt, was eine einfache Weiterverarbeitung ermöglicht (u.a. auch auf spezialisierten Hardwareumgebungen wie Grafikkarten).

Solche Bilder sind für Volltexte von wissenschaftlichen Verlagen mit restringiertem Zugang geeignet, weil der Volltext des Dokuments in die Ähnlichkeitsbetrachtung einbezogen werden kann, aber das PNG-Bild keine vollständige Rekonstruktion des Inhalts ermöglicht. Ein weiterer Vorteil bietet sich beim Einsatz für Suchmaschinen. Eine Suchmaschine kann einem Client auf der Basis einer Anfrage eine mittelgroße Menge an passenden PNG-Bildern zurücksenden. Die weitere Verarbeitung (z.B. eine nutzerspezifische Visualisierung oder eine Ähnlichkeitsberechnung zu einem Dokument) erfolgt erst beim Client.

Ein weiterer Beitrag umfasst das Grundkonzept, graduelle Zugehörigkeiten von Dokumenten zu Kategorien berechnen und grafisch darzustellen. Die Zuordnung von Dokumenten zu Kategorien erfolgt wahlweise standardisiert durch externe Quellen (z.B. Wikipedia) oder nutzerspezifisch aus kategorisierten Sammlungen von Dokumenten. Aus den

so ermittelten Kategorien werden ebenfalls Grauwert-Bilder im PNG-Format berechnet. Damit reduziert sich die Berechnung der graduellen Zugehörigkeiten eines neuen Dokuments auf einen Bildvergleich zwischen einem Dokument und verschiedenen Kategorien. Die Angabe gradueller Zugehörigkeiten erleichtert insbesondere das Auffinden von Dokumenten mit interdisziplinären Arbeiten, bei denen mehrere Teilgebiete einfließen. Somit kann eine Suche über eine logische Verknüpfung von Mindestzugehörigkeiten zu verschiedenen Kategorien erfolgen.

Die so entstandenen graduellen Zugehörigkeiten können auf verschiedene Art und Weise visualisiert werden. Dokumente können als Scatterplot der graduellen Zugehörigkeiten von ausgewählten Kategorien angezeigt werden.

Ein weiteres Potenzial liegt in einer Kategorisierung fremdsprachiger Dokumente (engl. CLIR: Cross-language information retrieval, vgl. Ansätze z.B. in [28, 29]). Mit einer 1:1-Übersetzung von Pixeln einer Sprache in korrespondierende Pixel einer anderen Sprache unter Nutzung eines Wörterbuchs gelingt sicherlich keine hochwertige Übersetzung, aber möglicherweise eine grobe Abschätzung des Inhalts. Hierzu muss zunächst die Sprache eines Dokuments ermittelt werden und danach eine 1:1-Übersetzungstabelle von Pixeln nach Pixeln angewendet werden. Die Übersetzungstabelle entsteht aus der Zuordnung der Adresse für das dominierende Wort für ein Pixel in der ursprünglichen Sprache des Dokuments (z.B. Zelle - $F_{385,194}$) in die Adresse für das wichtigste korrespondierende Wort in der Zielsprache (z.B. cell - $F_{446,52}$). Somit werden Grauwerte $F_{385,194}$ eines deutschen Dokuments in Grauwerte $F_{446,52}$ eines englischen Dokuments konvertiert. Somit lässt sich eine gerichtete Übersetzungstabelle in extrem kompakter Form in zwei PNG-Bildern mit einer Bittiefe von jeweils $\log_2(P_{max})$ codieren (ein Bild für die neue x-Adresse, ein Bild für die neue y-Adresse).

Danksagung: Die Autoren danken Björn Kindler für die Bereitstellung und Aufbereitung der Wikipedia-Kopie.

Literatur

[1] Berners-Lee, T.; Hendler, J.; Lassila, O.; et al.: The Semantic Web. *Scientific American* 284(5) (2001), S. 28–37.

[2] Völkel, M.; Krötzsch, M.; Vrandecic, D.; Haller, H.; Studer, R.: Semantic Wikipedia. In: *Proc., 15th International Conference on World Wide Web*, S. 585–594. 2006.

[3] Salton, G.; Wong, A.; Yang, C.: A Vector Space Model for Automatic Indexing. *Communications of ACM* 18(11) (1975), S. 613–620.

[4] Damashek, M.: Gauging Similarity with n-Grams: Language-Independent Categorization of Text. *Science* 267 (1995) 5199, S. 843–848.

[5] Liu, Y.; Dantzig, P.; Sachs, M.; Corey, J.; Hinnebusch, M.; Damashek, M.; Cohen, J.: Visualizing Document Classification: A Search Aid for the Digital Library. *Journal of the American Society for Information Science* 51(3) (2000) 3, S. 216–227.

[6] Wang, P.; Domeniconi, C.: Building Semantic Kernels for Text Classification using Wikipedia. In: *Proc., 4th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, S. 713–721. ACM New York, NY, USA. 2008.

[7] Salton, G.: *Automatic Text Processing: The Transformation Analysis and Retrieval of Information by Computer*. Addison-Wesley. 1989.

[8] Dhillon, I.; Modha, D.: Concept Decompositions for Large Sparse Text Data using Clustering. *Machine Learning* 42(1-2) (2001), S. 143–175.

[9] Lagus, K.; Kaski, S.; Kohonen, T.: Mining Massive Document Collections by the WEBSOM Method. *Information Sciences* 163(1-3) (2004), S. 135–156.