



den Dienst bzw. Website unterstützt. Auch sind Tag-Clouds nach Aussage der Verfasser in der Lage, durch einen zeitlichen Vergleich Trends im Tag-Gebrauch und damit im Content des Datenbestandes aufzuzeigen (5). Mit der Visualisierung von Tags über eine Zeitperiode beschäftigt sich beispielsweise Russel in seinem Tool Cloudalicious (6).

Inwieweit Tag-Clouds die Informationssuche unterstützen, untersuchten Sinclair und Cardew-Hall in einer Studie (1). Sie boten Nutzern zum einen eine Search Box und zum anderen eine Tag-Cloud zur Informationssuche. Als Ergebnis konnten sie festhalten, dass Tag-Clouds als visuelle Zusammenfassungen des Contents die von Rivadeneira genannten Funktionen erfüllen. Zudem stellten sie fest, dass das Scannen der Cloud und klicken auf die Tags den Nutzern leichter fällt als die Suchtermformulierung. Dies erleichtert nach Aussage der Verfasser auch den Zugang und das Retrieval in einer Fremdsprache (1).

Betrachtet man die Usability der Tag-Clouds in Studien genauer, sind interessante, aber auch zugleich unterschiedliche Ergebnisse zu beobachten. Rivadeneira et al. befassten sich mit verschiedenen Darstellungsmöglichkeiten der Tag-Clouds und testeten diese in Studien im Hinblick auf die Auswirkung verschiedener Parameter wie Größe, Farbe und Platzierung der Tags (4). Ihre Studien haben zum Ergebnis, dass Nutzer größer geschriebene Tags als signifikanter ansahen und ein vertikales nach der Tag-Größe gestaffeltes Listen-Layout zur Identifikation von thematischen Kategorien bevorzugten. Zu erwähnen ist zudem ein weiteres Ergebnis der Studie: werden in einer Tag-Cloud nur eine kleine Menge von Tags in einem räumlichen Layout visualisiert (Abb. 2), sollten kleiner geschriebene Tags im „upper left“ (UL) Quadranten platziert werden, um die Aufmerksamkeit der Nutzer auf sich zu ziehen. Dementsprechend sind größere Tags in den anderen drei Quadranten zu platzieren (a). Eine Platzierung von Tags in der Nähe des Größten jedoch hat keinerlei Auswirkung (b).

Eine weitere Untersuchung zur adäquateren Aufbereitung mit Fokus auf die genannten Parameter wurde von Halvey und Keane durchgeführt (7). Ihre Tag-Cloud Varianten umfassten neben den alphabetischen Tag-Clouds, horizontalen und vertikalen Listen auch die ungeordneten Pendants. Hier kamen die Nutzer am Besten mit der (traditionellen) alphabetischen Reihenfolge innerhalb der Tag-Clouds zurecht, schienen diese aber eher zu scannen als wirklich zu lesen, wobei auch hier die Darstellung der Tags in verschiedenen Größen als sehr hilfreich eingestuft wurde.

Auch Bateman et al. testeten in ihrer Studie, welche Wirkung die Veränderung verschiedener visueller Eigenschaften innerhalb der Tag-Cloud nach sich zieht (8). Abbildung 3 bildet diese Parameter ab und setzt sie insofern in Relation zueinander, als dass die Richtung der Pfeile angibt, inwieweit andere Eigenschaften von der Veränderung der Ersteren betroffen sind. Auf dieser Basis konstruierten Bateman et al. zehn Cloud-Sets, in welchen jeweils zehn verschiedene Tag-Clouds mit unterschiedlichen Eigenschaften visualisiert wurden und die Probanden nutzten, um Suchanfragen zu bearbeiten.

Bateman et al. stellten fest, dass

“Larger font size were chosen reliably and with the strongest effect among any visual features (8).”

Die Schriftgröße hat also auch in dieser Studie einen starken Einfluss auf das Retrieval. Ebenso konnte als Ergebnis festgehalten werden, dass die Veränderung der Schriftgewichtung und der Intensität eine gute Möglichkeit ist, die Aufmerksamkeit des Nutzers zu lenken (8).

Neben positiven Eigenschaften besitzen Tag-Clouds jedoch auch einige Nachteile. So erwähnen Hearst und Rosner beispielsweise, dass die Länge eines Wortes mit seiner Größe verschmolzen ist. Dadurch entsteht für Nutzer der Eindruck, dass Wörter mit mehr Zeichen populärer sind (5). Zudem impliziert die Abbildung populärer Tags innerhalb der Tag-Cloud, dass die

aufgezeigten Tags eine hohe semantische Dichte aufweisen, dabei besitzen diese auf Grund ihrer häufigen Verwendung einen eher schlechten Diskriminierungsgrad (9). So stellten Begelman et al. fest, dass auf diese Weise oftmals nur wenige Tags mit ihren Co-Tags eine ganze Tag-Cloud dominieren (10). Hieraus lässt sich ein weiterer Nachteil ableiten. Da Tag-Clouds zumeist den gesamten Datenbestand visualisieren, ist es sehr schwer für jüngere Tags und damit neue Bereiche in der Tag-Cloud überhaupt abgebildet zu werden. Trends nach Hearst und Rosner aufzuzeigen gelingt damit erst mit zeitlicher Verzögerung und nur über einen Vergleich mit beispielsweise dem Visualisierungstool Cloudalicious (6).

Als ein weiterer Nachteil kann festgehalten werden, dass Tag-Clouds fast keine Interaktivität hinsichtlich ihrer Darstellung erlauben. So können beispielsweise weder auf der Anbieter-Seite spezielle Aspekte personenbezogen hervorgehoben bzw. beworben werden noch besteht eine Möglichkeit auf der Nutzer-Seite, die Tag-Cloud individuell anzupassen.

Wohl das größte Problem innerhalb der Folksonomies und damit auch der Tag-Clouds ist jedoch das Fehlen jeglicher Semantik zwischen Tags (5, 9). Theoretische Ansätze für die strukturelle Erweiterung von Folksonomies werden beispielsweise unter den Stichworten „semantic enrichment“ (11) und „emergent semantics“ (12) diskutiert. Ein Versuch der wachsenden Semantik

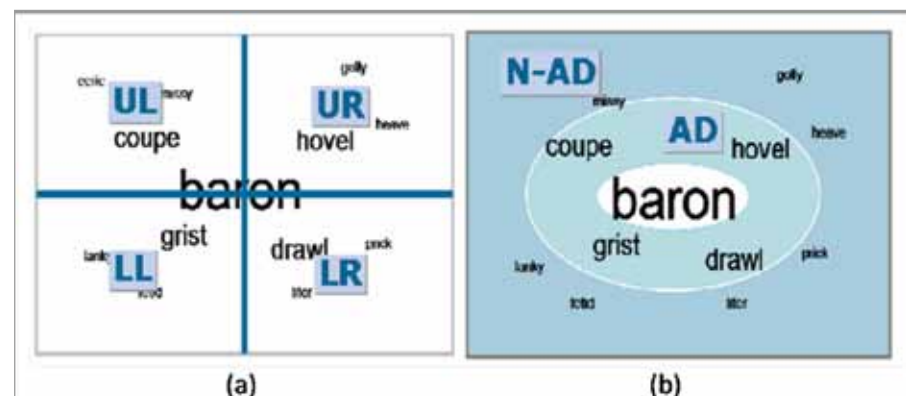


Abbildung 2: (a) In Quadranten eingeteilte räumliche Tag-Cloud (UL = upper-left, LL = lower-left, UR = upper-right, LR = lower-right); (b) Radiale Platzierung der Tags nach Größe in Abhängigkeit vom größten Tag (AD = adjacent, N-AD = non-adjacent) (4)

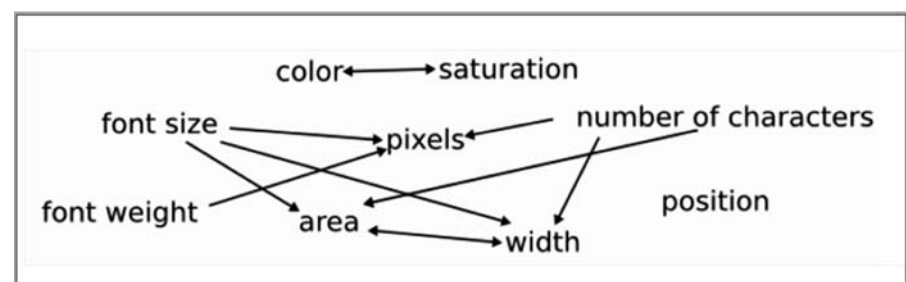


Abbildung 3: In der Studie betrachtete Merkmale und ihre Relation untereinander (8)

innerhalb von Folksonomies entgegenzu- kommen, ist die Berücksichtigung der Tag- Co-Occurrences bei einer Visualisierung.

Tag-Clouds weisen zwar einige Vorteile auf, aber es sollte Ziel sein, Nachteile zu beheben und vor allem den semantischen Anreicherungen durch die Beschreibung neuer Wege zu entsprechen.

Aus diesem Grund wird in dieser Studie ein alternativer Ansatz zur Verbesserung bzw. Ergänzung der Tag-Clouds vorgeschlagen und realisiert. Konkret thematisiert dieser Ansatz den Aufbau syntagmatischer Netze auf Basis von Tag-Co-Occurrences und mit Hilfe von verschiedenen Cluster-Verfahren.

### 3 Alternative Layouts durch semantische Anreicherung von Tag-Clouds

Um den Defiziten von Tag-Clouds entgegen zu wirken, gibt es Ansätze, welche neue Wege hinsichtlich der Repräsentation des Vokabulars einer Folksonomy einschlagen und damit auf verschiedene Weisen versuchen, semantische Bezüge abzubilden. Da diese Ansätze eine wissenschaftliche Grundlage für die vorliegende Studie bilden, sollen einige an dieser Stelle vorgestellt werden.

Neben den aufgezeigten Studien zur Usability der Tag-Clouds in Kapitel 2, beschäftigten sich auch Kaser und Lemire mit der Gebrauchstauglichkeit von Tag-Clouds (13). In ihren Untersuchungen betrachteten sie verschiedene Algorithmen zur Optimierung der Usability und orientierten sich daran, dass semantisch ähnliche Tags in Bezug gesetzt werden sollten. Ähnlichkeit bedeutet an dieser Stelle jedoch nicht, dass zwei Tags das gleiche semantische Konzept abbilden, sondern vielmehr dadurch in einer semantischen Relation zueinander stehen, als dass sie innerhalb einer Ressource zu inhaltlichen Beschreibung verwendet wurden. Damit gingen Kaser und Lemire hinsichtlich Repräsentation des Vokabulars innerhalb von Folksonomies in eine andere Richtung. Sie realisierten ihre Variante der Tag-Cloud durch Verwendung des klassischen Knuth-Plass Algorithmus für die Textausrichtung (14) und einer Platzierungsanwendung nach Skiena (15).

Doch bereits 2005 wurden erste Ansätze unternommen, die Ähnlichkeiten von Tags, Usern und Links zur Visualisierung zu nutzen und Tag-Clouds als Graph auf einer „Map“ (Abb. 4) abzubilden (16).

Begelman, Keller und Smadja verfolgten den Ansatz des automatischen Clusterings auf Basis von Tag-Co-Occurrences, um ein effektiveres Retrieval zu ermöglichen (10). Nach der Berechnung der Ähnlichkeiten von Tags, folgte ein anschließendes Spec-

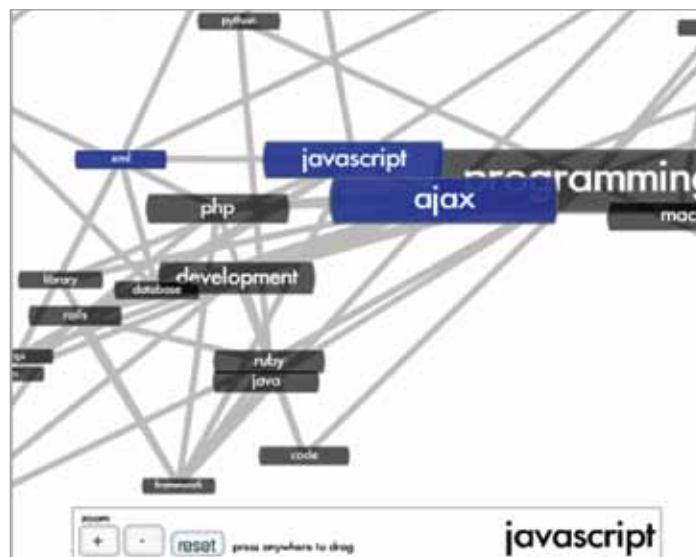


Abbildung 4: Abbildung einer Tag-Cloud als ungerichteter Graph auf Basis von Ähnlichkeitsalgorithmen (16)

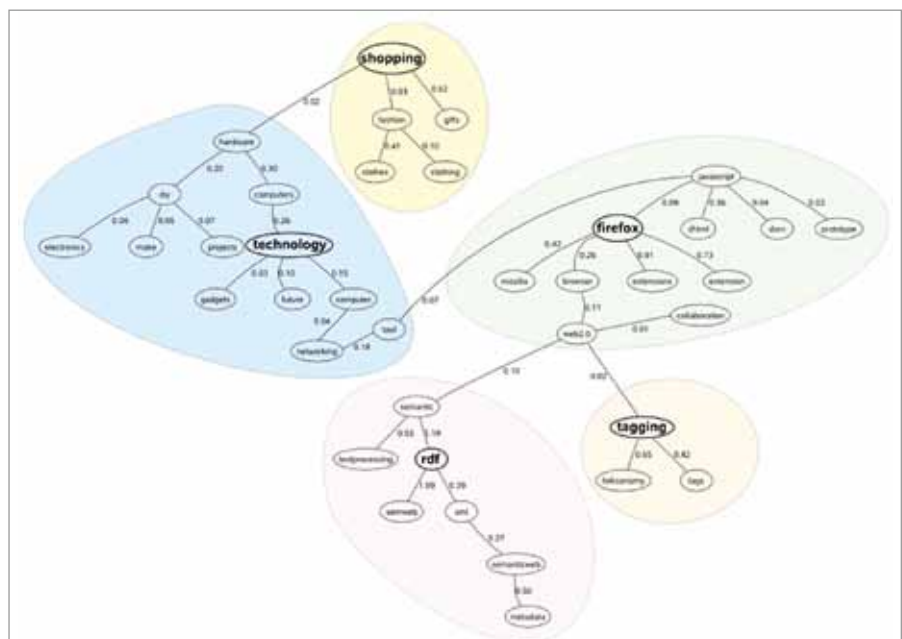


Abbildung 5: Ergebnis des Clusters (10)



Abbildung 6: Verbesserte Tag-Cloud auf Basis von Ähnlichkeitsberechnungen und Clustering (9)

tral Clustering der Tags. Begelman et al. gelang es, verschiedene thematische Cluster zu generieren, in denen an den Kanten die errechneten Ähnlichkeitswerte abgetragen wurden und die Tags die Knotenpunkte bildeten. Abbildung 5 zeigt das Ergebnis des Verfahrens.

Auch Hassan-Montero und Herrero-Solana führten Ähnlichkeitsberechnungen auf Basis der relativen Co-Occurrences durch, um Tags miteinander in Beziehung zu setzen (9). Nach der Berechnung der Tag-Ähnlichkeit nach Jaccard, clusterten sie die Tags hierarchisch nach dem k-means-Algorithmus (17).



Abbildung 6 zeigt eine nach dieser Vorgehensweise generierte Tag-Cloud, in welcher verwandte Tags in Bezug zueinander gesetzt wurden und damit den Nutzern das Retrieval nach verwandten Konzepten erleichtert werden soll.

Die verbesserte Tag-Cloud teilt mit der ursprünglichen Tag-Cloud 65 Tags. Viele, nach Aussage der Verfasser nicht signifikante Tags wie „toread“ oder „cool“ sind verschwunden und häufige Tags mit größerer Bedeutung und einem höheren Diskriminierungsgrad wie „philosophy“, „food“ oder „cinema“ sind hervorgetreten.

Eine ähnliche Vorgehensweise nutzten auch Cattuto et al., um semantische Netze auf Basis von Tag-Co-Occurrences aufzubauen, jedoch mit dem Ziel, die Netzwerkstruktur von Folksonomies zu vergleichen (18). Diesen Ansatz führten sie 2009 weiter und analysierten Ähnlichkeiten zwischen Tags und Ressourcen (19).

Inwieweit ein effektives Browsing innerhalb einer großen Menge von Annotationen mit Hilfe von Clustering möglich ist, zeigten Li et al. (20). Sie ermöglichten zudem eine hierarchische und semantische Suche innerhalb der von ihrem System ELSABer generierten Tag-Cloud (Abb. 7). In dieser haben Nutzer über einen für sie relevanten Suchterm Zugang zu semantisch ähnlichen bzw. verwandten Tags und den entsprechenden URLs.

Dass eine auf Basis von Tag-Co-Occurrences und Cluster Algorithmen generierte Tag-Cloud durchaus auch differenzierte Sichtweisen (personal, social, universal viewpoint) berücksichtigen kann, zeigten Huang et al. (21). Die Tag-Cloud passt sich hier je nach Wunsch beispielsweise den persönlichen oder den universellen Vorlieben an und erlaubt damit unterschiedliche Blickwinkel.

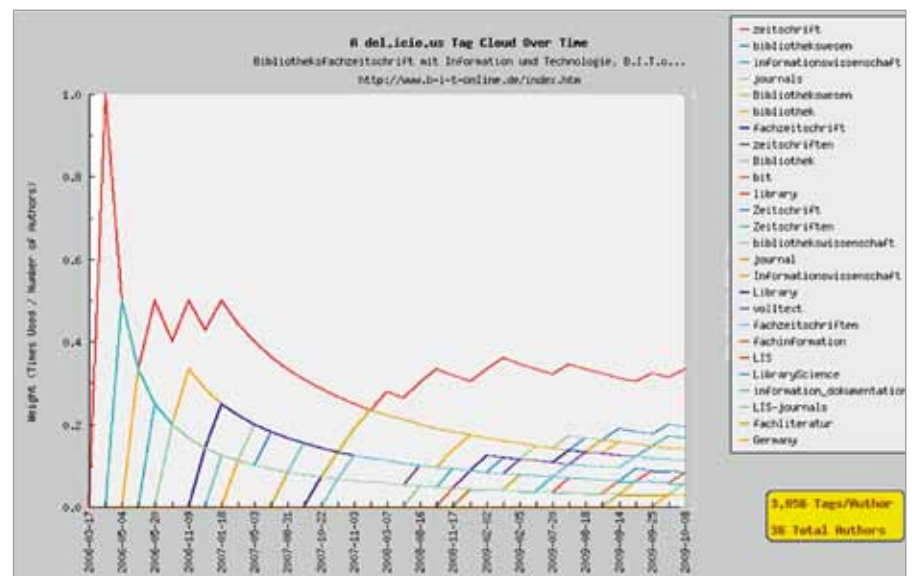
Eine Evaluation der Effekte von semantischer versus alphabetischer und zufälliger Anordnungen von Tags in Tag-Clouds führten Schrammel et al. durch (22). Sie konnten feststellen, dass bei einer spezifischen Suche die semantisch geclusterte Tag-Cloud mit zufälliger Anordnung der Tags eine Verbesserung bietet und zudem die Aufmerksamkeit der Nutzer auch auf kleiner geschriebene Tags lenkt.

Diese und andere Ansätze (12; 23) betrachten die Thematik der wachsenden semantischen Anreicherungen innerhalb von Folksonomies und versuchen, mögliche Lösungen anzubieten.

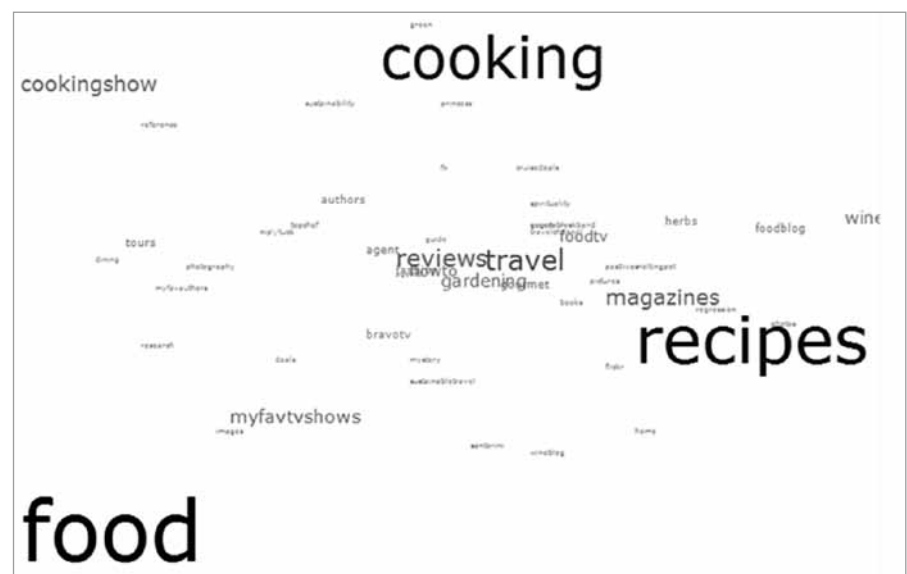
Zudem gibt es einige Demos zur Visualisierung von Tags (Abb. 8-12), welche im Web zugänglich sind. Exemplarisch seien an dieser Stelle fünf genannt.



**Abbildung 7: Interface mit Social Browsing System (20)**



**Abbildung 8: Screenshot Cloudalicious**



**Abbildung 9: Screenshot Extispicious**

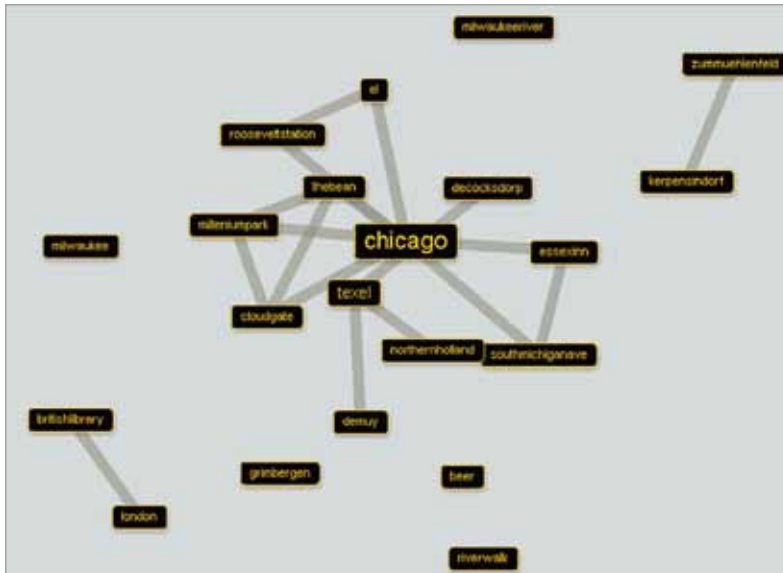


Abbildung 10: Screenshot Netr.it

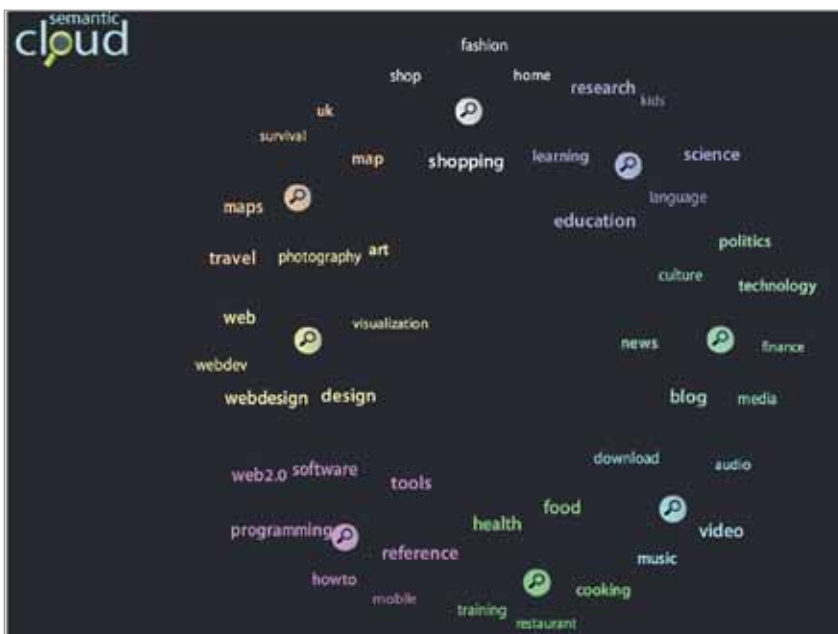


Abbildung 11: Screenshot Semantic Cloud

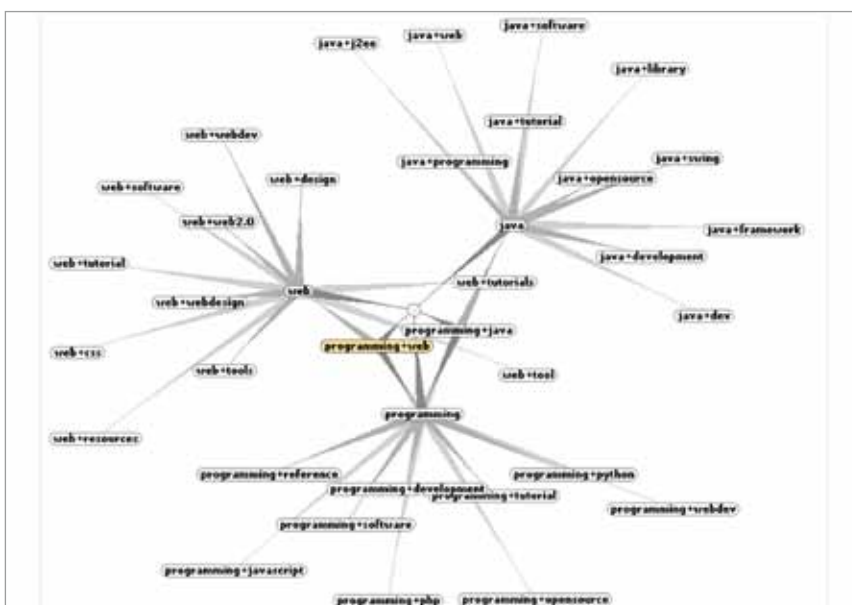


Abbildung 12: Screenshot HubLog bei Eingabe der Tags „Programming“, „Java“ und „web“

- Cloudalicious<sup>1</sup> (6) generiert Graphen, in welchen die vergebenen Tags einer URL auf Delicious hinsichtlich ihrer quantitativen Nutzung über einen bestimmten Zeitraum angezeigt werden (Abb. 8).
- Extispicious<sup>2</sup> gibt eine zufällige textliche Streuung der persönlichen Nutzertags wieder (Abb. 9).
- Netr.it<sup>3</sup> visualisiert ein Co-Occurrences-Network auf Basis der persönlichen Flickr Tags, welches sich zudem manuell verändern bzw. erweitern lässt (Abb. 10).
- Semantic Cloud<sup>4</sup> generiert eine semantische Tag-Cloud auf Basis von Ähnlichkeiten zwischen Tags und anschließendes Clustering und ermöglicht zudem eine hierarchisches Retrieval (Abb. 11).
- HubLog<sup>5</sup> ermöglicht ein visuelles Browsing zwischen verwandten Delicious Tags (Abb. 12).

## 4 Realisierung von Tag-Clustern

Im Folgenden wird die Vorgehensweise zur Realisierung von Tag-Clustern aufgezeigt. Hierfür wird zunächst auf die grundlegende Struktur und deren Implementierung eingegangen (4.1). Konkret wird in einem ersten Schritt das Data Set vorgestellt, welches die Grundlage für die weitere Vorgehensweise bildet. Auf Basis der extrahierten Daten können anschließend die für das Clustering benötigten Ähnlichkeiten berechnet werden. Letzte Schritte bilden die Visualisierung, die Implementierung der Ranking-Algorithmen und die Einbindung in ein User Interface. Die Betrachtung der Ergebnisse der Vorgehensweise und Realisierung von Tag-Clustern mit Hilfe eines Applets bildet den zweiten Teil des Kapitels (4.2).

### 4.1 Materialien und Methode

An dieser Stelle soll das Konzept der Implementierung aufgezeigt werden. Hierzu wird auf das Model-View-Controller-Modell (MVC-Pattern-Modell) zurückgegriffen, um eventuelle spätere Änderungen oder Erweiterungen zu erleichtern. Die Umsetzung erfolgt in der Programmiersprache Java. Wie in dem Klassendiagramm (Abb. 13) dargestellt, wurde die Applikation in vier Pakete unterteilt. Das Paket *cluster* ist dem Model, das Paket *parser* dem Control-Teil und das Paket *graph* dem View-Teil zuzuordnen. Die Klassen, welche sich direkt im Eltern-Paket *javatagclustering* befinden, zählen sowohl zum View- als auch zum Control-Teil. Das *cluster*-Paket besteht aus den Klassen

<sup>1</sup> <http://cloudalicio.us/>

<sup>2</sup> <http://kevan.org/extispicious>

<sup>3</sup> <http://www.netr.it/>

<sup>4</sup> <http://semanticcloud.rieskamp.info/>

<sup>5</sup> <http://hublog.hubmed.org/archives/001049.html>

*TagSimilarity*, *Cluster*, *TagEntry* und *URLEntry*. Die Klasse *Cluster* ist dabei diejenige, über die auf das Modell zugegriffen werden kann. Mit ihr sind über eine Aggregations-Verbindung die Klassen *TagEntry* und (indirekt über *TagEntry*) *URLEntry* verbunden.

Das Paket *parser* besteht nur aus einer einzelnen Klasse *Parser*, welche über Methoden den Zugriff auf Delicious-Ergebnisse ermöglicht. Die Klasse ist über eine 1:1-Assoziation mit der Klasse *Cluster* verbunden. Das *graph*-Paket ist für die Darstellung verantwortlich und wird über die *ClusterGraph*-Klasse verwendet. Diese Klasse ist über Assoziationsrelationen mit *Cluster* und *TagCluster* verbunden, wobei letztere Klasse zur Darstellung des Graphen nötig ist. Das Paket *javatagclustering* ist das Eltern-Paket und bietet mit der Klasse *MainFrame* eine grafische Benutzeroberfläche zum Zugriff auf das Modell an. Die Klasse ist über eine Assoziationsrelation mit der Klasse *ClusterGraph* verbunden.

#### 4.1.1 Data Set

Damit Tag-Cluster realisiert werden können, bildet eine Menge von Bookmarks und die zugehörigen Tags eine Voraussetzung. Um die Themenunabhängigkeit zu demonstrieren, wurden exemplarisch zwei Mengen zu verschiedenen Themen aus dem zentralen Bookmark-Dienst Delicious extrahiert. Die Themen umfassen zum Einen die Bookmarks rund um den Tag *stemmer* und zum anderen Bookmarks im Bereich *recipes*, *cooking* und *seafood*.

Da die von Delicious zur Verfügung gestellte API nur den Zugriff auf die von einem spezifischen Benutzer vergebenen Tags bzw. getaggtten Bookmarks erlaubt, musste ein auf den Kontext angepasster Parser geschrieben werden. Die Hauptaufgabe des Parsers besteht darin, bei einer vorgegebenen Suchanfrage, diese bei der Delicious-Tagsuche einzugeben und die Ergebnisse auszuwerten. Diese Ergebnisse setzen sich aus der URL des Bookmarks, dem Nutzernamen und der Anzahl, wie oft diese URL gebookmarkt wurde, zusammen.

Die URLs wurden extrahiert und gesammelt, wobei eine URL nur dann in den Datenpool aufgenommen wurde, wenn sie nicht bereits enthalten ist, um Dubletten zu vermeiden.

Mit Hilfe einer Bookmark-History ist es möglich, an die zu einem Bookmark zugehörigen Tags zu gelangen. Die History zeigt die Bookmarks inklusive der vergebenen Tags der letzten 2000 Nutzer. Damit es zu keinem Datenverlust kommt, wurde bei der Auswahl der Themen darauf geachtet, dass kein Bookmark von mehr als 2000 Nutzern getaggt wurde. Der Parser ist in der Lage diese History für alle gespeicherten URLs auf-

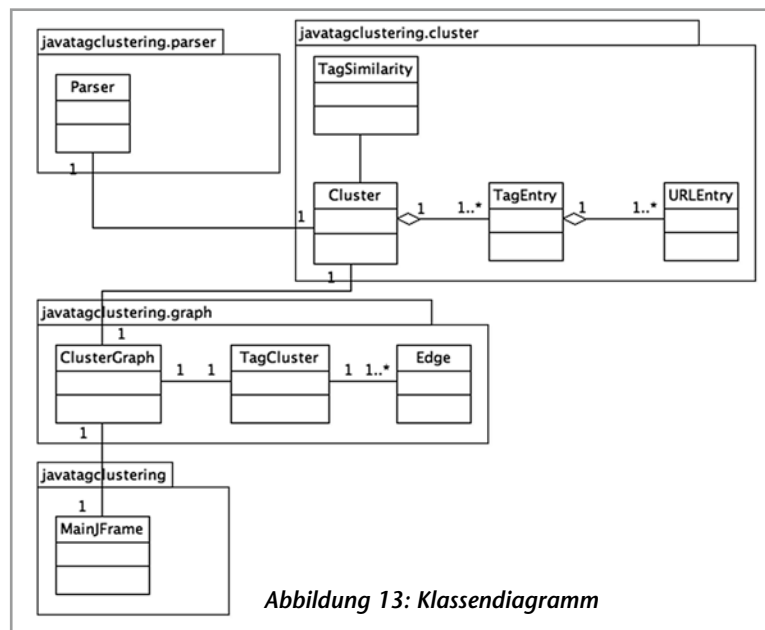


Abbildung 13: Klassendiagramm

```
stemmer 325 http://snowball.tartarus.org/ 54 1396 251 http://nlp.ipipan.waw
http://www.perl.com/pub/a/2004/02/19/plucene.html 1 150 44 http://www.mpi-i
stemming 112 http://snowball.tartarus.org/ 191 1396 251 http://www.dirolf.c
nlp 88 http://snowball.tartarus.org/ 73 1396 251 http://cst.dk/online/lemma
search 80 http://snowball.tartarus.org/ 71 1396 251 http://www.compuglobalh
programming 77 http://snowball.tartarus.org/ 74 1396 251 http://tartarus.or
language 75 http://snowball.tartarus.org/ 48 1396 251 http://nlp.ipipan.waw
algorithm 67 http://snowball.tartarus.org/ 32 1396 251 http://tartarus.org/
porter 59 http://snowball.tartarus.org/ 11 1396 251 http://www.dirolf.com/p
linguistics 50 http://snowball.tartarus.org/ 74 1396 251 http://tartarus.or
bildverarbeitung 49 http://www.idw-online.de/pages/de/news299153 1 7 6 http
imaging 49 http://www.idw-online.de/pages/de/news299153 2 7 6 http://vision
algorithms 48 http://snowball.tartarus.org/ 76 1396 251 http://tartarus.org
stem 47 http://snowball.tartarus.org/ 5 1396 251 http://tartarus.org/~marti
tools 44 http://snowball.tartarus.org/ 31 1396 251 http://www.compuglobalhi
text 43 http://snowball.tartarus.org/ 33 1396 251 http://tartarus.org/~mart
php 43 http://snowball.tartarus.org/ 2 1396 251 http://www.compuglobalhipe
```

Abbildung 14: Liste der Tags nach Häufigkeit und der zugehörigen URLs

zurufen und die vergebenen Tags zu extrahieren, ihre Häufigkeiten zu zählen und die URL zu speichern, in welcher sie auftreten. Im Folgenden wurden alle gefundenen Tags, die sich nur in der Groß- und Kleinschreibung unterscheiden, zusammenzufasst. Weitere Realisierungen in informationslinguistischer Hinsicht konnten in diesem Projekt noch nicht implementiert werden. Nach diesem Verfahren ergab die

erste Suchanfrage mit dem Tag *stemmer* zunächst 599 Bookmarks, wobei abzüglich der Dubletten noch 327 unterschiedliche URLs übrig blieben. Der Parser lieferte nach der Untersuchung aller URLs 2743 Tags. Die zweite Suchanfrage mit den Tags *recipes*, *cooking* und *seafood* ergab vor der Eliminierung der Dubletten 684 Bookmarks, welche dann auf 518 schrumpfte und 2575 Tags lieferte (Tab. 1).

Tabelle 1: Zusammenfassung der Extraktionsergebnisse

Tag Menge	stemmer	recipes
Anzahl Bookmarks	599	684
Abzüglich Dubletten	327	518
Anzahl unterschiedlicher Tags	2743	2575

```
stemmer 342 10001 55 1396 251 10002 2 8 5 10003 1 4 4 10004 1 5 5 10005 1
stemming 137 10001 196 1396 251 10006 1 5 4 10007 12 166 43 10008 295 2806
nlp 127 10001 80 1396 251 10004 1 5 5 10007 1 166 43 10008 57 2806 409 100
programming 109 10001 79 1396 251 10008 230 2806 409 10012 1 48 25 10014 7
search 109 10001 78 1396 251 10007 12 166 43 10008 95 2806 409 10014 78 11
language 92 10001 48 1396 251 10002 1 8 5 10008 135 2806 409 10012 2 48 25
porter 84 10001 13 1396 251 10006 1 5 4 10008 72 2806 409 10013 1 24 17 10
algorithm 81 10001 32 1396 251 10008 237 2806 409 10012 4 48 25 10014 32 1
algorithms 66 10001 77 1396 251 10008 188 2806 409 10012 3 48 25 10014 77
```

Abbildung 15: Liste der Tags mit eindeutiger URL-ID, um Speicherplatz zu sparen



Der endgültige Datensatz für jeden Tag enthält neben der Anzahl der URLs, in denen er als Tag enthalten war, eine Liste aller seiner zugehörigen URLs (vgl. Abb. 14). Diese Liste beinhaltet die URL, die Häufigkeit des Auftretens des Tags bei dieser URL, die Anzahl an insgesamt vergebenen Tags bei dieser URL und die Anzahl an verschiedenen Tags, die dieser URL hinzugefügt wurden. Für den Tag *stemmer* bedeutet dies, dass er 325mal in den URLs getaggt wurde, so z.B. bei <http://snowball.tartarus.org> 54mal. Insgesamt wurden bei diesem Bookmark 1396 Tags vergeben, worunter 251 verschiedene waren. Um Speicherplatz zu sparen, wurden die URLs in diesen Listen durch eine eindeutige Identifikationsnummer ersetzt (Abb. 15). Damit später allerdings auch die dazugehörige URL angezeigt werden kann, enthält der Parser eine Methode, die bei Übergabe einer ID die URL zurückliefert.

#### 4.1.2 Berechnung der Tag-Ähnlichkeiten

Die weitere Realisierung von Tag-Clustern geschieht auf der Basis von Tag-Co-Occurrences. An dieser Stelle werden die klassischen Ähnlichkeitsmaße Dice (1), Cosinus (2) und der Jaccard-Sneath-Koeffizient (3) eingesetzt werden, um die Koinzidenzwerte ( $\varphi$ ) zwischen zwei Tags *A* und *B* zu berechnen.

Auf den Dienst Delicious und die ausgewählten thematischen Mengen angewandt bedeutet dies, dass es sich in der Formel bei *a* um diejenigen Bookmarks handelt, welche den Tag *A* beinhalten, und analog bei *b*, dass es sich um die Menge der Bookmarks handelt, in welchen der Tag *B* vorkommt. Bei *g* handelt es sich um die Bookmarks, in denen beide Tags gemeinsam vorkommen. Ähnlichkeit bedeutet hier jedoch nicht, dass die Tags identische Konzepte repräsentieren, sondern wir gehen davon aus, dass die Tags in irgendeiner Weise in semantischer Beziehung zueinander stehen, da sie gemeinsam zur Erschließung eines Bookmarks verwendet wurden. Eine exakte Identifikation der Relation kann an dieser Stelle bisher nur intellektuell erfolgen.

Technisch stellen sich die Anforderungen in einem ersten Schritt die drei Algorithmen zur Ähnlichkeitsberechnung zu implementieren und in einem weiteren die Ähnlichkeiten der Tags nach einem ausgewählten Algorithmus zu berechnen.

Hierzu wurden zwei Klassen realisiert: *class Cluster* und *class TagCluster*. Diese sind über eine 1:1-Aggregation verbunden. Als öffentliche Schnittstelle werden eine Funk-

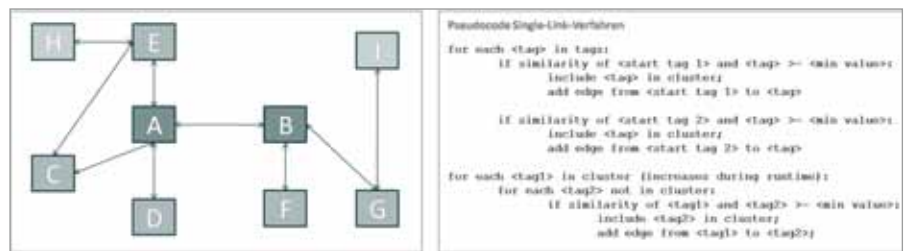


Abbildung 16: Clustering nach dem Single-Link-Verfahren und Pseudocode

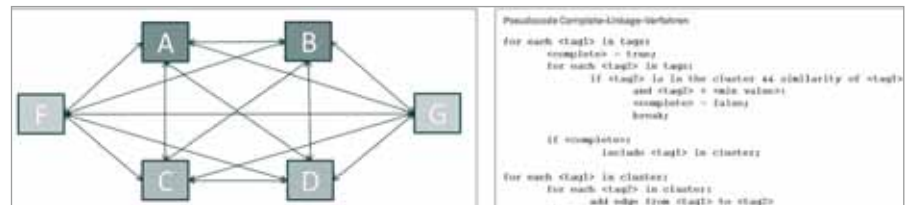


Abbildung 17: Clustering nach dem Complete-Link-Verfahren und Pseudocode

tion zur Auswahl des eingesetzten Ähnlichkeitsmaßes und ein öffentliches Feld zum Auslesen der Ähnlichkeiten zwischen den Tags zur Verfügung gestellt.

Die entwickelte Software bietet dem Nutzer die Möglichkeit an dieser Stelle auszuwählen, welches Ähnlichkeitsmaß zur Berechnung verwendet werden soll. Da die verschiedenen Berechnungsmethoden Einfluss auf die Cluster-Bildung und damit auf die Visualisierung haben, soll die Evaluation zeigen, welches Ähnlichkeitsmaß sich besser zur Visualisierung eignet bzw. von den Nutzern bevorzugt wird.

#### 4.1.3 Clustering und Visualisierung

Nach der Berechnung der Koinzidenzwerte für alle Tag-Paare in der Treffermenge kann nun eine Klassifikation der Tags nach den errechneten Ähnlichkeiten vorgenommen werden. Für diese Clusteranalyse wurden drei verschiedene Verfahren ausgewählt: das Single-Link-, das Complete-Link- und das Group-Average-Verfahren (24).

Beim Single-Link-Verfahren (Abb. 16) wird von dem Tag-Paar (A, B) ausgegangen, welches die höchste Ähnlichkeit besitzt und damit das Ausgangspaar bildet. Dieses muss jedoch eine weitere Bedingung erfüllen: es muss in mindestens 50 Bookmarks vorkom-

men, um eine laborierte Clusterbildung zu gewährleisten. Anschließend werden alle Tags hinzugefügt, die mit dem Tag A ähnlich sind (C, D und E). Hierbei kann bei der Clusterbildung ein Schwellenwert angegeben werden, der eine Ausuferung des Clusters verhindert. Ebenso wird mit dem Tag B (F, G) und den neu gewonnenen Tags verfahren (H, I).

Erfolgt die Cluster-Bildung durch das Complete-Link-Verfahren (Abb. 17), so bildet wieder das Tag-Paar (A, B) mit der höchsten Ähnlichkeit das Ausgangspaar, jedoch werden nur die Tags mit in den Cluster aufgenommen, welche mit A und B eine Ähnlichkeit über einen bestimmten Schwellenwert besitzen (C, D). Mit den neugewonnenen Tags wird ebenso verfahren (F und G).

Das Group-Average-Verfahren (Abb. 18) geht in einem ersten Schritt wie das Single-Link-Verfahren vor. Nachdem der Cluster auf die aufgezeigte Weise gebildet wurde, wird für diesen jedoch hinsichtlich der Ähnlichkeit ein arithmetischer Mittelwert berechnet. Der Mittelwert in Abbildung 19 beträgt 0,59. Im letzten Schritt werden alle Tags entfernt, die nicht eine Ähnlichkeit über diesen Schwellenwert mit ihren jeweiligen Ausgangstags besitzen. Auf diese Weise fallen die Tags H, G und I aus dem Tag-Cluster heraus.

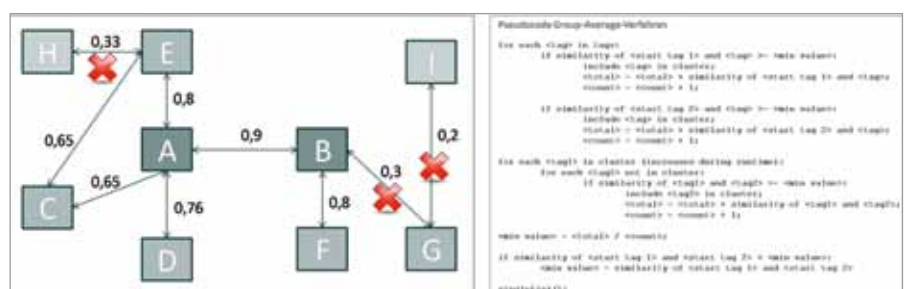


Abbildung 18: Clustering nach dem Group-Average-Verfahren (Mittel 0,59) und Pseudocode

Auf Basis der Koinzidenzwerte für alle Tag-Paare in der Bookmarkmenge lassen sich nun Tag-Cluster in Form von syntagmatischen Netzen realisieren. Um den komplexen Prozess der Visualisierung eines Graphen zu vereinfachen, wurde an dieser Stelle auf das „Java Universal Network/Graph Framework“ (JUNG) zurückgegriffen, welches unter der BSD-Lizenz zur freien Verfügung steht. Somit besteht der Cluster aus ungerichteten Graphen, an welchen der errechnete Ähnlichkeitswert durch die Stärke der Graphen zwischen den Tags visualisiert wird. Aus Gründen der Übersichtlichkeit wurden die Ähnlichkeiten in zweier Schritten zusammengefasst, so dass die Ähnlichkeiten 0.1 und 0.2, 0.3 und 0.4 etc. über eine gemeinsame Strichstärke verfügen. Da die Nutzer in den zu Beginn erwähnten Studien besonders die unterschiedliche Größe der Tags innerhalb der Folksonomy-Realisierung als positiv betrachten, wurde dieser Aspekt auch in der Bildung der Tag-Cluster berücksichtigt. Die Größe der Tags repräsentiert demnach wie bei herkömmlichen Tag-Clouds die quantitative Nutzung dieser. In diesem Fall werden die Tags mit Hilfe der Minimum-Maximum-Normalisierung durch zehn verschiedene Größen im Cluster realisiert (Abb. 19).

#### 4.1.4 Ranking-Algorithmen

Die Treffermenge wird den Nutzern (zu Evaluationszwecken) auf zwei Weisen dargeboten. In einer ersten Variante werden die Bookmarks nach der absoluten Häufigkeit der Tags gerankt. Kommt es durch Klicken auf die Linie zwischen zwei Tags zu einer UND-Verbindung, wird ebenso für beide Tags die absolute Häufigkeit berechnet und anschließend beide Werte addiert. Die Höhe der Summe bestimmt das Ranking. Die zweite Variante erfolgt nach der Berechnung von WDF \* ITF (25). Hierbei handelt es sich um die Betrachtung der *Within Document Frequency* (WDF), also der Logarithmisierung der relativen Häufigkeit, und der *Inverse Tag Frequency* (ITF), einen textstatistischen Wert, welcher auf die Gesamtzahl der Tags in der Bookmark-Menge Bezug nimmt.

$$WDF * ITF = \left[ \frac{\log \text{freq}(t,b) + 1}{\log L} \right] * \left[ \left( \frac{\log M}{m} \right) + 1 \right]$$

Bei  $\text{freq}(t,b)$  handelt es sich um das Vorkommen eines Tags ( $t$ ) in einem Bookmark ( $b$ ), d.h. wie oft wurde ( $t$ ) vergeben.  $L$  ist Gesamttagzahl des Bookmarks, also alle Tags UND deren Häufigkeit eines Bookmark. Die Gesamtzahl aller Tags in der Treffermenge (alle Tags in der gesamten Menge und deren Häufigkeit addieren) wird als  $M$  daklariert. Bei  $m$  handelt es sich um die Anzahl des Tags ( $t$ ) in der Menge.

Pseudocode Tag-Gewichtung	Pseudocode Linien-Gewichtung
<pre> &lt;relative occurrence&gt; = number of times the tag occurs - minimum number of times any tag in the cluster occurs &lt;cap&gt; = maximum number of times any tag in the cluster occurs - minimum number of times any tag in the cluster occurs  &lt;freq0&gt; = &lt;cap&gt; * 9/10 &lt;freq0&gt; = &lt;cap&gt; * 8/10 ... &lt;freq1&gt; = &lt;cap&gt; * 1/10  if &lt;relative occurrence&gt; &gt; &lt;freq0&gt;:     set font size to 22 if &lt;relative occurrence&gt; &gt; &lt;freq0&gt;:     set font size to 21 ... if &lt;relative occurrence&gt; &gt; &lt;freq1&gt;:     set font size to 14 else:     set font size to 13 </pre>	<pre> &lt;relative sim&gt; = (similarity - minimum similarity) / (1 - minimum similarity);  if &lt;relative sim&gt; &gt;= 0.9:     return 4.5 if &lt;relative sim&gt; &gt;= 0.8:     return 4.0 if &lt;relative sim&gt; &gt;= 0.7:     return 3.5 ... if &lt;relative sim&gt; &gt;= 0.1:     return 0.5 if &lt;relative sim&gt; &lt; 0.1:     return 0.25 </pre>

Abbildung 19: Pseudocode der Tag-Gewichtung und der Linien-Stärke

#### 4.1.5 User Interface

Letzter Schritt ist das Design des Applets und damit des User Interfaces. In diesem wird darauf Wertgelegt, dass die Retrieval-Schnittstelle möglichst übersichtlich und damit nutzerfreundlich ist. Im Applet werden alle implementierten Funktionen verknüpft und der Nutzer ist in der Lage, selbstständig alle Werte, welche zur Clusterbildung herangezogen werden können, auszuwählen. Abbildung 20 zeigt die Realisierung des User Interfaces.

So kann über einen Schieberegler der Schwellenwert (0 - 1.0) und über verschiedene Buttons das Ranking- sowie das Clusterverfahren und der Ähnlichkeitsalgorithmus eingestellt werden. Auch die beiden exemplarisch ausgewählten Testmengen (*Tag Sets*) können einzeln ausgewählt werden.

#### 4.2 Ergebnisse

Die neue Retrieval-Schnittstelle in Form von Tag-Clustern ist in der Lage, nicht nur eine Übersicht über den Content einer Datenbank zu geben, sondern zudem nach einer initialen Suchanfrage eine Hilfestellung innerhalb der Fülle an Treffern zu geben. Hierfür werden die Tags innerhalb der Treffermenge nach dem aufgezeigten Verfahren geclustert und visualisiert.

Anstelle oder als Ergänzung zu einer herkömmlichen Tag-Cloud (Abb. 1), hat der Nutzer nun einen Cluster zur Verfügung, welchen er interaktiv anpassen kann. Abbildung 21 zeigt einen Screenshot eines Cluster-Aufbaus. Hier ist erkennbar, dass 2575 Tags zur Cluster-Bildung herangezogen werden und dass das Tag-Paar mit der höchsten Ähnlichkeit eine Koinzidenz von 0.995 besitzt.

```

Number of Tags:2575
max:0.995107136031466
tag1:1
tag2:2
Log: getGraph - Thickness:true
got GraphEvent type:VERTEX_ADDED for 0
got GraphEvent type:VERTEX_ADDED for 1
got GraphEvent type:VERTEX_ADDED for 2
got GraphEvent type:VERTEX_ADDED for 3
got GraphEvent type:VERTEX_ADDED for 4
got GraphEvent type:VERTEX_ADDED for 5
got GraphEvent type:VERTEX_ADDED for 6
got GraphEvent type:VERTEX_ADDED for 7
got GraphEvent type:VERTEX_ADDED for 8
got GraphEvent type:EDGE_ADDED for 0
got GraphEvent type:EDGE_ADDED for 1
got GraphEvent type:EDGE_ADDED for 2
got GraphEvent type:EDGE_ADDED for 3
got GraphEvent type:EDGE_ADDED for 4

```

Abbildung 21: Screenshot des Cluster-Aufbaus

Abbildungen 22 und 23 zeigen automatisch generierte Cluster aus einer initialen Treffermenge von rund 400 Bookmarks und den 2575 Tags. Zur Bildung wurden an dieser Stelle exemplarisch das Complete-Link-Verfahren und eine Ähnlichkeitsberechnung nach Cosinus ausgewählt. Abbildung



Abbildung 20: Tag-Cluster User Interface



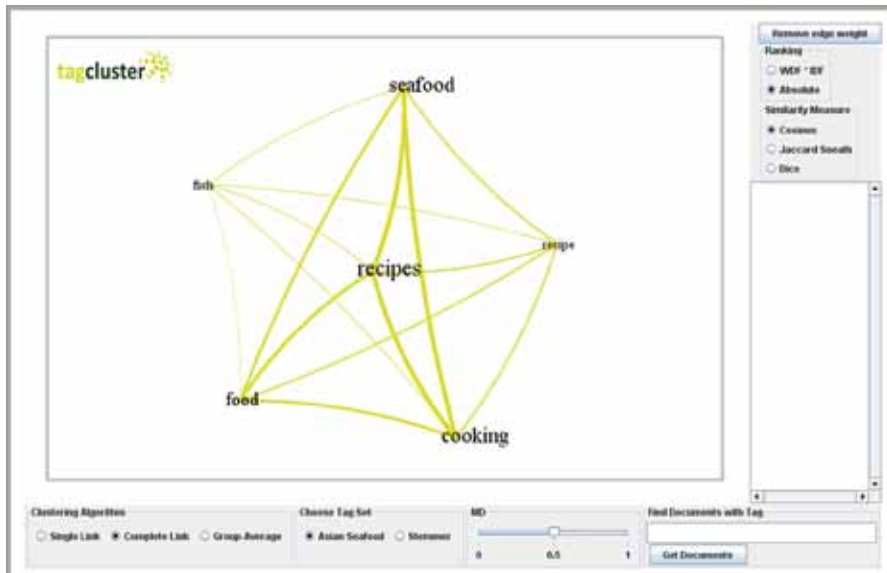


Abbildung 22: Automatisch generierter Complete-Link-Cluster mit einer Tag-Ähnlichkeit von 0.5; Ähnlichkeit berechnet nach Cosinus

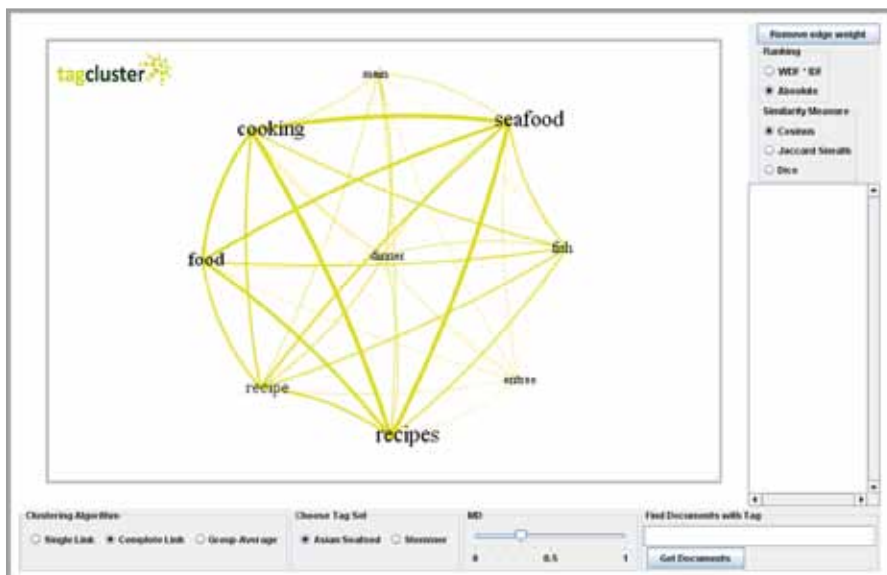


Abbildung 23: Automatisch generierter Complete-Link-Cluster mit einer Tag-Ähnlichkeit von 0.3; Ähnlichkeit berechnet nach Cosinus

22 zeigt einen Cluster mit einem eingestellten Schwellenwert von 0.5, in Abbildung 23 ist eine Cluster-Visualisierung mit einem Schwellenwert von 0.3 zu sehen. Durch die Einstellung der Vielzahl an Werten und Verfahren, ergeben sich eine große Anzahl an Cluster-Varianten. Die Senkung des Schwellenwerts von 0.5 auf 0.3 hat zur Folge, dass mehr Tags zur Clusterbildung zugelassen sind. Der Nutzer erhält hierdurch Zugang zu weiteren semantisch ähnlichen Tags, welche ihm evtl. vorher nicht bekannt oder bewusst waren und damit neue Aspekte beim Retrieval mit einbringen. Die Ähnlichkeit der Konzepte wird ihm durch die Liniestärke angezeigt. Die Größe der Tags gibt die Nutzung des Tags innerhalb der Treffermenge an. Durch Klicken auf einen Tag erhält der Nutzer die zugehörige Treffermenge. Eine UND-Verknüpfung und damit

eine Einschränkung erhält er durch Klicken auf eine Verbindungslinie zwischen zwei Tags oder durch ein Anklicken von weiteren Tags. Das Ergebnis der Trefferliste passt sich dynamisch seiner Suchanfrage an und der Nutzer ist so in der Lage, aktiv die Treffermenge zu beeinflussen und Subsets auszuwählen.

## 5 Evaluation

Dieses Kapitel thematisiert die Evaluation der Software und deren Ergebnisse. Hierzu wird zunächst auf die Methode und das Modell eingegangen, welches sich aus verschiedenen Bestandteilen, sowohl klassischen als auch neuen Elementen, zusammensetzt (5.1). In einem zweiten Teil werden die auf Grundlage des Modells

erstellten Fragebögen ausgewertet und Ergebnisse präsentiert (5.2).

### 5.1 Methode

Um ein konkretes Web2.0 Service-Tool wie das Vorliegende zu evaluieren, ist es nicht sehr nützlich nur mit klassischen Qualitäts-Indikatoren wie beispielsweise Recall und Precision zu arbeiten. Das vorliegende IT-Evaluationsmodell Modell (Abb. 24) besteht aus drei Teilen: der Qualität des Informationssystems – inklusive des Retrieval Systems – (*IT system quality*), der Qualität der Wissensbasis (*knowledge quality*) und der Qualität des Services (*IT service quality*).

Eine etablierte Methode zur Evaluation von Diensten ist die Nutzung des SERVQUAL-Ansatzes (26). Dieser ist ein standardisiertes Verfahren zur Messung der Qualität von Dienstleistungen und stützt sich auf die Betrachtung von zwei Werten: zum Einen auf den Erwartungswert (*expectations E*) und zum Anderen auf den Beobachtungswert der Nutzer (*perceptions P*). Bei einer Evaluation hat der Nutzer bei jeder Frage – sowohl auf der Erwartungs- als auch bei der Beobachtungsseite – die Möglichkeit, auf Skalen von 1 (gar nicht) - 7 (absolut) seinen Standpunkt zu fixieren. Die Differenz zwischen Erwartung und Beobachtung ( $P-E$ ) ist ein Ansatzpunkt hinsichtlich der Qualität ( $Q$ ) des Systems. Parasuraman, Zeithaml und Berry definieren zudem fünf Dimensionen der Qualität, auf welchen das Verfahren basiert: *Reliability*, *Aussurance*, *Tangibles*, *Empathy* und *Responsiveness*.

Zentral jedoch ist der Wert, welcher sich durch die Differenz von Erwartung und Beobachtung ergibt (27). Der SERVQUAL-Ansatz kann auch zur Bewertung der Qualität bzw. Effektivität von Informationssystemen adaptiert werden (28). Jedoch kommt es in diesem Fall zu Kontroversen auf Grund der ausschließlichen Betrachtung der Differenz (29) und der zuvor definierten fünf Dimensionen der Qualität (30). Um den Defiziten des Ansatzes entgegenzuwirken, wird daher in der Forschung ein anderer Ansatz diskutiert: SERVPERF. Hier liegt der Fokus der Qualitätsmessung bei der Wahrnehmungskomponente bzw. bei der Beobachtung von Seiten der Nutzer. Das Qualitätsniveau, bestimmt durch die einzelnen Nutzer-Bewertungen, kann entweder als ergänzender Wert bei einer Evaluation herangezogen werden (31) oder als der einzige ausschlaggebender Faktor einer Evaluation dienen (32).

In der vorliegenden Evaluation wurden beide Varianten berücksichtigt, so dass der SERVQUAL-Ansatz eine solide Basis bildet, die um Aspekte des SERVPERF-Ansatzes ergänzt wurde und damit eine nützliche

Methode ist, um die Qualität von Software zu messen (33; 34). Neben der Berechnung der Differenz, werden somit auch die Erwartungs- und Beobachtungswerte an sich zur Evaluation herangezogen. Auf die vordefinierten fünf Dimensionen wurde jedoch nicht zurückgegriffen, da eigene, besser passende, Dimensionen entwickelt wurden.

Zusätzlich wird die Qualität der Software auf Basis klassischer Indikatoren für das Research in IT-Systemen gemessen: der wahrgenommenen Einfachheit der Bedienung (*perceived ease of use*) und der wahrgenommenen Nützlichkeit (*perceived usefulness*) (35). Mit Vertrauen (*trust*) wird in der Literatur ein weiterer Qualitäts-Faktor diskutiert (36), welcher ebenfalls mit in die Evaluation aufgenommen wird. Als relativ neuer, aber durchaus interessanter Aspekt, wird der empfundene Spaß (*perceived playfulness*) (37) bzw. *perceived enjoyment* (38), den die Testpersonen bei der Benutzung des Tools haben, berücksichtigt. Dies ist insbesondere daher wichtig, als das gerade dieser Unterhaltungsfaktor in Web2.0 Umgebungen im Vordergrund steht und maßgeblich über den Erfolg einer Applikation entscheiden kann.

In dem vorliegenden Modell (Abb. 24) werden mit der Qualität der Zugangspunkte zu den Dokumenten und der Qualität derer (*knowledge quality*) zusätzliche Qualitätsindikatoren bzw. -dimensionen berücksichtigt. Diese können jedoch ebenso wie die klassischen Parameter Precision und Recall als Indikatoren des Information Retrieval Subsystems zur Qualitätsbewertung eines Informationssystems nicht berücksichtigt werden, da sie abhängig von dem Informationssystem (*Service*) selbst und nicht von den Applikationen wie Tag-Clouds oder Tag-Clustern sind. Zur Bewertung des Retrieval-Systems müssen differenzierte und vielfältige Anfragen gestellt werden, was auf Grund der geringen Datenbasis der Prototyp-Applikation nicht möglich ist und daher in zukünftigen Untersuchungen thematisiert werden sollen.

Das vorliegende IT-Evaluationsmodell ist damit eine modifizierte Version des DeLone

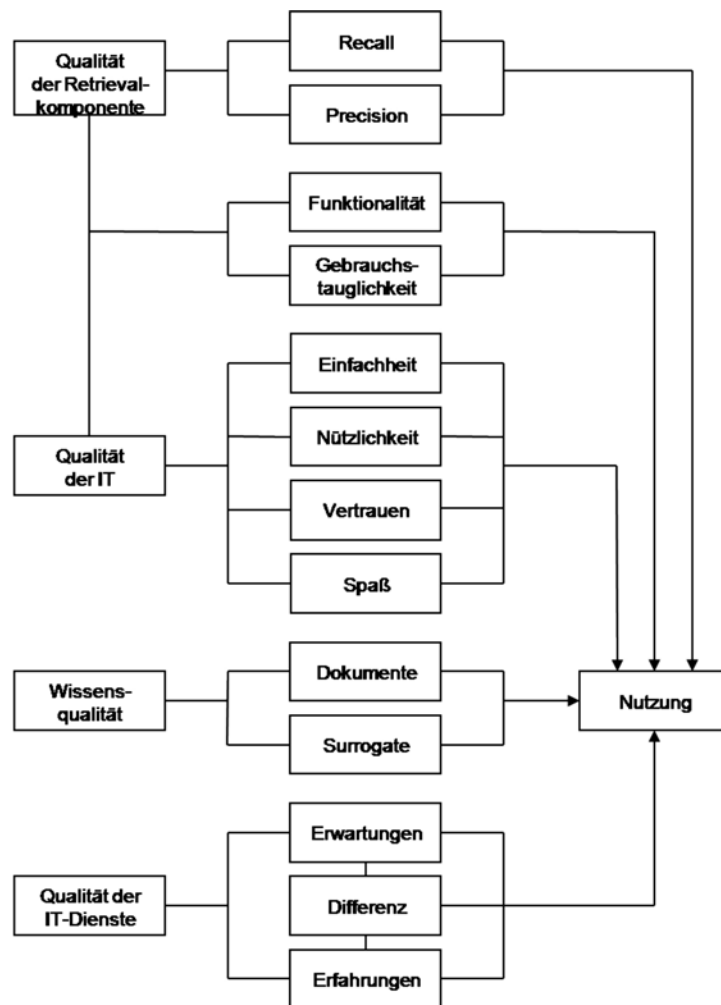


Abbildung 24: Tag-Cluster Evaluationsmodell

und McLean Modells (39; 40) und des Jenex und Olman Modells (41).

## 5.2 Ergebnisse

Um einen Vergleich zwischen herkömmlichen Tag-Clouds und den neu generierten Tag-Clustern anzustellen, wurde ein Fragebogen erstellt, welcher die im Methodenteil genannten Parameter und Indikatoren berücksichtigt.

Ein Pre-Test im Mai 2009 mit Informations Professionals der Abteilung Informationswissenschaft am Institut für Sprache und Information der Heinrich-Heine-Universität (N=6) führte zur teilweisen Ergänzung bzw. Umformulierung der Fragen und damit zu einem Update des Evaluationsbogens. Die Endfassung des Evaluationsbogens umfasst 10 übergeordnete Fragen, welche sich auf insgesamt 26 Teilfragen aufteilen. Der Test zielt bei den ersten 6 Fragen darauf ab, eine Korrelation zwischen Erwartung und Beobachtung zu ziehen. Die letzten 4 Fragen zielen auf die genannten klassischen Parameter ab. Die Evaluation von Tag-Clouds (Delicious) und Tag-Clustern wurde durch 28 Studenten des 2. Semesters im Rahmen des Tutoriums zur Vorlesung

„Wissensrepräsentation“ und zwei Mitgliedern des Fachbereichs Informationswissenschaft der Heinrich-Heine-Universität im Mai 2009 nach einer thematischen Einführung durchgeführt.

Tabelle 2 zeigt die Ergebnisse der IT Service Qualität von Tag-Clouds und Tag-Clustern. Hieraus ist erkenntlich, dass die Testpersonen Tag-Cluster als Visualisierungsmethode von Folksonomies bevorzugten. Hinsichtlich der verschiedenen Varianten der Ähnlichkeitsberechnungen konnte keine Präferenz von Seiten der Nutzer festgestellt werden, nur eine leichte Tendenz zur Bevorzugung nach Cosinus. In Bezug auf die Clustering-Algorithmen eignete sich nach Evaluation durch die Nutzer das Group-Average-Verfahren, gefolgt vom Complete-Link-Verfahren, am besten. Als weiteres Ergebnis konnte festgehalten werden, dass bezüglich der zwei angebotenen Ran-

king-Algorithmen das WDF\*ITF-Verfahren leicht besser bewertet wurde als das Ranking nach absoluter Häufigkeit. Der Erwartungswert von Nutzerseite war in Bezug auf das Einstellen des Schwellenwertes mit durchschnittlich 6.07 von 7.00 Punkten sehr hoch. Zwar konnte die Beobachtung diesen Wert nicht erreichen, war aber mit einem Wert von 5.60 Punkten ebenfalls hoch. Gleiches ist bei dem Erwartungswert bezüglich der Darstellung der Ähnlichkeit über die Linienstärke zu beobachten. Dieser Wert lag mit 6.37 Punkten von 7 noch höher und konnte mit einem hohen Beobachtungswert von 5.55 Punkten bestätigt werden.

Da der Fragebogen auch ein freies Anmerkungsfeld zur Verfügung stellte, nutzten die Evaluierenden dieses, um Verbesserungswünsche in Bezug auf die visualisierte Ähnlichkeit zu äußern. Vorschläge waren beispielsweise die Anbringung der Ähnlichkeitswerte an den Linien (Fragebogen 8), die Visualisierung von ähnlichen Tags in geringerer Distanz zueinander (Fragebogen 12) oder eine farbliche Abstufung der Linien je nach Ähnlichkeit (Fragebogen 10).

Tabelle 2: Ergebnisse der Evaluation hinsichtlich der IT service qualität

IT service quality indicator	Erwartungswert (SD)	Beobachtungswert (SD)	Differenz (SD)
1. Visualisierung von Folksonomies			
Tag-Clouds	5.69 (0.76)	3.52 (1.12)	-2.17
Tag-Cluster		5.39 (0.83)	-0.30
2. Berechnung der Ähnlichkeiten			
Dice	5.59 (1.05)	4.87 (1.36)	-0,72
Jaccard		4.73 (1.31)	-0.85
Cosine		5.10 (1.06)	-0.49
3. Cluster-Algorithmen			
Single-Link Verfahren	5.83 (1.00)	3.87 (1.50)	-1.96
Complete-Link Verfahren		4.76 (1.30)	-1.07
Group-Average Verfahren		5.10 (1.09)	-0.73
4. Relevance Ranking			
Absolute Häufigkeit	5.46 (1.29)	4.39 (1.60)	-1,07
WDF*ITF		4.86 (1.43)	-0.61
5. Tag-Cluster Charakteristiken			
Schwellenwert	6.07 (0.88)	5.60 (1.28)	-0.47
Linien-Stärke	6.37 (0.85)	5.55 (1.30)	-0.82

N: zwischen 28 und 30; SD: Standard-Abweichung; Skala: 1 (stimme nicht zu) bis 7 (stimme absolut zu)

In Bezug auf die Qualität des IT-Systems (Tab.3), konnte mit Hilfe der Evaluation festgestellt werden, dass sich sehr signifikante statistische Unterschiede zwischen Tag-Clouds und Tag-Clustern in Bezug auf die wahrgenommene Nützlichkeit (*perceived usefulness*), Vertrauen (*trust*) und Spaß (*perceived enjoyment*) ergeben. Nutzer präfe-

rierten in den genannten Punkten teilweise sehr deutlich Tag-Cluster. Tag-Cluster wurden nicht nur als nützlicher wahrgenommen, sondern die Benutzung brachte den Nutzern auch mehr Spaß und der visualisierte Content wurde von ihnen als vertrauenswürdiger eingestuft.

Tabelle 3: Evaluationsergebnisse hinsichtlich der Qualität des IT-System

Qualitäts-Indikator des IT-Systems	Durchschnitt Tag-Clouds (SD)	Durchschnitt Tag-Cluster (SD)	Signifikanz-Unterschied?
Einfachheit der Bedienung	5.17 (1.34)	5.57 (1.22)	Keine Signifikanz
Wahrgenommene Nützlichkeit	4.10 (1.35)	5.66 (1.23)	$\alpha < 0.001$
Vertrauen	2.90 (1.01)	4.43 (1.17)	$\alpha < 0.001$
Spaß	4.10 (1.37)	5.45 (1.35)	$\alpha < 0.001$

N: zwischen 28 und 30; SD: Standard-Abweichung; Skala: 1 (gar nicht) to 7 (absolut)



**Tabelle 4: Korrelation (Pearson, beidseitig) der IT-Qualitäts-Indikatoren**

	Bedienung	Nützlichkeit	Vertrauen	Spaß
Bedienung	1			
Nützlichkeit	+0.42	1		
Vertrauen	+0.32	- 0.08	1	
Spaß	+0.31	+0.40	+0.19	1

Zudem konnte eine positive Korrelation zwischen der Bedienungsfreundlichkeit (*ease of use*) und allen anderen Indikatoren sowie mit Spaß identifiziert werden (Tab. 4). Die wahrgenommene Nützlichkeit korreliert stark mit der Bedienungsfreundlichkeit (+0.42) und mit Spaß (+0.40). Insgesamt scheint es, dass der Indikator Spaß eine große Bedeutung bei der Benutzung von Tag-Clustern besitzt, was die Akzeptanz einer solchen Retrieval-Schnittstelle durch die Nutzer erleichtern kann. Daher wurden im Folgenden auch die Korrelationen von Spaß mit allen anderen Indikatoren betrachtet. Eine hohe positive Korrelation (+0.43) ergab sich zwischen Spaß und den Erwartungen in Bezug auf die Einstellung des Schwellenwertes zur Clusterbildung mittels des Schiebereglers (Tab. 5). Je größer die Erwartungen der Testpersonen hinsichtlich der Justierung des Schiebereglers waren, umso stärker war der Spaß-Effekt bei der Clusterbildung (und umgekehrt) zu beobachten.

**Tabelle 5: Korrelationen (Pearson, beidseitig) zwischen der Erwartung der Nützlichkeit der Schwellenwerteinstellung und der IT-Qualitäts-Indikatoren**

Korrelation der Erwartung der Nützlichkeit der Schwellenwerteinstellung	
Bedienung	- 0.01
Nützlichkeit	+ 0.16
Vertrauen	+ 0.08
Spaß	+ 0.43

Offensichtlich ist das Einstellen der Clusterauflösung via Schieberegler und das daraus resultierende Hinzufügen von Tags durch einen geringeren Schwellenwert bzw. eine Abnahme von Tags durch einen höheren Schwellenwert mit Spaß von Seiten der Nutzer verbunden. Das Resultat dieser Option ist je nach Einstellung entweder eine differenzierteres Bild der Treffermenge oder die Visualisierung der Basis-Strukturen (Themen) in der Treffermenge. Unterstützt wird diese Aussage auch durch Kommentare im freien Anmerkungsfeld des Evaluationsbo-

gens. So macht es Spaß „[...] die Schwellenwerte zu verändern und den Einfluss auf die Bookmarks zu sehen [...]“ (Fragebogen 8) und „die Interaktivität durch Einstellung des Schwellenwertes ist ein erheblicher Spaßfaktor, der Tag-Cluster als Visualisierung attraktiv macht“ (Fragebogen 5). Zusammenfassend lässt sich sagen, dass Tag-Cluster sehr gut von den Nutzern angenommen wurden und einen klaren Vorzug gegenüber Tag-Clouds erhielten. Bei einer Implementierung sind jedoch einige Dinge zu beachten. So sollte auf Grund der Ergebnisse der Evaluation eine Bildung der Tag-Cluster über das Group-Average Verfahren erfolgen. Hinsichtlich der Berechnung der Tag-Ähnlichkeiten und der Ranking-Variante konnte keine deutliche Präferenz von Seiten der Nutzer festgestellt werden, so dass dies über eine Erprobung der Applikation erfolgen sollte. Da der Spaß bei der Clusterbildung erheblich durch die Einstellung des Schwellenwertes erfolgte, sollte diese Möglichkeit der Interaktivität, beispielsweise über einen Schieberegler, auch bei einer Implementierung zur Verfügung gestellt werden.

## 6 Abschlussbemerkung und Ausblick

Der in Web2.0-Diensten von Nutzern generierte Content steigt stetig an und bedarf neben der Strukturierung auch der Verbesserung des Zugangs, um ein effizientes Retrieval für die Nutzer zu gewährleisten. Daher wurde in dieser Arbeit mit Tag-Cluster ein alternativer Ansatz vorgestellt, der den derzeitigen Defiziten herkömmlicher Visualisierungsmethoden von Folksonomies versucht entgegen zu wirken und den Forderungen den semantischen Anreicherungen zu entsprechen nachkommt. Die Nutzung syntagmatischer Relationen auf Basis der Tag-Co-Occurrences und das anschließende Clustering ermöglicht eine Reihe von Möglichkeiten und Verbesserungen hinsichtlich der visuellen Realisierung von Folksonomies. Zusammenfassend lässt sich festhalten:

- das Clustering ermöglicht eine große-

re kohärente visuelle Verteilung als die alphabetische Anordnung;

- den Forderungen der Einbeziehung semantischer Bezüge wird durch diesen Ansatz entsprochen;
- „Gleichberechtigung“ aller Tags bei der Clusterbildung ermöglicht, dass auch neue Tags und Bereiche die Chance auf die Visualisierung haben; somit wird die eingangs erwähnte fehlende zeitliche Abgrenzung einer Tag-Cloud kompensiert;
- nach einer initialen Suche über Tag-Clouds kann die Fülle an Treffern durch Visualisierung der Tags in der Treffermenge in Form von Clustern ein weiterer/zusätzlicher thematischer Überblick geboten werden;
- durch die aktive Beeinflussung der Darstellung wird die Interaktivität für Nutzer und Anbieter realisiert;
- neue Tags im Cluster durch einen niedrigeren Schwellenwert geben Nutzern einen differenzierten Einblick, eine verminderte Anzahl von Tags auf Grund eines höheren Schwellenwerts lassen den Themenschwerpunkt erkennen;
- durch die Nutzung von Tag-Clustern sind User in der Lage, ihre Suchanfrage zu reformulieren und aktiv die Treffermenge anzupassen.

Neben den zahlreichen Vorteilen, gibt es auch einige Verbesserungen der Retrieval-Software, welche in Zukunft Gegenstand der Betrachtung sein könnten. So ist der Cluster-Aufbau in zeitlicher Hinsicht etwas ineffizient und muss für einen Gebrauch der neu entwickelten Retrieval-Schnittstelle innerhalb eines Web2.0-Dienstes optimiert werden. Mögliche zukünftige Betrachtungen sind zudem die informationslinguistische Tag-Bearbeitung wie beispielsweise die relativ einfache Generalisierung der Numeri oder die Identifizierung von Synonymen durch das Clustering. Auch die Bildung von Sub-Clustern ist ein interessanter Forschungspunkt, welcher ein differenzierteres hierarchisches Clustering und damit ein verbessertes Retrieval ermöglicht. Auf Grund der Ergebnisse dieser und anderer Arbeiten kann festgehalten werden, dass die Betrachtung semantischer Strukturen innerhalb von Folksonomies einen wichtigen Ansatzpunkt bildet, um den Zugang zu den stetig wachsenden Ressourcen in Web2.0-Diensten zu gewährleisten und zu verbessern. Mit Tag-Cluster wurde eine visuelle Retrieval-Schnittstelle geschaffen, die diesen Aspekt aufgenommen hat und das Potential dieses Ansatzes erkennen lässt. Der vorliegende Artikel ist eine deutsche, erweiterte Version eines Fachbeitrages der Hawaii International Conference on System Sciences (HICSS 2010).

## Literaturverzeichnis

- (1) Sinclair, J., & Cardew-Hall, M. (2008). The Folksonomy Tag Cloud: When Is It Useful? *Journal of Information Science*, 34 (1), S. 15–29.
- (2) Mathes, A. (2004). *Folksonomies - Cooperative Classification and Communication Through Shared Metadata*. Abgerufen am 25. Mai 2009 von adammathes.com: <http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html>
- (3) Dieberger, A., Dourish, P., Höök, K., Resnick, P., & Wexelblatt, A. (2000). Social Navigation – Techniques for Building More Usable Systems. *Interactions* 7, S. 36–45.
- (4) Rivadeneira, A. W., Gruen, D. M., Muller, M. J., & Millen, D. R. (2007). Getting Our Head In The Clouds: Toward Evaluation Studies of Tagclouds. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (S. 995–998). New York: ACM.
- (5) Hearst, M. A., & Rosner, D. (2008). Tag clouds: Data analysis tool or social signaller? In *Proceedings of 41st Hawaii International Conference on System Sciences (HICSS 2008)*, Social Spaces minitrack.
- (6) Russell, T. (2006). Cloudalicious: Folksonomy Over Time. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries* (S. 364). New York: ACM.
- (7) Halvey, M., & Keane, M. (2007). An Assessment of Tag Presentation Techniques. In *Proceedings of the 16th International Conference on World Wide Web* (S. 1313–1314).
- (8) Bateman, S., Gutwin, C., & Nacenta, M. (2008). Seeing Things in the Clouds: the Effect of Visual Features on Tag Cloud Selections. In *Proceedings of the Nineteenth ACM Conference on Hypertext and Hypermedia, Pittsburgh, PA, USA* (S. 193–202). New York: ACM.
- (9) Hassan-Montero, Y., & Herrero-Solana, V. (2006). Improving Tag-Clouds as Visual Information Retrieval Interfaces. In *Proceedings of International Conference on Multidisciplinary Information Sciences and Technologies (InSciT2006)*. Mérida, Spanien.
- (10) Begelman, G., Keller, P., & Smadja, F. (2006). Automated Tag Clustering: Improving Search and Exploration in the Tag Space. In *Proceedings of the WWW 2006 Collaborative Web Tagging Workshop at WWW2006*. Edinburgh, Scotland.
- (11) Angeletou, S., Sabou, M., Specia, L., & Motta, E. (2007). Bridging the Gap Between Folksonomies and the Semantic Web: An Experience Report. In *Workshop: Bridging the Gap between Semantic Web and Web 2.0, European Semantic Web Conference* (S. 30–43).
- (12) Zhang, L., Wu, X., & Yu, Y. (2006). Emergent Semantics from Folksonomies: A Quantitative Study. In *Journal on Data Semantics VI (Lecture Notes in Computer Science)* (S. 168–186). New York: Springer.
- (13) Kaser, O., & Lemire, D. (2007). Tag-Cloud Drawing: Algorithms. In *Proc. WWW'07 Workshop on Taggings and Metadata for Social Information Organization*.
- (14) Knuth, D. E., & Plass, M. F. (1981). Breaking Paragraphs Into Lines. *Software – Practice and Experience* 11, S. 1119–1184.
- (15) Skiena, S. S. (1997). *The Algorithm Design Manual*. Berlin: Springer.
- (16) Shaw, B. (December 2005). Utilizing Folksonomy: Similarity Metadata from the Del.icio.us System. Project Proposal. Abgerufen am 9. 5 2009 von <http://www.metablake.com/webfolk/web-project.pdf>
- (17) Steinbach, M., Karypis, G., & Kumar, V. (2000). A Comparison of Document Clustering. In *Proceedings of KDD Workshop on Text Mining*.
- (18) Cattuto, C., Schmitz, C., Baldassarri, A., Servidio, V. D., Loreto, V., Hotho, A., et al. (2007). Network Properties of Folksonomies. *AI Communications*, 20 (4), S. 245–262.
- (19) Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., & Stumme, G. (2009). Evaluating Similarity Measures for Emergent Semantics of Social Tagging. In *Proceedings of the 18th International Conference on World Wide Web*. New York: ACM.
- (20) Li, R., Bao, S., Yu, Y., Fei, B., & Su, Z. (2007). Towards Effective Browsing of Large Scale Social Annotations. In *Proceedings of the 16th International Conference on World Wide Web*. New York: ACM.
- (21) Huang, Y.-C., Hung, C.-C., & Hsu, J. Y. (2008). You Are What You Tag. In *Proceedings of AAAI 2008 Spring Symposium Series on Social Information Processing*.
- (22) Schrammel, J., Leitner, M., & Tscheligi, M. (2009). Semantically Structured Tag Clouds: An Empirical Evaluation of Clustered Presentation Approaches. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems* (S. 2037–2040).
- (23) Capocci, A., & Caldarelli, G. (2008). Folksonomies and Clustering in the Collaborative System CiteULike. *Journal of Physics A-Mathematical and Theoretical*, S. 1–7.
- (24) Rasmussen, E. (1992). Clustering Algorithms. In W. Frakes, & R. Baeza-Yates, *Information Retrieval: Data Structures and Algorithms* (S. 419–442). Upper Saddle River, NJ, USA: Prentice-Hall, Inc.
- (25) Peters, I. (2009). Folksonomies: Indexing and Retrieval on the Web 2.0. In *Knowledge and Information. Studies in Information Science*, Vol. 1. München: Saur.
- (26) Parasuraman, A., Zeithaml, V., & Berry, L. (1988). SERVQUAL: A Multiple-Item Scale for Measuring Consumer Perceptions of Service Quality. *Journal of Retailing*, 64(1), S. 12–40.
- (27) Buttle, F. (1996). SERVQUAL: Review, Critique, Research Agenda. *European Journal of Marketing* 30(1), S. 8–32.
- (28) Pitt, L., Watson, R., & Kavan, C. (1995). Service Quality: A Measure of Information Systems Effectiveness. *MIS Quarterly*, 19(2), S. 173–187.
- (29) Van Dyke, T. P., Kappelman, L., & Prybutok, V. (1997). Measuring Information Systems Service Quality: Concerns on the Use of the SERVQUAL Questionnaire. *MIS Quarterly*, 21(2), S. 195–208.
- (30) Pitt, L., Watson, R., & Kavan, C. (1997). Measuring Information Systems Quality: Concerns for a Complete Canvas. *MIS Quarterly*, 21(2), S. 209–221.
- (31) Kettinger, W., & Lee, C. (1997). Pragmatic Perspectives on the Measurement of Information Systems Service Quality. *MIS Quarterly*, 21(2), S. 223–240.
- (32) Van Dyke, T. P., Prybutok, V. R., & Kappelman, L. A. (1999). Cautions on the Use of the SERVQUAL Measure to Assess the Quality of Information Systems Services. *Decision Sciences*, 30 (3), S. 1–15.
- (33) Kettinger, W., & Lee, C. (1999). Replication of Measures in Information Systems Research: The Case of SERVQUAL. *Decision Sciences*, 30(3), S. 893–899.
- (34) Jiang, J., Klein, G., & Carr, C. (2002). Measuring Information System Service Quality: SERVQUAL from the Other Side. *MIS Quarterly*, 26(2), S. 145–166.
- (35) Davis, F. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13, S. 319–340.
- (36) Kim, B., & Han, I. (2009). The Role of Trust Belief and its Antecedents in a Community-driven Knowledge Environment. *Journal of the American Society for Information Science and Technology*, 60(5), S. 1012–1026.
- (37) Lin, C., Wu, S., & Tsai, R. (2005). Integrating Perceived Playfulness Into Expectation-Conformation Model for Web Portal Context. *Information and Management*, 42(5), S. 683–693.
- (38) Van der Heyden, H. (2004). User Acceptance of Hedonic Information Systems. *MIS Quarterly*, 28(4), S. 695–703.
- (39) DeLone, W., & McLean, E. (1992). Information Systems Success: The Quest for the Dependent Variable. *Information Systems Research*, 3, S. 60–95.
- (40) DeLone, W., & McLean, E. (2003). The DeLone and McLean Model of Information Systems Success: A ten-year Update. *Journal of Management Information Systems*, 19(4), S. 9–30.
- (41) Jennex, M., & Olfman, L. (2006). A Model of Knowledge Management Success. *International Journal of Knowledge Management*, 2(3), S. 51–68.

## ■ AUTORIN

**KATHRIN KNAUTZ, B.A.**

(Informationswissenschaft &amp; Sprachtechnologie)

Wissenschaftliche Hilfskraft

(und Studentin im Master-

Studium Informationswis-

senschaft &amp; Sprachtechnologie)

Abteilung Informationswissenschaft

Institut für Sprache und Information

Heinrich-Heine-Universität Düsseldorf

Universitätsstr. 1

40225 Düsseldorf

Kathrin.Knautz@uni-duesseldorf.de

