Themen in der Wirtschaftsinformatik: Eine Topic-Analyse der wichtigsten Konferenzen

Bachelor-Thesis

Florian Friedrichs | 1538623

Wirtschaftsinformatik

TECHNISCHE UNIVERSITÄT DARMSTADT

INFORMATION SYSTEMS

Florian Friedrichs
Matrikelnummer: 1538623
Studiengang: Bachelor of Science - Wirtschaftsinformatik

Bachelor-Thesis
Thema: "Themen in der Wirtschaftsinformatik: Eine Topic-Analyse der wichtigsten Konferenzen"

Eingereicht: 31.03.2012

Betreuer: Dipl.-Wirtsch.-Inform. Anton Pussep

Prof. Dr. Peter Buxmann
Fachgebiet Information Systems / Wirtschaftsinformatik
Fachbereich Rechts- und Wirtschaftswissenschaften
Technische Universität Darmstadt
Hochschulstraße 1
64289 Darmstadt

Ehrenwörtliche Erklärung

Ich erkläre hiermit ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig angefertigt habe. Sämtliche aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und noch nicht veröffentlicht.

Darmstadt, den 31.03.2012

Inhaltsverzeichnis

Abbildungsverzeichnis IV

Tabellenverzeichnis V

Abkürzungsverzeichnis VI

1 Einführung in das Thema 1

1.1 Problemstellung 1

1.2 Zielsetzung 2

1.3 Lösungsansätze 3

1.4 Aufbau der Arbeit 5

2 Latent Dirichlet Allocation 6

2.1 Grundlagen 7

2.1.1 Topic Models 7

2.1.2 Dirichlet Verteilung 8

2.2 Beschreibung des Verfahrens 11

2.2.1 Anwendung 14

2.2.2 Modell-Berechnung 17

2.3 Parameter und Ergebnisse 18

2.3.1 Beeinflussbare Parameter 18

2.3.2 Qualitätsmaße und Ergebnisse des Verfahrens 20

2.4 Weiterführende Informationen 22

2.4.1 Weitere Abwandlungen 22

2.4.2 Implementierungen von LDA 24

2.4.3 Weiterführende Literatur / führende Quellen 25

3 Daten und Methode 27

3.1 Daten 27

3.1.1 Kriterien 28
3.1.2 Auswahl der Daten 29
3.1.3 Struktur der Daten 31
3.1.4 Qualität der Daten 34
3.2 Prozesse 35
3.2.1 Prozess (1) - Vorbereitung 36
3.2.2 Prozess (2) – Daten-Beschaffung 37
3.2.3 Prozess (3) - Daten-Aufbereitung 42
3.2.4 Prozess (4) - Latent Dirichlet Allocation 48
3.3 Software 56
3.3.1 Systemanforderungen 56
3.3.2 Starten der Software 57
3.3.3 Vorstellung der grafischen Oberfläche 57
3.3.4 Benutzte Bibliotheken 58
3.3.5 Ergebnisse 61
4 Anwendung 63
4.1 Technische Ausstattung 63
4.2 Verarbeitung 65
4.3 Anwendung von Latent Dirichlet Allocation 67
4.3.1 Bestimmung der Anzahl an Topics 68
4.3.2 Berechnung des Topic-Modells 69
4.3.3 Anwendung des Modells 70
4.4 Interpretation der Ergebnisse 71
4.4.1 Visualisierung von Topics 71
4.4.2 Visualisierung und Berechnung der Ähnlichkeit von Dokumenten 72
4.4.3 Bestimmung relevanter Topics 74

4.4.4 Topics im zeitlichen Verlauf 78

5 Fazit 85

Anhang V

A Stoppwörter von TMT VI

B Auflistung aller Topics im untersuchten Datenbestand VII

C Die 50 bzw. 20 % häufigsten Wörter des Korpus XII

D Topic-Entwicklung von 1993 - 2011 XIII

Literaturverzeichnis XIV

Abbildungsverzeichnis

Abbildung 1 - Vereinfachte Darstellung einer Dirichlet-Verteilung [...] 9

Abbildung 2 - Dirichletverteilungen Dir(α) für unterschiedliche alpha-Werte. [...].....10

Abbildung 3 - Beispiel für eine Topic-Zuordnung einzelner Wörter in einem Dokument bei einem existierenden Topic Model. [...] 12

Abbildung 4 - Beispiel für die Ausgangslage vor Erstellung eines Topic Modells. [...] 13

Abbildung 5 - Modell von Latent Dirichlet Allocation mit der dazugehörigen Legende.....14

Abbildung 6 - Wachstum der kumulierten Menge an Veröffentlichungen von AIS-Konferenzen von 1993 bis 2011 33

Abbildung 7 - Datenwachstum der letzten 10 Jahre im Vergleich zum jeweiligen Vorjahr von 2001 bis 2011.....34

Abbildung 8 - Gesamtprozess zur Datenanalyse, unterteilt in 4 Hauptprozesse 36

Abbildung 9 - Grafische Oberfläche von SIERRA mit Beschreibungen der Bedienelemente 58

Abbildung 10 - Perplexity-Werte zur Berechnung der Topics in 30er-Schritten. [...] 68

Abbildung 11 - Log-Likelihood-Schätzer im Bereich von 0 bis 250 Iterationen 69

Abbildung 12 - Anzahl Dokumente für ausgewählte Topics von 1993 bis 2011 79

Abbildung 13 - 5 Topics mit den stärksten Zuwächsen 2006-2011. [...] 81

Abbildung 14 - 5 Topics mit den höchsten Verlusten 2006-2011. [...] 82

Abbildung 15 - 5 Topics mit den stärksten Zuwächsen 2001-2011. [...] 84

Abbildung 16 - 5 Topics mit den höchsten Verlusten 2001-2011. [...] 84

Tabellenverzeichnis

Tabelle 1 - Beispiel für K = 3 Dirichlet-Verteilungen über V = 1000 Wörter. 15

Tabelle 2 - Beispiel für ein Vokabular mit V = 1000 Wörtern. 16

Tabelle 3 - Beispiel für eine Dirichlet-Verteilung über D = 2 Dokumente und K = 3 Topics...16

Tabelle 4 - Dokumente je Sprache und Konferenz [...] 30

Tabelle 5 - Quantitative Übersicht der Daten nach Konferenz und Jahr auf der AISeL-Seite...32

Tabelle 6 - Aufbau der download_links.csv für den Webcrawler.....41

Tabelle 7 - Format der CSV-Datei für LDA 47

Tabelle 8 - Nötige Einstellungen einzelner LDA-Prozesse.....49

Tabelle 9 - Schematischer Aufbau der Datei document-topic-distributions.csv 52

Tabelle 10 - Schematischer Aufbau der Datei converted_file-usage.csv 54

Tabelle 11 - Schematischer Aufbau der Datei converted_file-top-terms.csv 54

Tabelle 12 - Schematischer Aufbau der Datei converted_file-sliced-usage.csv 55

Tabelle 13 - Schematischer Aufbau der Datei converted_file-sliced-top-terms.csv 55

Tabelle 14 - Benötigter Speicherplatz in GB nach Ordner in der Domäne [...].....65

Tabelle 15 - Gesamtlaufzeit alle Prozesse.....67

Tabelle 16 - Zwei beispielhafte Topics über mehrere Iterationen.....70

Tabelle 17 - Distanzen zwischen mehreren Dokumenten. [...].....72

Tabelle 18 - Topics im Zusammenhang mit sozialen Netzwerken. [...] 73

Tabelle 19 - Rangreihenfolgen der Topics nach Relevanz und der absoluten Häufigkeit zugeordneter Dokumente und Wörter, absteigend sortiert.....75

Tabelle 20 - Auszug aus der Topic-Liste mit den am häufigsten zugeordneten Wörtern, die markiert sind, falls diese zu den 50 häufigsten Wörtern gehören.....76

Tabelle 21 - Die 10 relevantesten Topics mit den häufigsten Dokumenten nach Aussortieren nicht-relevanter Topics. 77

Tabelle 22 - Ausgewählte Topics zum Thema "Internet" 79

Abkürzungsverzeichnis

AIS	Association of Information Systems
AISeL	Association of Information Systems Electronic Library
AMCIS	Americas Conference on Information Systems
Conf-IRM	International Conference on Information Resources Management
CSV	Comma Separated Values
ECIS	European Conference on Information Systems
GB	Gigabyte
GPU	Graphical Processing Unit
ICIS	International Conference on Information Systems
ID	Identifikator
IS	Information Systems
JVM	Java Virtual Machine
KDD	Knowledge Discovery
LDA	Latent Dirichlet Allocation
LSI	Latent Semantic Indexing
MB	Megabyte
MCIS	Mediterranean Conference on Information Systems
NLP	Natural Language Processing
OCR	Optical Character Recognition
PACIS	Pacific Asia Conference on Information Systems
PDF	Portable Document Format
SIERRA	Scientific Information Extraction for rigorous relevance analysis
TMT	Topic Modeling Toolbox
XML	Extensible Markup Language

1 Einführung in das Thema

Die digitale Revolution sorgt für eine Beschleunigung von Prozessen und beeinflusst unsere Arbeit und unser Leben.[1] Die zunehmende Digitalisierung von bereits existierendem Wissen erhöht zusammen mit einer zunehmenden Entstehung von neuem Wissen in digitaler Form die gesamten Datenbestände. Immer größere Festplatten können zwar mehr Daten speichern, aber nur durch die bloße Existenz von Informationen kann kein Mehrwert erzeugt werden.[2]

Den Überblick über die Daten zu behalten, ist eine Herausforderung. Hat beispielsweise eine Bibliothek kein System für die Strukturierung ihrer Bücher und Medien, ist es unmöglich, Informationen gezielt zu suchen und möglicherweise auch noch zu finden.[3] Um den Überblick über die zur Verfügung stehenden Informationen zu erhalten, müssen diese und ihre beschreibenden Meta-Daten erfasst, ausgewertet und in Beziehung gesetzt werden.

Welches Potenzial die Unterstützung einer Suche nach Informationen hat, beweist z.B. der Erfolg von Google. 1996 hat das Unternehmen sein Ziel deutlich formuliert: „Die Informationen der Welt zu organisieren und für alle zu jeder Zeit zugänglich und nützlich zu machen.“[4]. Das Unternehmen geht über eine bloße Suche nach Meta-Daten noch einen Schritt weiter als beispielsweise die Verzeichnisse von Bibliotheken und zeigt, dass zusätzliche wertvolle Informationen aus einem Datenbestand gewonnen werden können. Anhand verschiedener Verfahren wird die Relevanz der verschiedenen Quellen bewertet und der Benutzer erhält die für die Suche relevantesten Ergebnisse.

1.1 Problemstellung

Das Sammeln von Daten und die Extraktion von Wissen werden als „Data Mining“ definiert. Wissen besteht wiederum aus Mustern, die interessant erscheinen[5]. Interessante Muster sind „[...] schließlich solche, die allgemein gültig, nicht trivial, neu, nützlich und verständlich

---

1 Vgl. Thierse (2003), S. 56.

2 Vgl. Witten et al. (2011), S.3 ff.

3 Vgl. Blei; Lafferty (2009), S. 1.

4 Google (2012).

5 Vgl. Runkler (2010), S. 2.

[sind]“[6]. Die Muster liegen verteilt in den gesammelten Informationen und sind verborgen, bis sie durch ein Verfahren entdeckt werden. Dieser Prozess ist unabhängig davon, ob er manuell oder automatisch durchgeführt wird.[7]

Oft als Synonym für Data Mining verwendet, beschreibt „Knowledge Discovery in Databases“ (KDD) sowohl den Data-Mining-Prozess der Datensammlung und der Extraktion von Informationen (Knowledge) als auch die Vorbereitung der Daten und die Interpretation des gewonnenen Wissens.[8]

Im Bereich der Knowledge Discovery laufen die Verfahren weitgehend automatisch ab, da der zeitliche Aufwand für eine manuelle Analyse langsam, teuer und subjektiv ist.[9] Allein Zwischen- und End-Ergebnisse – das Wissen – werden durch den Menschen bewertet. Dies ist durch die Größe der vorhandenen Daten bedingt, die in ihrer Gesamtheit mit menschlichen Fähigkeiten nicht ausführlich analysiert werden können. [10]

Ein Automatismus hat den weiteren Vorteil, dass alle Informationen nach dem gleichen Verfahren klassifiziert werden können und die Ergebnisse im Gegensatz zur manuellen Verarbeitung nicht subjektiv sind[11]. Obwohl Menschen eine gute Leistung im Extra-Polieren von Informationen zeigen – auch wenn sie nur den Titel und einige Wörter oder Absätze lesen[12] – werden sie über einen großen Datenbestand nicht alle Themen identifizieren.

1.2 Zielsetzung

Im Rahmen dieser Thesis soll eine Lösung im Bereich der Knowledge Discovery diskutiert werden, mit der Trends in der Software-Industrie und häufig diskutierte Problemstellungen identifiziert werden können. Die Auswertung soll möglichst automatisch ablaufen und für eine praktische Nutzung als Software implementiert werden. Das Ziel ist die Offenlegung von verborgenen Strukturen, die den Inhalt von wissenschaftlichen Veröffentlichungen in der

6 Vgl. Runkler (2010), S. 2.

7 Vgl. Witten et al. (2011), S. 4.

8 Vgl. Fayyad et al. (1996), S. 37.

9 Vgl. ebd., S. 38.

10 Vgl. Han et al. (2004), S. 1.

11 Vgl. Chang (2009), S. 4 ff.

12 Vgl. ebd., S. 4.

Softwareindustrie beschreiben, Interpretationen zulassen und über einen definierten Zeitraum beobachtbar sind.

Die Offenlegung der beschriebenen Informationen ist wichtig, da in der Software-Industrie Veränderungen schnell stattfinden[13]. Es ergeben sich immer wieder neue Anwendungszwecke. Die neuen Anwendungen ergeben sich zum einen durch externe Einflüsse wie z.B. leistungsstärkere Hardware – das mooresche Gesetz besagt die Verdopplung der Rechenleistung alle 18 Monate[14] [15] – und vor allem durch neue Ideen im Bereich der Software. Google, Facebook, eBay, Amazon – um nur einige zu nennen – sind Größen im Internet, die neue Technologie in damals jungen Märkten zu einer Reife brachten, die ihnen eine große Marktmacht zusichert. Als Pionier in einem Markt tätig zu werden, hat also Vorteile.[16]

Die Identifikation von Trends und neuen Märkten mit wirtschaftlichem Potenzial ist bei den vielfältigen Forschungsfeldern in der Wirtschaftsinformatik eine Herausforderung. Es gibt weltweit zahlreiche Verlage, Journale, Konferenzen und Forschungsgruppen, die wissenschaftliche Publikationen veröffentlichen. Damit bei einem quantitativen Wachstum der Veröffentlichungen die Qualität der marktrelevanten Bewertungen gewährleistet werden kann, ist die Anwendung von Verfahren aus dem Bereich von Knowledge Discovery sinnvoll.

1.3 Lösungsansätze

Die Suche nach einem geeigneten Verfahren zur Klassifizierung von Texten erweist sich als schwierige Aufgabe. Vom Brocke beschreibt verschiedene Möglichkeiten für die Identifizierung von relevanten Publikationen zu einem Thema.[17] So können Stichwörter von ausgewählten Arbeiten als Datengrundlage dienen, um weitere Arbeiten und deren Stichwörter in den zu untersuchenden Datenbestand aufzunehmen, die mindestens ein übereinstimmendes Stichwort vorweisen können. Eine andere Möglichkeit stellt die Vorwärts- bzw. Rückwärtssuche dar. In diesem Fall werden weitere Arbeiten durch die Erwähnung der im Datenbestand vorliegenden Arbeiten (vorwärts) oder die in der Datengrundlage

[13] Vgl. Pussep et al. (2012), S. 1.

[14] Vgl. Intel (2005), S. 1.

[15] Moore hat eine konkrete zeitbezogene Aussage selbst nie getroffen, wie aus der Quelle ersichtlich wird.

[16] Vgl. Lieberman; Montgomery (1988), S. 25.

[17] Vgl. vom Brocke et al. (2009), S. 2 ff.

vorhandenen Verweise (rückwärts) in den Datenbestand aufgenommen. Die genannten Verfahren können über mehrere Iterationen ausgeführt werden, bis eine ausreichende Anzahl an Publikationen für weitere Analysen vorhanden ist.[18]

Zur Generierung einer Datengrundlage ist ein erster Überblick über die Thematik notwendig. Der Bücherbestand der Bibliothek des Fachbereichs Informatik soll für die Identifizierung möglicher Grundlagen dienen. Mit der Speicherung, Klassifizierung und Interpretation von Daten beschäftigen sich die Bereiche „Datenbank-Systeme“ und „Information Retrieval“.

In Frage kommende Bücher werden anhand ihres Titels ausgewählt. Besonders interessant erscheinen Titel mit „Data Mining“, „Information Retrieval“, „Analysis“ und „Text“. Bei einer detaillierten Betrachtung ausgewählter Bücher lassen sich Gemeinsamkeiten erkennen wie z.B. die immer wiederkehrende Verwendung von Stichwörtern wie „pattern recognition“, „genetic niching“, „thesaurus“ und „automatic indexing“.

In mehreren Büchern wird das Verfahren „Latent Semantic Indexing“ (LSI) beschrieben, dass auf die Problemstellung dieser Thesis passt.[19] [20] [21] Das Verfahren kann für eine vordefinierte Menge an Stichwörtern schätzen, ob diese mit einem Dokument – etwa eine Publikation – in Verbindung steht. Das Verfahren ist patentiert und wurde erstmals von Deerwester[22] beschrieben.

Über die Vorwärtssuche soll ersichtlich werden, ob es bereits Verfahren gibt, die möglicherweise auf LSI basieren. Diese Suche kann durch Google Scholar[23] unterstützt werden. Die spezifische Suchmaschine beinhaltet Meta-Daten wissenschaftlicher Dokumente als Datengrundlage und erfasst Zitationen. Dies ermöglicht eine Vorwärts- und Rückwärtssuche von Publikationen. Für Deerwester[24] sind 6898 Zitationen verzeichnet. Die Ergebnisliste der gefundenen Dokumente ist nach der Häufigkeit der jeweiligen Zitationen sortiert. So kann auf eine gewisse Relevanz geschlossen werden. Bei Prüfung der ersten 100

---

18 Vgl. vom Brocke et al. (2009), S. 4.

19 Vgl. Grossmann; Frieder (2004), S. 70 ff.

20 Vgl. Baeza-Yates; Ribeiro-Neto (1999), S. 44-46.

21 Vgl. Westphal; Blaxton (1998), S. 399-406.

22 Deerwester et al. (1990).

23 http://scholar.google.com.

24 Deerwester et al. (1990).

Einträge anhand ihrer Titel und Kurzbeschreibungen sind viele praktische Anwendungen und leichte Modifikationen von LSI aufgelistet. Jedoch ist in der Ergebnisliste an fünfter Stelle ein neues Verfahren verlinkt, dass LSI als Grundlage benutzt.[25] Das Verfahren trägt die Bezeichnung „Latent Dirichlet Allocation“ (LDA) und wurde erstmals von Blei[26] beschrieben.

LDA kann diskutierte Themen eines Datenbestandes selbstständig erkennen und den Anteil jedes Themas an einem Dokument berechnen. So besteht die Möglichkeit, sich einen Überblick über die behandelten Themen eines Datenbestandes zu verschaffen und bei der Auswahl von Themen zugehörige Dokumente zu identifizieren. Zugleich können die Daten nach weiteren Kriterien ausgewertet werden, beispielsweise nach der Zeit, um Trend-Analysen durchzuführen.[27] Damit scheint LDA die Ziele dieser Thesis erfüllen zu können und soll detaillierter betrachtet werden.

1.4 Aufbau der Arbeit

In Kapitel 2 wird das Verfahren LDA detaillierter ausgeführt. Dabei stehen insbesondere die Funktionsweise, Parameter und Ergebnisse im Fokus. Die Erklärung ist auf Verständnis ausgelegt und nur soweit mathematisch beschrieben, wie es nötig ist. In Kapitel 3 stehen die zu untersuchenden Daten, deren Beschaffung und Verarbeitung im Mittelpunkt. Des Weiteren wird die zu implementierende Software vorgestellt. Kapitel 4 stellt die Ergebnisse bei Anwendung der Methode auf die zu untersuchenden Daten vor. Abschließend werden in Kapitel 5 die Erkenntnisse aus der Bearbeitung der Thesis diskutiert.

[25] Vgl. Google (2012).

[26] Blei et al. (2003).

[27] Vgl. Griffiths; Steyvers (2004), S. 5228.

2 Latent Dirichlet Allocation

Der Algorithmus LDA wurde erstmals von Blei[28] beschrieben und beruht auf der Annahme, dass Texte eine latente Dirichlet-Verteilung über Themen beinhalten, die auch als latente Struktur bezeichnet werden kann[29]. LDA kann ohne Anlernen verwendet werden und soll die latente Struktur durch die Anwendung mehrerer statistischer Verfahren aufdecken. Es basiert auf den Arbeiten von Deerwester[30] zu „Latent Semantic Indexing“ (LSI) und „probabilistic LSI“ von Hofmann[31]. Die Grundlagen bilden Topic-Modelle und statistische Verteilungen wie die Dirichlet- und die Multinomial-Verteilung.[32]

Um ein Beispiel zu geben, wie LDA angewendet werden kann: Die Bücher einer Bibliothek sollen nicht nur kategorisch, sondern auch inhaltlich eingeordnet werden können. Die vorliegenden Informationen der Bücher sind homogen in ihrer Beschaffenheit wie z.B. die Titel und Herausgeber. Diese Informationen können übersichtlich erfasst und einheitlich dargestellt werden. Die Inhalte der Bücher sind bei genauer Betrachtung jedoch sehr heterogen. Es gibt Text, Tabellen, Bilder, Formeln und viele Elemente mehr. Eine automatische Einordnung von Inhalten durch LDA kann helfen, Informationen zu organisieren und themenorientiert wiederzugeben.[33]

LDA ordnet Dokumenten in bestimmten Verhältnissen unterschiedlichen Themen zu, die aus dem Text extrahiert werden. Ein Thema besteht aus einer Ansammlung von Wörtern. Es können folgende Informationen gewonnen werden[34]:

- Ähnlichkeit von Dokumenten
- Diskutierte Themen je Dokument
- Wörter, die einem Thema angehören

Das Verfahren ist universell auf alle Arten von Text einsetzbar, es unterscheidet beispielsweise nicht zwischen Titel, Inhalt und Stichwörtern.

---

28 Vgl. Blei et al. (2003), S. 994.

29 Vgl. ebd., S. 994.

30 Deerwester et al. (1990).

31 Hofmann (1999).

32 Vgl. Blei; Lafferty (2009), S. 2 f.

33 Vgl. ebd., S. 1.

34 Vgl. ebd., S. 6.

Beispiele für interessante Untersuchungen liefern unter anderem Griffiths[35] mit einer ausführlichen Analyse der Veröffentlichungen aus den „Proceedings oft the National Academy of Sciences“ über einen Zeitraum von 1991 bis 2001, Ramage[36] mit einer Echtzeitanalyse von oft diskutierten Themen in Twitter und deren Kategorisierung und Schieber[37] mit einer ähnlichen Analyse bestimmter Themen in Twitter.

In diesem Kapitel werden neben den Grundlagen zur Funktionsweise von LDA die Parameter und Interpretation der Ergebnisse beschrieben. Ergänzt wird die Theorie durch weiterführende Informationen, z.B. welche Implementierungen es von LDA bereits gibt.

2.1 Grundlagen

Im Wesentlichen ist LDA ein Verfahren, dass sich aus der Anwendung von mehreren statistischen Verteilungen basierend auf einem „Topic Model“ zusammensetzt.[38] Das Topic Modell beschreibt eine Sichtweise auf inhaltliche Strukturen in Texten. Für das grundlegende Verständnis der Funktionsweise ist eine Erklärung von Topic Modellen notwendig. Die zentrale Frage lautet: Welche Themen werden in einem Text behandelt?

Auf diesem Modell aufbauend benutzt LDA neben multinomialen Verteilungen die Dirichlet-Verteilung. Ihre zentrale Frage lautet im Zusammenhang mit LDA: Wie sieht die Verteilung von Themen in Dokumenten und von Wörtern in Themen aus?

2.1.1 Topic Models

„Topic Models“ basieren auf der Idee, dass Texte eine inhaltliche Struktur von zugrunde liegenden Topics besitzen.[39] Ein Topic bzw. Thema wird als eine Verteilung über ein konstantes Vokabular von Wörtern definiert, die eine gewisse Kohärenz aufweisen, also

35 Griffiths; Steyvers (2004).

36 Ramage et al. (2010).

37 Sommer et al. (2011).

38 Vgl. Blei; Lafferty (2009), S. 2 f.

39 Vgl. Carenini et al. (2011), S. 43.

thematisch zusammengehören[40]. Im weiteren Verlauf dieser Thesis wird der Begriff „Topic“ verwendet werden, um die Terminologie der Topic Modelle beizubehalten.

Das Modell dient der Offenlegung der verborgenen semantischen Struktur einer Sammlung von Dokumenten.[41] Jedes Dokument besitzt jeweils eine Verteilung über alle Topics. So kann ein Dokument bestimmten Themen wahrscheinlicher als anderen Themen zugeordnet werden.[42]

Topic Models können hierarchisch oder flach aufgebaut sein. Ein hierarchisches Modell ist in der Lage, eine Art Inhaltsverzeichnis mit mehreren Ebenen über den zugrunde liegenden Text zu bilden. Ein flaches Modell hingegen ist eine ungeordnete Abfolge von Topics ohne hierarchische Differenzierung. [43]

Die Komplexität des Modells unterscheidet sich je nach Typ der zugrunde liegenden Dokumente. Aufzeichnungen von Unterhaltungen bilden beispielsweise einen hohen Komplexitätsgrad, da die Inhalte nicht strukturiert sind und sich verschiedene Topics überschneiden können. Bücher bieten hingegen die geringste Komplexität, da sie bereits inhaltlich strukturiert und die Themen gegeneinander abgegrenzt sind.[44] Treten in einer Sammlung von Dokumenten verschiedene Komplexitätsgrade auf, muss das Modell den jeweils höchsten Komplexitätsgrad abbilden können.

2.1.2 Dirichlet Verteilung

Die Dirichlet-Verteilung geht auf den deutschen Mathematiker Peter Gustav Lejeune Dirichlet zurück und gehört zu den stetigen und multivariaten Wahrscheinlichkeitsverteilungen. Sie ist eine Verteilung über multinomiale Verteilungen. [45] Zur Erklärung der Funktionsweise von LDA werde ich die Verteilung grundsätzlich visuell anhand von Beispielen beschreiben.

---

40 Vgl. Blei; Lafferty (2009), S. 2 f.

41 Vgl. ebd., S. 1.

42 Vgl. ebd., S. 3.

43 Vgl. Carenini et al. (2011), S. 43 f.

44 Vgl. ebd., S. 44.

45 Vgl. Wikipedia (o. J.): Dirichlet-Verteilung.

Sei K die Anzahl an Zufallsvariablen und $\vec{\alpha}$ ein Vektor über alle Zufallsvariablen, dann kann die Dirichlet-Verteilung mit $Dir_K(\vec{\alpha})$ beschrieben werden. Abbildung 1 zeigt eine Draufsicht auf die eine Dirichlet-Verteilung, die dreidimensional wie eine Erhebung vorzustellen ist. Die Punkte a und b kennzeichnen unterschiedliche Maxima – bzw. den Peak – der Verteilung. Die daneben abgebildeten Diagramme zeigen die äquivalenten Verteilungen. Die Summe der Balken beschreibt die relativen Verhältnisse der K Zufallsvariablen und ergibt in Summe 1.

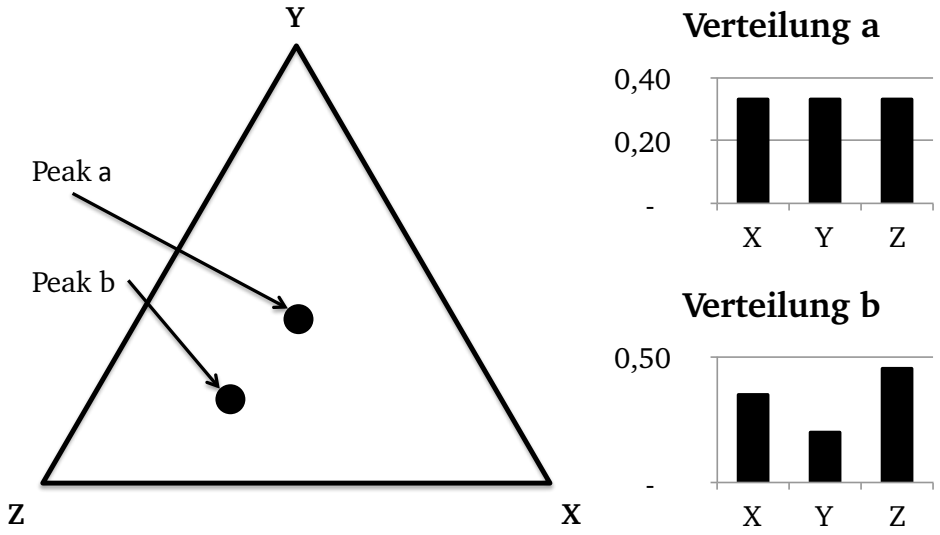


Abbildung 1 - Vereinfachte Darstellung einer Dirichlet-Verteilung von $Dir_3(x, y, z)$ mit (a) $Dir_3(1,1,1)$ und (b) $Dir_3(35,20,45)$. Die äquivalenten Diagramme zeigen die dazugehörigen Verteilungen.

Die Dirichlet-Verteilung wird auch Verteilung über Verteilungen bezeichnet und kann beispielhaft anhand von Würfeln beschrieben werden[46]. Während eine multinomiale Verteilung die Wahrscheinlichkeit für eine gewürfelte Augenzahl anzeigt, beschreibt die Dirichlet-Verteilung die Wahrscheinlichkeit von multinomialen Verteilungen, also wie wahrscheinlich es z.B. ist, eine eins, fünf und sechs zu würfeln. Werden die 6 Würfelseiten eines Würfels jeweils 10mal geworfen, kann sich eine symmetrische Wahrscheinlichkeit $Dir(10, 10, 10, 10, 10, 10)$ mit identischen Würfelfolgen ergeben. Steigt die Wahrscheinlichkeit, dass die eins in einer der Verteilungen öfter geworfen wird, könnte die Dirichlet-Verteilung mit $Dir(20, 10, 10, 10, 10, 10)$ beschrieben werden.

[46] Vgl. Wikipedia (o. J.): Dirichlet-Verteilung.

Je kleiner die Werte in α sind, desto deutlicher werden die Unterschiede in den Verteilungen. So lassen sich die Topics deutlicher gegeneinander abgrenzen. Wird α jedoch zu klein gewählt, können weniger Gemeinsamkeiten zwischen den Topics hergestellt werden und die Ergebnisse sind ebenfalls nicht deutlich. In Abbildung 2 wird dieses Verhalten verdeutlicht. Je Reihe ist eine zufällig erstellte Dirichlet-Verteilung mit $K = 5$ Dimensionen abgebildet. Jeder Kasten stellt die multinomiale Verteilung innerhalb der Dirichlet-Elemente dar. Mit $\alpha = 100$ sind alle Verteilungen ungefähr gleichverteilt und es gibt kaum Unterschiede. Bei $\alpha = 0.01$ ist das Ergebnis sehr deutlich, pro Kasten gibt es nur 1-2 Ausreißer. Es gilt also einen guten Mittelwert für α zu bestimmen. α wird oft als Hyperparameter bezeichnet.

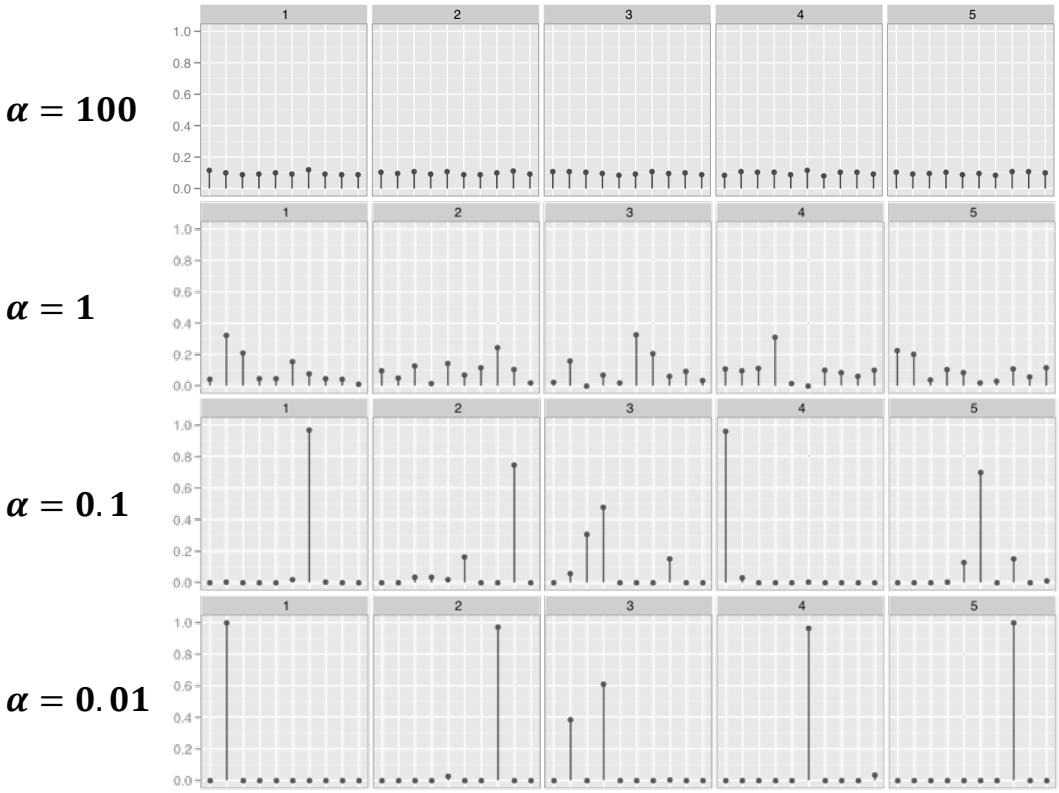


Abbildung 2 - Dirichletverteilungen $Dir(\alpha)$ für unterschiedliche alpha-Werte.[47] Jede Dirichlet-Verteilung enthält $K = 5$ multinomiale Verteilungen mit 10 Elementen, die in Summe 1 ergeben.

[47] Vgl. Blei (2011b), S. 36-42.

2.2 Beschreibung des Verfahrens

LDA nutzt das Topic Modell zum Offenlegung von inhaltlichen Strukturen von Texten anhand von Wahrscheinlichkeitsverteilungen. In Abbildung 3 ist ein praktisches Beispiel zu sehen, wie der Algorithmus vorgeht, um eine Verteilung von Topics über ein Dokument zu bilden. Auf der linken Seite sind Topics aus einem Topic Modell abgebildet, das aus mehreren Farben besteht. Jeder Kasten stellt ein Topic dar und besteht aus einer Ansammlung von Wörtern. Jedes Wort besitzt eine Wahrscheinlichkeit, einem Topic zugehörig zu sein. Alle Wörter zusammen bilden das Vokabular. Es ist zu beachten, dass ein Wort auch mehreren Topics zugeordnet werden kann, wie z.B. im Fall der Farben Violett und Cyan. Ist ein Wort im Text des auszugsweise abgebildeten Dokuments im Vokabular vorhanden, kann dieses einem Topic zugeordnet werden. Die jeweilige Zuordnung ist durch die unterschiedlichen Hintergrund-Muster dargestellt. Aus den einzelnen Zuordnungen bildet sich eine Verteilung – auf der rechten Seite abgebildet – aller Topics über das gesamte Dokument.

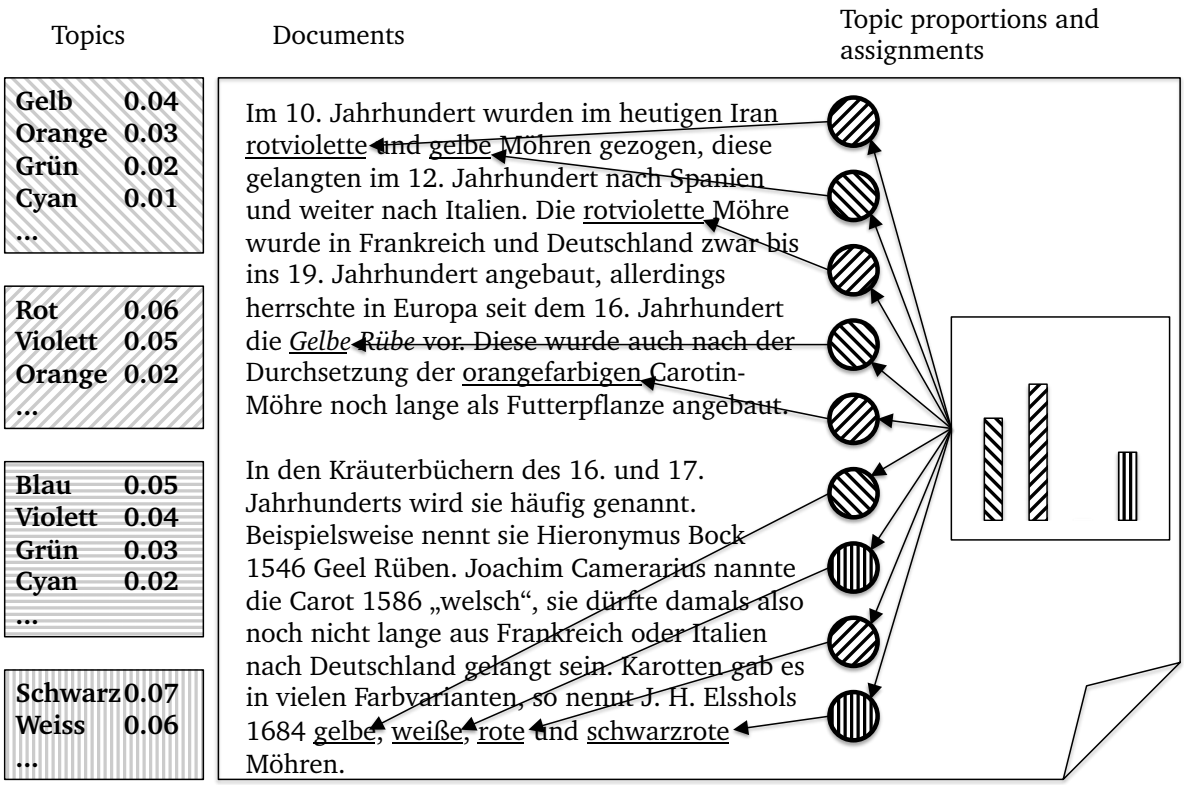


Abbildung 3 - Beispiel für eine Topic-Zuordnung einzelner Wörter in einem Dokument bei einem existierenden Topic Model. Die Abbildung ist in Anlehnung an die Darstellung von Blei[48] gestaltet. Der Text stammt von Wikipedia.[49]

[48] Vgl. Blei (2011a), S. 3.

[49] Wikipedia (o. J.): Karotten.

Bei einer Ansammlung von Texten besteht jedoch das Problem, das noch kein Topic Modell existiert und erst erstellt werden muss. Allein die Sammlung der Dokumente und die darin enthaltenen Wörter stehen für eine Verarbeitung zu Verfügung. In Abbildung 4 ist dieses Problem verdeutlicht.

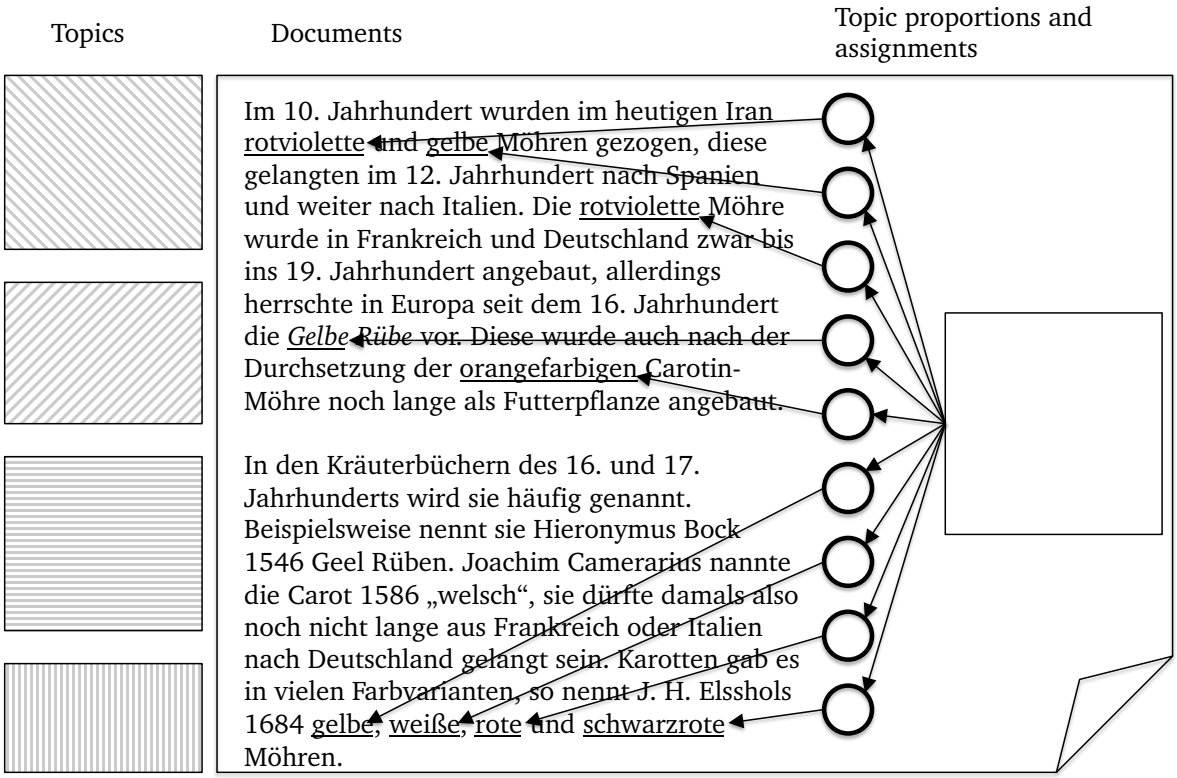


Abbildung 4 - Beispiel für die Ausgangslage vor Erstellung eines Topic Modells. Es existieren ausschließlich Wörter. Topics und Verteilungen sind noch nicht vorhanden. Die Abbildung ist in Anlehnung an die Darstellung von Blei[50] gestaltet. Der Text stammt von Wikipedia.[51]

In den folgenden Unterkapiteln soll LDA als Verfahren allgemeingültig erklärt werden. Der Fokus liegt auf einer theoretischen Abhandlung der einzelnen Schritte, wie sie in Abbildung 3 dargestellt sind. Danach werden Verfahren vorgestellt, die es ermöglichen, ein Topic Modell zu erstellen, um LDA anwenden zu können.

[50] Vgl. Blei (2011b), S. 21.

[51] Wikipedia (o. J.): Karotten.

2.2.1 Anwendung

LDA verwendet in seiner einfachsten Form ein flaches Topic Modell, das auf der Zuordnung von Themen durch Wahrscheinlichkeiten basiert.[52] Dies schafft die Möglichkeit, einzelne Wörter eines Texts einem bestimmten Thema mit einer gewissen Wahrscheinlichkeit zuzuordnen. In Abbildung 5 ist das Modell von LDA grafisch dargestellt. Die dazugehörige Legende der grafischen Elemente ist direkt neben der Abbildung zu sehen.

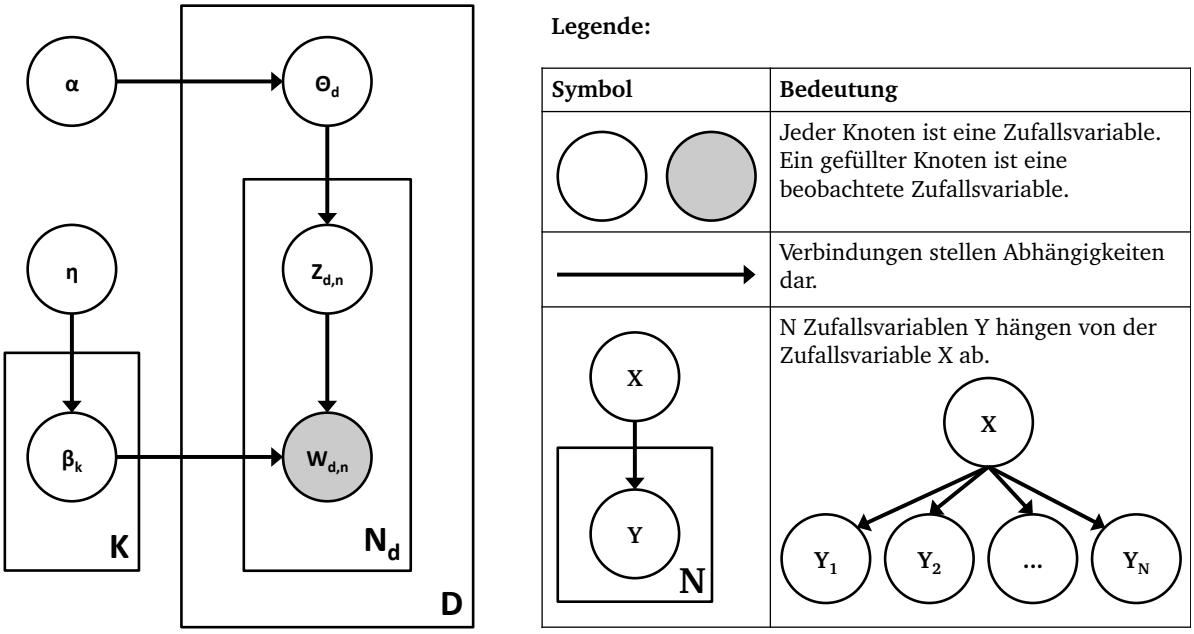


Abbildung 5 - Modell von Latent Dirichlet Allocation mit der dazugehörigen Legende[53]

[52] Vgl. Carenini et al. (2011), S. 45.

[53] Vgl. Blei; Lafferty (2009), S. 4.

Das Modell basiert auf Zufallsvariablen, dargestellt durch Knoten, die untereinander verschiedene Abhängigkeiten besitzen. Sie stellen die zugrunde liegende Themen-Struktur – die „latent topical structure“[54] – des zu durchsuchenden Datenbestandes dar. Ein Rechteck ist ein Quantor für die inneren Knoten. In der folgenden Ausführung werde ich das Modell ausgehend von β_k und θ_d hingehend zu $W_{d,n}$ beschreiben. Sie basiert auf der Vorstellung von LDA durch Blei[55]. Die Zufalls-Variablen $\vec{\alpha}$ und η werde ich im Anschluss erläutern.

Im Rahmen des LDA-Modells werden zwei verschiedene Dirichlet-Verteilungen verwendet, die im Anschluss detailliert erläutert werden:

- $\vec{\theta}_d \sim Dir(\vec{\alpha})$ - Verteilung der Themen über alle Dokumente
- $\vec{\beta}_k \sim Dir_V(\eta)$ - Verteilung des Vokabulars über alle Topics

Tabelle 1 - Beispiel für $K = 3$ Dirichlet-Verteilungen über $V = 1000$ Wörter.[56]

		Word n_1	Word n_2	...	Word n_{1000}
β^1	Topic T_1	0.001	0.21	...	0.00006
β^2	Topic T_2	0.0021	0.006	...	0.03
β^3	Topic T_3	0.0065	0.0043	...	0.009

Gegeben sei ein Korpus mit K Topics. Pro Topic gibt es ein β_k in K. Jedes β_k entspricht einer Dirichlet-Verteilung[57] $Dir_V(\eta)$ je Topic über das gesamte Vokabular V aller Dokumente D. Das Vokabular V beinhaltet alle Wörter, die in der Menge D enthalten sind. Beispiele für β und V sind tabellarisch in Tabelle 1 bzw. Tabelle 2 abgebildet.

54 Vgl. Blei; Lafferty (2009), S. 3.

55 Vgl. Blei et al. (2003), S. 995 ff.

56 Vgl. Carenini et al. (2011), S. 47.

57 Siehe Kapitel 2.1.2.

Tabelle 2 - Beispiel für ein Vokabular mit $V = 1000$ Wörtern.[58]

Word n_1	Word n_2	...	Word n_{1000}
ability	abrasion	...	youth

Alle Dokumente ergeben in Summe die Menge D und werden als Korpus bezeichnet. Innerhalb von D gibt es die Zufallsvariable $\vec{\theta}_d$ mit der Dirichlet-Verteilung $Dir(\vec{\alpha})$ je Dokument. Sie ergibt die Verteilung aller Topics pro Dokument und ist exemplarisch in Tabelle 3 abgebildet.

Tabelle 3 - Beispiel für eine Dirichlet-Verteilung über $D = 2$ Dokumente und $K = 3$ Topics.[59]

		Topic T_1	Topic T_2	Topic T_3	Σ
θ^1	Document D_1	0.1	0.7	0.2	1
θ^2	Document D_2	0.3	0.3	0.4	1

Ein Dokument d kann wiederum in weitere n Untereinheiten eingeteilt werden. Im Rahmen dieser Ausarbeitung besteht ein Dokument d aus n Wörtern. N ist damit die Menge aller Wörter in einem Dokument. Der ausgegraute Knoten $W_{d,n}$ in Abbildung 5 stellt eine Einzeleinheit – das Wort im Text – dar, während $Z_{d,n}$ für das zum Wort zugehörige Topic steht. Die Knoten für die Wörter werden farblich hervorgehoben, da es sich um die einzigen beobachtbaren Zufallsvariablen handelt. Alle anderen Variablen sind berechnet.

Die Variablen $\vec{\alpha}$ und η stellen die jeweiligen Parameter für die Dirichlet-Verteilungen dar. $\vec{\alpha}$ wird als Dirichlet-Parameter, η als Topic-Hyperparameter bezeichnet. Der Dirichlet-Parameter $\vec{\alpha}$ ist ein K-dimensionaler Vektor und beeinflusst die Form und die Ausprägung der Dokument-Topic-Verteilung $\vec{\theta}_d$. Der Topic-Hyperparameter η ist ein Skalar, der die Ausprägung der Topic-Verteilung β_k beeinflusst.[60]

58 Vgl. Carenini et al. (2011), S. 47.

59 Vgl. ebd., S. 47.

60 Siehe Kapitel 2.1.2.

Das Modell ist äquivalent zu der folgenden Text-Schreibweise, wobei Dir() eine Dirichlet und Mult() eine multinomiale Verteilung beschreibt. Das Verfahren basiert auf Blei (2009)[61]:

1. Für jedes Topic
    a. Ermittle eine Verteilung $\vec{\beta}_k \sim Dir_V(\eta)$ über das Vokabular
2. Für jedes Dokument
    a. Ermittle einen Vektor $\vec{\theta}_d \sim Dir(\vec{\alpha})$ über alle Topics
    b. Für jedes Wort
        i. Ermittle ein Topic $Z_{d,n} \sim Mult(\vec{\theta}_d), Z_{d,n} \in \{1, \dots, K\}$
        ii. Ermittle das Wort $W_{d,n} \sim Mult(\vec{\beta}_{Z_{d,n}}), W_{d,n} \in \{1, \dots, V\}$

2.2.2 Modell-Berechnung

Für die Anwendung von LDA auf einen Korpus gilt ein Topic Modell als Voraussetzung. Damit das gezeigte Modell erzeugt werden kann, werden für die Zuordnung der Wörter folgende Variablen benötigt[62]:

- $Z_{d,n}$ – Topic je Wort
- $\vec{\theta}_d$ – Topic-Verteilung je Dokument
- $\vec{\beta}_k$ – Topic-Verteilung des Korpus

Die genannten Variablen liegen zum Zeitpunkt der ersten Ausführung von LDA nicht vor und müssen berechnet werden. Dafür muss das im vorigen Kapitel vorgestellte Verfahren in umgekehrter Reihenfolge durchgeführt werden. Ausgehend von den Wörtern muss auf die einzelnen Verteilungen geschlossen werden. Aufgrund der Komplexität der Berechnungen[63] existieren diverse Näherungsverfahren[64], die nachfolgend aufgeführt, aber nicht näher vorgestellt werden:

[61] Vgl. Blei; Lafferty (2009), S. 4.

[62] Vgl. Blei (2009a), S. 21.

[63] Vgl. Blei et al. (2003), S. 1002.

[64] Vgl. Blei; Lafferty (2009), S. 7.

- Mean field variational methods[65]
- Expectation propagation[66]
- Collapsed Gibbs sampling[67]
- Collapsed variational inference[68]

Jedes Verfahren hat Vor- und Nachteile. Sie unterscheiden sich in Geschwindigkeit, Genauigkeit und Komplexität.[69] In Bezug auf die Berechnung am Computer drücken sich diese Unterschiede in Laufzeiten, der Belegung des Arbeitsspeichers und der Auslastung der Prozessoren aus.

Alle Verfahren erstellen das Topic Modell iterativ. D.h. dass zu Beginn der Berechnungen ein Modell zufallsmäßig erstellt wird. Die einzelnen Topics bestehen in diesem Zeitpunkt aus Sammlungen von Wörtern, die zufällig ausgewählt wurden. Das Modell wird je Iteration verändert und getestet, damit sich das Modell über mehrere Iterationen verbessert. Je nach Näherungsverfahren benötigt die Berechnung unterschiedlich viele Iterationen, bis gute Ergebnisse entstehen. Dies ist auch von der Größe der Anzahl der Topics und der Wörter abhängig.

2.3 Parameter und Ergebnisse

Nach der Beschreibung des Verfahrens werden die nötigen Parameter für die Berechnung des LDA-Modells detaillierter erklärt. Danach folgt die Interpretation der Ergebnisse, die von LDA erzeugt werden.

2.3.1 Beeinflussbare Parameter

Zur Optimierung des Modells können verschiedene Parameter die Berechnungen der Verfahren beeinflussen:

---

65 Blei et al. (2003).

66 Minka; Lafferty (2002).

67 Griffiths and Steyvers (2006).

68 Teh et al. (2006).

69 Vgl. Blei; Lafferty (2009), S. 8.

- α als Eingangsparameter (termsmoothing)
- η als Eingangsparameter (topicsmoothing)
- Anzahl an Topics
- Anzahl an Iterationen
- Größe des Vokabulars

Die Parameter $\boldsymbol{\alpha}$ **und** $\boldsymbol{\eta}$ beeinflussen die gesamte Berechnung. Sie sind die Hyperparameter für die jeweiligen Dirichlet-Verteilungen[70] und beeinflussen die Ergebnisse.[71] Es gilt, für LDA einen guten Mittelwert für α und η zu ermitteln, damit die Ergebnisse verwertbar sind. In verschiedenen Ausarbeitungen werden unterschiedliche Werte zur Berechnung verwendet, z.B. $\alpha = \eta = 0{,}1$[72], $\alpha = \eta = 0{,}01$[73] und $\alpha = \frac{50}{K}\ und\ \eta = 0{,}1$[74].

Zur Berechnung der **Anzahl an Topics** gibt es kein direktes Berechnungs-Verfahren. Mit der Perplexität eines Modells existiert jedoch ein Maß, mit dem die Qualität unmittelbar nach der Berechnung eines Modells bewertet werden kann. Im Optimum kann ein Minimum der Perplexitätskurve über unterschiedliche Topic-Anzahlen ermittelt werden. Durch die oft begrenzte Rechenleistung kann das Minimum oft nicht errechnet werden. Es werden jedoch bereits gute Ergebnisse erzielt, sobald die Kurve offensichtlich konvergiert.[75]

Wie die Topic-Anzahl kann auch die optimale **Anzahl an Iterationen** nicht vor der Berechnung ermittelt werden. Während der Laufzeit kann jedoch beurteilt werden, ob die bereits berechneten Iterationen gute Ergebnisse liefern. So kann für jede Iteration ein Log-Likelihood-Schätzer berechnet werden. Dieser Schätzer ist negativ und soll maximiert werden. Sobald der Wert konvergiert, sind keine weiteren Iterationen nötig und das Modell ist ausreichend berechnet.[76]

---

70 Siehe Kapitel 2.1.2.

71 Siehe Abbildung 2.

72 Vgl. The Stanford Natural Language Processing Group (2011).

73 Vgl. Ramage et al. (2010), S. 32.

74 Vgl. Griffiths; Steyvers (2004), S. 5231.

75 Vgl. Blei et al. (2003), S. 1008.

76 Vgl. Griffiths; Steyvers (2004), S. 5230.

Das **Vokabular** stellt die Gesamtmenge an Wörtern dar. Bei einer Reduzierung dieser Menge kann die Berechnung beschleunigt und die Qualität der Ergebnisse verbessert werden. Dies kann beispielsweise durch Herausfiltern der am häufigsten vorkommenden Wörtern, dem Entfernen sogenannter Stopp-Wörter [77](im Deutschen z.B. „der“, „die“, „das“) oder durch Reduzierung der Wörter auf ihren Wortstamm[78] - sogenanntes „Stemming“ - geschehen. Allgemein gibt es noch weitere Verfahren, um Wörter ohne Relevanz für die Ergebnisse zu identifizieren und zu entfernen. Einige dieser Verfahren werden in Kapitel 3.2.4 gezeigt. Eine detaillierte Ausführung von Text-Vorverarbeitung ist in Agarwal[79] zu finden.

Die Anwendung von Stemming auf ein Vokabular soll kurz ausgeführt werden, da es die Daten erheblich beeinflusst. „Stem“ ist englisch für Wortstamm und bezeichnet das Kürzen unterschiedlicher Wortformen auf ihren Wortstamm.[80] Beispielsweise würden die Wörter „Sitz“, „sitzen“, „Sitze“ auf den Stamm „Sitz“ reduziert werden. Die Topics eines Modells verändern sich durch Stemming inhaltlich oft nicht. Die zusammengehörenden Wörter werden trotzdem dem gleichen Topic zugeordnet.[81] Es ist jedoch hilfreich zur Reduzierung des Vokabulars, um den bei der Berechnung benötigten Speicher zu entlasten.[82] Dadurch enthalten die Topics weniger Wörter und sind übersichtlicher. Ein üblicher Algorithmus zum Stemming ist der Porter Stemmer[83].

2.3.2 Qualitätsmaße und Ergebnisse des Verfahrens

Neben den verschiedenen Verteilungen[84], die das Modell nach der Berechnung bietet, existieren weitere Maße, die zur Beurteilung der Qualität des Modells oder des Korpus dienen:

- Perplexität

---

77 Vgl. Baeza-Yates; Ribeiro-Neto (1999), S. 167 f.

78 Vgl. ebd., S. 168 f.

79 Vgl. Agarwal; Liu (2009), S. 59 f.

80 Vgl. ebd., S. 59.

81 Vgl. The Stanford Natural Language Processing Group (2011).

82 Vgl. Ramage et al. (2011), S. 6.

83 Vgl. Agarwal; Liu (2009), S. 59.

84 Siehe Tabelle 1, Tabelle 2 und Tabelle 3.

- Relevanz
- Log-Werte
- Ähnlichkeit von Dokumenten

Die **Perplexität** ist eine allgemeine Maßgröße in der Stochastik, mit der die Qualität von Modellen von Wahrscheinlichkeitsverteilungen bestimmt werden kann[85]. In Bezug auf LDA beschreibt dieses Maß, wie überraschend neue Wörter für das bereits bestehende Modell sind. Perplexität kann genutzt werden, um eine optimale Topic-Anzahl für ein Modell zu bestimmen. Zur Berechnung kann z.B. die erste Hälfte eines jeden Dokuments in das Modell einbezogen werden, um das Modell danach auf die zweite Hälfte anzuwenden. Je niedriger die Perplexität ist, desto weniger haben die Wörter in der zweiten Hälfte überrascht und das Ergebnis ist besser. Das Ziel ist eine möglichst minimale Perplexität. Da das Minimum durch begrenzte Rechenleistung unter Umständen nicht erreicht werden kann, darf auch eine Topic-Anzahl in einem Bereich gewählt werden, in der die Perplexitätskurve offensichtlich gegen ein unbekanntes Minimum konvergiert.[86] Unter anderem hängt die Perplexität von der Größe des Vokabulars ab und kann daher nur innerhalb eines Korpus zum Vergleich verschiedener Topic-Anzahlen genutzt werden.

Topics können aufgrund der summierten Häufigkeit ihrer zugeordneten Wörter als mehr oder weniger **relevant** für weitere Auswertungen bewertet werden. Die Dirichlet-Verteilung $\vec{\beta}_k \sim Dir_V(\eta)$ beinhaltet alle Topics und ihre zugeordneten Wörter. Jedes Wort hat dabei eine Häufigkeit im gesamten Korpus. Diese Häufigkeit weist in Summe auf eine Kohärenz jedes einzelnen Topics hin. Beispielsweise können so Topics ausgeordnet werden, die eine im Vergleich zu anderen Topics hohe bzw. niedrige kumulierte Häufigkeit besitzen. Diese beinhalten meist oft bzw. selten vorkommende Wörter, die wenig Bezug zu anderen Topics aufweisen. Diese Häufigkeit wird im Rahmen dieser Thesis als **Relevanz** von Topics definiert. Es handelt sich dabei um einen Dezimalwert, da ein Wort mehreren Topics zugeordnet sein kann.

Unabhängig vom gewählten Näherungsverfahren kann je Iteration ein **Log-Schätzer** ausgegeben werden. Damit kann die Qualität des Modells bereits während der Berechnung

---

85 Vgl. Wikipedia (o. J.): Perplexität (Statistik).

86 Vgl. Blei et al. (2003), S. 1008.

bestimmt werden. Pro Iteration verbessert sich der Schätzer. Es handelt sich um einen Log-Likelihood-Schätzer, der negativ ist und maximiert werden soll. Sobald die Werte deutlich konvergieren, kann auf weitere Iterationen verzichtet werden. Die Topics sind damit näherungsweise optimal berechnet.

LDA kann auch genutzt werden, um **ähnliche Dokumente** zu identifizieren. Blei[87] hat die Formel dafür eine einfache Formel definiert:

$$document - similarity_{d,f} = \sum_{k=1}^{K} \left(\sqrt{\hat{\theta}_{d,k}} - \sqrt{\hat{\theta}_{f,k}} \right)^2$$

Dabei gilt es für die beiden Dokumente d und f die Differenz aus der Wurzeln der jeweiligen Wahrscheinlichkeitswerte je Topic zu quadrieren und über alle Topics zu summieren. Das Ergebnis ist eine Distanz zwischen zwei Dokumenten. Je kleiner die Distanz ist, desto ähnlicher sind sich die Dokumente in ihren Topics.

2.4 Weiterführende Informationen

Nach der ersten Vorstellung LDA im Jahr 2003[88] sind bereits speziellere Verfahren entwickelt worden, die unterschiedlichen Zielsetzungen dienen. Neben den theoretischen Modellen wurde LDA mit seinen unterschiedlichen Varianten bereits für mehrere Programmier-Plattformen implementiert, so dass bei populären Sprachen wie z.B. Java auf bestehende Programme aufgesetzt werden kann. Des Weiteren sind inzwischen zum Einstieg in das Thema LDA unterschiedliche Vorlesungen, Publikationen und Tutorien verfügbar. Die bereits genannten Informationen sollen in den nächsten Unterkapiteln genauer ausgeführt werden, um die Entwicklung von LDA seit 2003 exemplarisch abzubilden.

2.4.1 Weitere Abwandlungen

In diesem Kapitel wurde die erste veröffentlichte Fassung von LDA aus dem Jahr 2003 ausgeführt. Inzwischen haben Wissenschaftler weitere Variationen entwickelt, die für

[87] Vgl. Blei; Lafferty (2009), S. 6.

[88] Blei et al. (2003).

bestimmte Anwendungsszenarien zugeschnitten sind. Die folgende Auflistung basiert auf den verlinkten Publikationen der Princeton University zum Thema LDA.[89]

Supervised LDA (sLDA)[90] kann bereits vorhandene Meta-Informationen von Dokumenten in die Berechnung integrieren. Dies können z.B. mit Sternchen versehene Produktbewertungen, Kategorien etc. für ein Dokument sein.

Labeled LDA (LLDA)[91] ist ein erweitertes sLDA. Hier wird das Modell um Labels für Dokumente erweitert. Labels können z.B. tags oder Stichwörter sein, die für ein Dokument bereits vergeben wurden. Ein Topic wird einem Label zugeordnet. LLDA hat den Vorteil, dass gegenüber normalem LDA ähnliche Topics eher automatisch zusammengefasst werden, da durch die Labels je Dokument bereits eine erste thematische Einordnung stattgefunden hat.

Partially Labeled LDA (PLDA)[92] ist ähnlich zu LLDA. Im Gegensatz zu LLDA sind Labels für jedes Dokument nicht zwingend notwendig.

Hierarchical LDA (hLDA)[93] generiert die Topics in einer hierarchischen Anordnung.

Yahoo! LDA (Y!LDA)[94] verwendet das Ursprungs-LDA-Modell (in diesem Kapitel beschrieben), ist aber auf eine hohe Skalierbarkeit und Effizienz ausgerichtet.

Bei **Discriminatively trained LDA** (DiscLDA)[95] wird ein abgewandeltes Berechnungsmodell angewandt, dass mindestens so gute Ergebnisse wie klassisches LDA liefern soll.

Relational supervised Latent Dirichlet Allocation (rsLDA)[96] ist ein erweitertes sLDA. Es kombiniert ein hierarchisches Bayes-Modell mit einem relationalen probabilistischem Lernansatz zur Verbesserung der Ergebnisse.

---

89 Mimno (2012).

90 Vgl. Blei (2011a). S. 11.

91 Vgl. Ramage et a. (2010), S. 132.

92 The Stanford Natural Language Processing Group (2011).

93 Blei (2012).

94 Narayanamurthy (2011).

95 Lacoste-Julien et al. (2008).

96 Taranto et al. (2011).

2.4.2 Implementierungen von LDA

Die Näherungsverfahren von LDA sind bereits in vielen Programmiersprachen implementiert worden. Die folgende Auflistung beinhaltet die in den veröffentlichten Ausarbeitungen beschriebenen Implementierungen, die von Princeton zum Thema LDA zusammengestellt wurden[97]:

- CUDA[98] (Berechnung auf „Graphical Processing Units“ (GPUs))
- C [99] [100] [101]/ C++[102]
- R[103]
- Matlab[104]
- Hadoop[105]
- Scala (Stanford TMT)[106]
- Java (Mallet[107])
- Python[108]

Für die praktische Anwendung von LDA wurden die Implementierung für R und Scala getestet. Die im Rahmen dieser Thesis zu entwickelnde Software soll möglichst einfach zu bedienen zu sein und leicht interpretierbare Ergebnisse liefern. Eine Implementierung auf Basis von Scala wird von der Stanford University entwickelt und „Topic Modeling Toolbox“ (TMT) genannt. Die Software hat trotz der niedrigen Versionsnummer 0.4.0 viele Vorteile wie

---

97 Mimno (2012).

98 Yan et al. (2009).

99 Blei (2006).

100 Nallapati (2012).

101 Buntine (2012).

102 Phan; Nguyen (2008).

103 Chang (2012).

104 Steyvers; Griffiths (2011).

105 Narayanamurthy (2011).

106 The Stanford Natural Language Processing Group (2011).

107 McCallum (2011).

108 Řehůřek (2012).

Multithreading, eine ausführliche Dokumentation, integrierte NLP[109]-Funktionen (z.B. Stemming), Excel-Export und ist auf einer „Java Virtual Machine“ (JVM) lauffähig. Scala und Java können direkt miteinander interagieren. Die Näherungsverfahren „Collapsed Gibbs sampling“ und „Collapsed variational inference“ sind bereits integriert. Die Software TMT steht unter der GPLv2-Lizenz[110] und wird im Rahmen dieser Thesis zur Berechnung von LDA genutzt.

2.4.3 Weiterführende Literatur / führende Quellen

Die Grundlagen von LDA werden von Blei in mehreren Publikationen ausgeführt. So sind in der Erstveröffentlichung von LDA „Latent Dirichlet Allocation“[111] mathematische Grundlagen des Verfahrens festgehalten. „Topic Models“[112] ist eine praktischer angelegte Ausarbeitung, die sich auf die Erklärung der zugrunde liegenden Modelle, der Interpretation der Ergebnisse und der Diskussion der zu verwendenden Parameter konzentriert. Zu dieser Ausarbeitung wurde eine im Internet verfügbare Vorlesung[113] und die Folien[114] veröffentlicht. 2011 hat Blei ein Tutorial für den Einstieg in das Thema LDA und Topic Models veröffentlicht[115].

In diesem Kapitel wurde die Funktionsweise von LDA erklärt. Das zugrundeliegende Topic Modell und die Dirichlet-Verteilung wurden soweit ausgeführt, wie es für ein Verständnis im Rahmen dieser Thesis nötig ist. LDA baut auf unterschiedlichen Wahrscheinlichkeitsverteilungen auf und ist ein geeignetes Verfahren, um Texte auf ihre thematischen Zusammenhänge zu untersuchen. Da LDA-Modelle sehr aufwändig zu berechnen sind, gibt es verschiedene Approximationsverfahren. Des Weiteren wurde gezeigt, dass es bereits mehrere Abwandlungen von LDA für bestimmte Anwendungszwecke gibt und diese bereits für zahlreiche Plattformen und Programmiersprachen implementiert wurden. Für

---

109 Natural Language Processing.

110 Free Software Foundation (1991).

111 Blei et al. (2003).

112 Blei; Lafferty (2009).

113 Blei (2009b).

114 Ders. (2009a).

115 Ders. (2011b).

einen detaillierten Einstieg in LDA gibt es bereits empfehlenswerte Vorlesungsfolien, Video-Vorlesungen und wissenschaftliche Ausarbeitungen.

3 Daten und Methode

Für eine Anwendung von LDA ist der Korpus der zu analysierenden Dokumente zu definieren. Im Rahmen dieser Thesis sind der thematischen Auswahl Grenzen gesetzt, jedoch sind weitere Kriterien notwendig. Im Bereich von wissenschaftlichen Ausarbeitungen gibt es weitere Gesichtspunkte, die z.B. die Qualität und Quantität von Dokumenten betreffen. Diese sollen klar definiert sein, um eine gute Auswahl zu treffen.

Die für eine Verarbeitung der ausgewählten Daten notwendigen Prozesse werden die Grundlage für die zu implementierende Software darstellen. An dieser Stelle sind ebenfalls Anforderungen und Kriterien zu berücksichtigen, um die Datenqualität über mehrere Verarbeitungsschritte nicht zu vermindern. Im Bereich der Datenverarbeitung können vielseitige Probleme auftauchen, deren Lösungen mit in die Prozesse einfließen sollen.

Die zu implementierende Software bildet das Kernstück der Verarbeitung, integriert die definierter Prozesse und erstellt aus den definierten Daten Ergebnisse, die für weiterführende Analyse verwertet werden können. Der Fokus der Beschreibung liegt auf der Funktionsweise und der Bedienung.

Im folgenden Kapitel werden die Daten ausgewählt und in ihrer Struktur analysiert. Darauf aufbauend werden die einzelnen Prozesse für die Datenverarbeitung definiert. Die Software stellt das letzte Unterkapitel dar und implementiert die definierten Prozesse.

3.1 Daten

LDA ist bezüglich des Daten-Inputs universell einsetzbar. Mit dem Algorithmus lassen sich nicht nur Text-Mengen analysieren, sondern z.B. auch Bilder und andere digitalen Inhalte, die Gemeinsamkeiten aufweisen können wie z.B. die Ähnlichkeiten von Bildern.[116]

Anhand von LDA sollen wissenschaftliche Ausarbeitungen analysiert werden, damit diese einem oder mehreren Themen als relevant zugeordnet werden können. Neben der Einordnung der Publikationen soll zusätzlich die Erstellung einer Zeitreihe und die Identifikation ähnlicher Dokumente möglich sein.

[116] Vgl. Griffiths; Steyvers (2004), S. 5230 f.

Nach der Auswahl der Daten werden diese quantitativ vorgestellt und Chancen zu einer qualitativen Datenverarbeitung diskutiert. So sollen die Inhalte der Konferenzen sinnvoll verarbeitet werden, um gute Ergebnisse zu erhalten.

3.1.1 Kriterien

Das Ziel dieser Thesis ist die Untersuchung von wissenschaftlichen Publikationen in der Software-Industrie zur Identifikation von Trends und häufig diskutierten Problemstellungen. Für eine sinnvolle Untersuchung wird ein Datenbestand benötigt, der Entwicklungen in der Software-Industrie beschreibt und eine ausreichende Größe besitzt. Für den Datenbestand sollen nach eigener Definition folgende Voraussetzungen gelten:

- Quantitativ großer Datenbestand
- Wissenschaftliche Publikationen
- Repräsentatives Abbild der Entwicklung in der Software-Industrie

Durch einen **großen Datenbestand** von mehreren tausend Dokumenten ist die Wahrscheinlichkeit höher, dass die Relevanz der Daten steigt und weniger Fehler entstehen. Beispielsweise könnten viele Dokumente über gehypte Themen im Datenbestand enthalten sein. Je mehr Dokumente vorhanden sind, desto weniger Gewicht erhalten thematische Ausreißer.

Die Konzentration auf **wissenschaftliche Publikationen** soll die Qualität der Auswertung sicherstellen, damit sich ein **repräsentatives Abbild der Entwicklung in der Software-Industrie** ergibt. Über einen definierten Zeitraum sollen Themen identifiziert werden können, denen sich Wissenschaftler mehr oder weniger gewidmet haben.

Für die Suche nach geeigneter Literatur hat vom Brocke[117] Möglichkeiten zur Identifikation von hoch-qualitativen Publikationen und deren Eignung untersucht. Bei der Suche darf die Genauigkeit nicht leiden. Die kann im Einzelnen durch die Nachvollziehbarkeit der Suche und eine Aussagekraft der Literaturgrundlage sichergestellt werden. Die Aussagekraft soll durch Entscheidungen wie z.B. die Wahl der Literaturquellen und eines Zeitraumes bewahrt

[117] Vom Brocke et al. (2009).

werden.[118] Diese Voraussetzungen sollen auch für die Wahl der Datengrundlage dieser Thesis gelten.

3.1.2 Auswahl der Daten

Für eine Auswertung der Daten sollen wissenschaftliche Publikationen genutzt werden. Dafür ziehe ich Beiträge von Journalen und Konferenzen in Betracht. Bei Konferenzen auf den Seiten der AISeL[119] gibt es mehr Beiträge, da diese quantitativ in höherer Zahl frei – aus dem Netzwerk der Technischen Universität Darmstadt – verfügbar sind. Beiträge von Journalen sind quantitativ weniger vorhanden, aber qualitativ hochwertiger, da strengere Kriterien für die Auswahl von Publikationen gelten. Anhand von Rankings lassen sich qualitativ hochwertige Journale auswählen.[120]

LDA soll dazu dienen, große Datenbestände analysieren zu können.[121] Dadurch lassen sich realistischere Zeitreihen in Bezug auf Entwicklungen in der Wirtschaftsinformatik erstellen. Da auch die zeitliche Entwicklung von Trends im Fokus dieser Thesis steht, ist mit Konferenz-Beiträgen ein besseres Ergebnis zu erwarten. Die Datengrundlage wird sich daher auf wissenschaftlichen Publikationen von Konferenzen konzentrieren.

Im Bereich der Wirtschaftsinformatik gibt es viele Konferenzen, die für die Datengrundlage in Frage kommen. Die wichtigsten Konferenzen sind als Mitglieder in der „Association of Information Systems“ (AIS) organisiert und sollen detaillierter untersucht werden. Die Publikationen sind in der AIS Electronic Library (AISeL)[122] abgelegt und können aus dem Netzwerk der TU Darmstadt frei heruntergeladen werden. Konkret handelt es sich um folgende Konferenzen:

- Americas Conference on Information Systems (AMCIS)
- International Conference on Information Systems (ICIS)
- European Conference on Information Systems (ECIS)

---

118 Vgl. vom Brocke et al. (2009), S. 4.

119 Association for Information Systems (o. J.).

120 Vgl. vom Brocke et al. (2009), S. 4.

121 Vgl. Blei; Lafferty (2009), S. 1.

122 Vgl. Association for Information Systems (o. J.).

- International Conference on Information Resources Management (CONF-IRM)
- Mediterranean Conference on Information Systems (MCIS)
- Pacific Asia Conference on Information Systems (PACIS)

Als weitere Einschränkung soll die Konzentration auf gleichsprachige Artikel gelten. LDA kann grundsätzlich unabhängig von der jeweiligen Sprache eingesetzt werden. Der Algorithmus benötigt jedoch Gemeinsamkeiten zwischen verschiedenen Dokumenten, damit die Wahrscheinlichkeit der zu einem Thema zugehörigen Wörtern gebildet werden kann.[123] Gerade im Bereich von Information Systems ist es durch die Verwendung von vielen englisch-sprachigen Begriffen in nicht-englischen Texten denkbar, dass auch bei unterschiedlichen Sprachen der Artikel Gemeinsamkeiten gefunden werden. Die Zusammenhänge wären jedoch nicht so hoch wie bei rein gleichsprachigen Artikeln und könnten die Ergebnisse insgesamt verschlechtern. Einzelne Themen wären nicht mehr eindeutig abzugrenzen.

Ein Großteil der Publikationen der ausgewählten Konferenzen ist in englischer Sprache verfasst. Daher werden die Daten auf englisch-sprachige Artikel eingeschränkt.

Tabelle 4 - Dokumente je Sprache und Konferenz – es ist zu beachten, dass diese Unterscheidung durch die Sprachanalyse erst vor der Durchführung von LDA vorgenommen werden kann. Die Summen unterscheiden sich aufgrund von fehlerhaften Dateien von den online verfügbaren Dateien.

Sprache	CONF-IRM	MCIS	ECIS	AMCIS	ICIS	PACIS	Summe
Englisch	194	357	2251	6821	2238	1653	**13520**
Spanisch	5			61		1	**67**
Französisch		1				21	**22**
Italienisch		1	1	2		35	**39**
Portugiesisch	10			18			**28**
Summe	**209**	**359**	**2252**	**6902**	**2238**	**1716**	**13676**

Als technische Voraussetzung sollten die Daten in einem gemeinsamen Datei-Format vorliegen, das für eine automatische Verarbeitung geeignet ist. In der Wissenschaft hat sich

[123] Vgl. Ramage et al. (2010), S. 135.

dabei das PDF-Format als Quasi-Standard für Veröffentlichungen durchgesetzt. Damit werden nur Dateien in die Auswertung einbezogen, die im PDF-Format vorliegen.

3.1.3 Struktur der Daten

Vor dem Herunterladen der Daten sollen die auf den Internetseiten der AIS zur Verfügung stehenden Dokumente genauer betrachtet werden. Zur Unterstützung dieser Aufgabe wird ein Crawler[124] genutzt, um alle Links zu PDF-Dateien zunächst zu extrahieren. Um zu einer Seite zu gelangen, die PDF-Links enthält, wird auf der AIS-Seite zunächst die Konferenz, gefolgt von dem Jahr, ausgewählt. Es öffnet sich eine Seite, auf der PDF-Dateien verlinkt sind. Beträgt die Anzahl an Publikationen mehr als 100, so werden mehrere Seiten zur Anzeige verwendet. Die PDF-Links je Seite werden in Microsoft Excel für eine Auswertung gesammelt. Eine Pivot-Auswertung der erhobenen Daten ist in Tabelle 5 zu sehen. Die Links sind jeweils nach Jahr und Konferenz aufgeschlüsselt. Insgesamt sind 13.689 PDF-Dokumente online verfügbar. Allein 50 % der Dokumente stammen von der Konferenz AMCIS.

[124] Software zur automatisieren Analyse von Internetseiten auf bestimmte Informationen. Vgl. Wikipedia (o. J.): Webcrawler.

Tabelle 5 - Quantitative Übersicht der Daten nach Konferenz und Jahr auf der AISeL-Seite[125]

Jahr	CONF-IRM	MCIS	ECIS	AMCIS	ICIS	PACIS	Summe
1993						67	**67**
1994					74		**74**
1995					44	100	**144**
1996					86		**86**
1997					80	84	**164**
1998				428	64		**492**
1999				358	80		**438**
2000			200	437	94	82	**813**
2001			127	440	80	84	**731**
2002			167	360	93		**620**
2003			101	467	106	115	**789**
2004			175	577	89	143	**984**
2005			161	518	85	111	**875**
2006			210	549	123	115	**997**
2007		23	194	511	158	154	**1040**
2008	63	48	250	407	212	138	**1118**
2009	53	135	225	836	205	117	**1571**
2010	59	89	167	557	268	191	**1331**
2011	34	64	275	465	299	218	**1355**
Summe	**209**	**359**	**2252**	**6910**	**2240**	**1719**	**13689**

Abbildung 6 und Abbildung 7 verdeutlichen die in der Einführung erwähnte steigende Datenmenge. Gegenüber 2001 hat die Anzahl an Veröffentlichungen im Jahr 2011 innerhalb von 10 Jahren um 85 % zugenommen. In Abbildung 6 ist die kumulierte Anzahl an Publikationen im Jahresverlauf abgebildet. Wie zu sehen ist, steigt die Menge nicht kontinuierlich. In den Jahren 1999, 2001, 2002, 2005 und 2010 sinkt die Anzahl an Veröffentlichungen. Bei einer Betrachtung der Gesamtmenge wird der Anstieg der Veröffentlichungen jedoch deutlich. Abbildung 7 stützt diese These und zeigt das Wachstum bzw. die Schrumpfung der Dokumenten-Anzahl im Vergleich zum Vorjahr. Auch hier muss differenziert betrachtet werden. Da die Dokumente der AMCIS jedes Jahr einen großen Anteil darstellen, wirken sich Veränderungen an Veröffentlichungen dieser Konferenz auf die Gesamtmenge stark aus. Alle anderen Konferenzen verzeichnen auch 2010 Zuwächse.

[125] Association for Information Systems (o. J.).

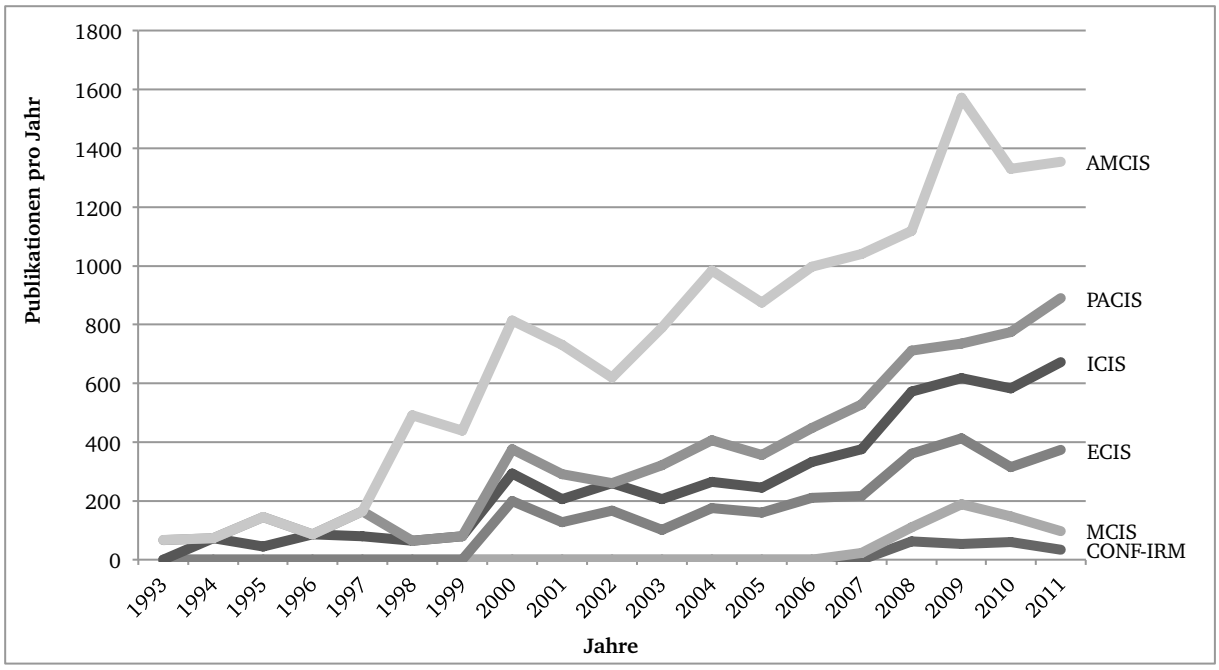


Abbildung 6 - Wachstum der kumulierten Menge an Veröffentlichungen von AIS-Konferenzen von 1993 bis 2011

Zur Extraktion der PDF-Links wird „DownThemAll“[126] als Crawler genutzt. Dieses ist ein Plugin für den Browser „Mozilla Firefox“[127] und durchsucht Internetseiten nach verlinkten Datei-Typen. Nachdem eine Seite geöffnet wird, können Dateitypen sofort heruntergeladen werden. Die Links lassen sich in eine Text-Datei exportieren. Der Aufbau der Links lässt Rückschlüsse auf die Konferenz und das entsprechende Jahr zu[128], so dass Auswertungen über Konferenzen und Jahre möglich ist.

126 Maier et al. (2012).

127 Mozilla Corporation (2012).

128 Beispiel: Die Datei aus dem folgenden Link ist der Konferenz AMCIS im Jahr 1998 zuzuordnen: http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1464&context=amcis1998

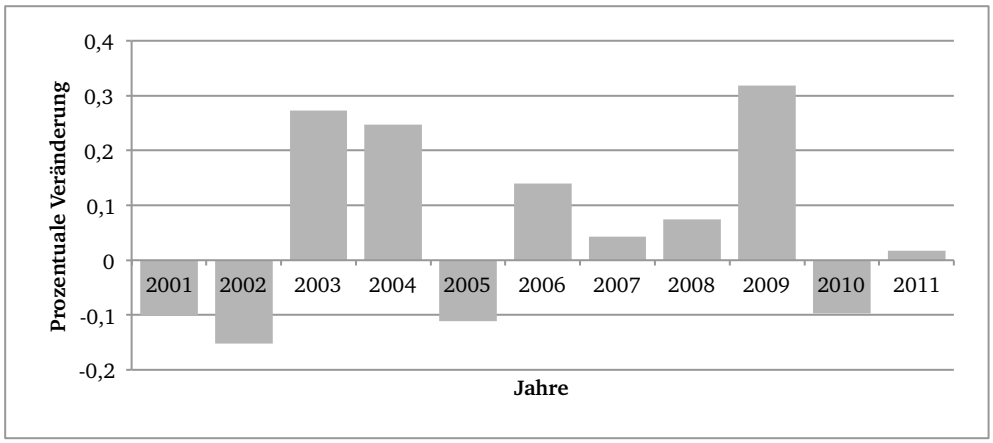


Abbildung 7 - Datenwachstum der letzten 10 Jahre im Vergleich zum jeweiligen Vorjahr von 2001 bis 2011

3.1.4 Qualität der Daten

Vor der Definition der Prozesse ist eine weitere Definition nötig, wie die Daten verarbeitet und analysiert werden sollen. Die definierte Datenmenge besteht aus wissenschaftlichen Publikationen, die aus Text bestehen. Diese Texte sind teilweise unterschiedlich strukturiert, weisen aber oft gemeinsame Merkmale auf, z.B. einen Titel, eine Zusammenfassung, mehrere Stichwörter, Inhalt und ein Literaturverzeichnis.

Eine Identifikation von Trends könnte auf der Analyse des Titels, der Stichwörter, der Verweise und des Inhalts basieren. Sowohl Titel als auch Stichwörter sind nicht unbedingt eindeutig für die Identifizierung von Themen. Es gibt bei Stichwörtern unter anderem viele Synonyme, die nicht vollständig aufgeführt werden. Titel können kurz und lang sein und können nicht treffend für den eigentlichen Inhalt vergeben sein. Die Datenmenge ohne Inhalt würde außerdem für eine Analyse durch LDA nicht ausreichend sein.

Durch die Verwendung von Verweisen könnte ein Netz für die Beziehungen der Publikationen erstellt werden. Dies gibt Hinweise auf oft zitierte Arbeiten, die detaillierter betrachtet werden sollten. Jedoch kann nicht davon ausgegangen werden, dass alle aufgeführten Quellen von Publikationen in einem Datenbestand vorliegen. Des Weiteren ist LDA für solche Zwecke nicht ausgelegt.

Der Inhalt hat die meiste Aussagekraft über eine Publikation, auf der Gegenseite aber die höchste Komplexität für eine automatische Analyse. Der Inhalt repräsentiert eine große Datenmenge und stellt für LDA daher einen guten Dateninput dar.

Die Extraktion des Inhalts stellt eine komplexe Aufgabe an das Verfahren, da die genannten anderen Elemente wie der Titel identifiziert und entfernt werden müssten. Da LDA jedoch universell gegenüber dem Dateninput ist[129], kann der gesamte Text mit seinen Merkmalen verarbeitet werden.

Ein Nachteil dieses Verfahrens ist die Verarbeitung von Titelseiten, Kopf- und Fußzeilen. Merkmale, die oft verwendet werden und für Topics mit starker Kohärenz sorgen[130]. Topics mit einem starken Zusammenhang können jedoch einfach identifiziert und daher entfernt werden. So ist eine manuelle Überarbeitung nötig, die in einem vertretbaren Zeitaufwand durchgeführt werden kann. Die Wörter könnten außerdem als Stoppwörter[131] erfasst werden.

Die Vorteile einer Verarbeitung des kompletten Texts sind einfacher gestaltete Prozesse und die Verarbeitung von Titel und Zusammenfassung. Diese Merkmale wiederholen die wichtigsten Wörter und sorgen für eine stärkere Kohärenz der Wörter in einem Topic. Je einfacher die Prozesse gestaltet sind, desto weniger Fehler können auftreten. Für die Prozesse wird also der gesamte Text einer Datei verarbeitet.

3.2 Prozesse

Die Prozesse bilden die Grundlage für die zu implementierende Software und sind damit die Vorlage für die grafische Oberfläche und den dahinter stehenden Programmier-Code. Es gibt vier Hauptprozesse, die sequentiell aufeinander folgen. Sie sind in Abbildung 8 dargestellt. Nach dem Anlegen einer Domäne (1) wird die Datengrundlage für weitere Berechnungen (2) erstellt. Wahlweise durch automatische Herunterladen oder manuelles Hinzufügen von PDF-Dateien. Die weitere Datenaufbereitung (3) stellt sicher, dass die zur Verfügung stehenden Daten von TMT verarbeitet werden (4) können.

[129] Vgl. Griffiths; Steyvers (2004), S. 5230.

[130] Siehe Kapitel 2.3.2.

[131] Siehe Kapitel 2.3.1.

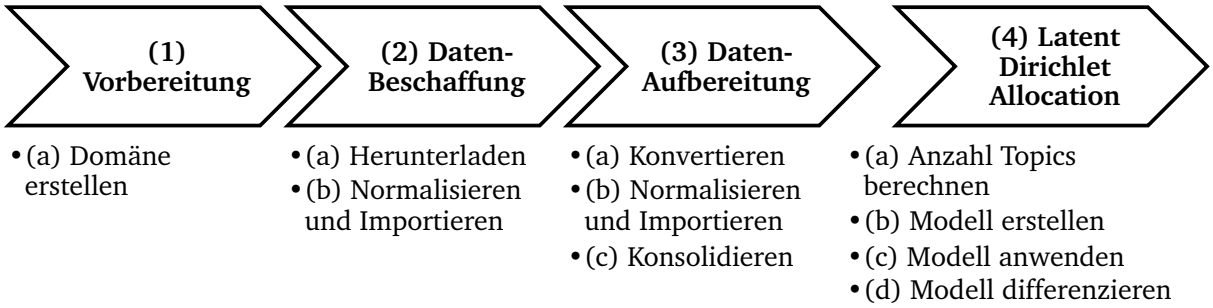


Abbildung 8 - Gesamtprozess zur Datenanalyse, unterteilt in 4 Hauptprozesse

3.2.1 Prozess (1) - Vorbereitung

Zur Speicherung aller Dateien ist ein lokaler Speicherort aller Dateien notwendig. Alle an den Prozessen beteiligten Informationen werden so an einem Ort gehalten, um dem Datenfluss nachvollziehen zu können. Aufgrund von Erfahrungen bei ersten Auswertungen ist es nötig, Dateien auch nachträglich in den Datenbestand aufnehmen zu können. Dieser Speicherort wird als **Domäne** definiert und soll die eben genannten Voraussetzungen erfüllen.

Alle zu Auswertungen zugehörigen Dateien sollen in der Domäne gespeichert werden, einschließlich individueller Einstellungen der Software. Die Speicherung der Dateien ist nach verschiedenen Prozessschritten aufgeteilt damit nachvollziehbar ist, woher eine Datei kommt und ob sie im Laufe der Auswertung bearbeitet wurde. Die Domäne bietet außerdem die Möglichkeit, Daten nicht nur automatisch herunterzuladen. PDF- und TXT-Dateien können auch nachträglich importiert werden. Änderungen an der Domäne sind durch die verschiedenen Ordner und XML-Dateien zur Speicherung von Meta-Daten nachvollziehbar. Zur Übersichtlichkeit aller zur Berechnung geladenen und erstellten Dateien wird die Domäne in folgender Struktur erstellt:

- Ordner
    - 1_pdf_download
    - 2_pdf_import
    - 3_pdf_change
    - 3b_pdf_error
    - 4_txt_convert
    - 5_txt_import
    - 6_txt_change

    - 6b_txt_error
    - 7_lda_algorithm
- Dateien
    - download_links.csv
    - stopwords.txt

Die **Ordner** stellen die logische Abfolge der einzelnen Prozesse dar. In „1_pdf_download“ werden alle vom Programm heruntergeladenen Dateien gespeichert. In den Ordnern „2_pdf_import“ und „5_txt_import“ können PDF- bzw. TXT-Dateien nachträglich in den Datenbestand der Domäne aufgenommen werden. Die Ordner „3_pdf_change“ und „6_txt_change“ führen die Dateien aus den beiden vorher genannten Ordnern zusammen. In „4_txt_convert“ werden die Ergebnisse der Konvertierung aller PDF-Dateien in TXT-Dateien abgelegt. Können Dateien nicht verarbeitet werden, wird eine Kopie in dem Ordner „3b_pdf_error“ bzw. „6b_txt_error“ abgelegt, abhängig vom Typ der Datei. Abschließend ist der Ordner „7_lda_algorithm“ für die Ablage der Ergebnisse des LDA-Algorithmus vorhanden.

Die **Dateien** stellen individuelle Einstellungen für die Domäne dar. In der Datei „download_links.csv“ müssen alle Links eingepflegt werden, die vom Programm zum Download auf PDF-Dateien analysiert werden sollen. In der Datei „stopwords.txt“ können manuell Stoppwörter[132] gepflegt werden. Diese hierin enthaltenen Wörter werden dann beim Ausführen von LDA nicht berücksichtigt.

Der zur Domäne zugehörige Prozess gestaltet sich sehr einfach. Nach der Auswahl eines lokalen Speicherortes für die Domäne werden an diesem Pfad die genannten Ordner und Dateien erstellt.

3.2.2 Prozess (2) – Daten-Beschaffung

Der Daten-Beschaffungs-Prozess dient dem Zweck der Sammlung aller Dateien, auf denen spätere Auswertungen erstellt werden sollen. Mit ca. 13.500 PDF-Dateien ist ein manuelles Verfahren zum Herunterladen der Dateien mit einem nicht vertretbarem Aufwand verbunden. Der Fokus liegt daher auf einem automatischen Herunterladen von PDF-Dateien – nachfolgend auch als Dokumente bezeichnet.

[132] Siehe Kapitel 2.3.1.

Dazu werden verschiedene Internetseiten, die auf PDF-Dokumente verweisen, in einer CSV-Datei eingepflegt. Nach dem Start des Download-Prozesses arbeitet die Software die Auflistung in der CSV-Datei ab, identifiziert alle PDF-Dateien und lädt diese in den durch die Domäne vordefinierten Ordner „1_pdf_download“ herunter. Bestimmte Meta-Daten wie z.B. Download-Zeitpunkt, Konferenz, Jahr, Download-Link werden in einer XML-Datei je Dokument gespeichert.

Unter Umständen können PDF-Dateien nicht automatisch heruntergeladen werden. Die Domäne soll in diesem Fall die Möglichkeit bieten, den Datenbestand manuell ergänzen zu können. Die Speicherung der Meta-Daten soll auch für diese Dateien möglich sein.

In den folgenden Abschnitten werden zuerst die technischen Voraussetzungen für eine Software zum automatischen Herunterladen definiert. Darauffolgend sind die Prozesse abgebildet, die alle genannten Voraussetzungen erfüllen sollen.

3.2.2.1 Voraussetzungen

Für die Prozesse sollen die nachfolgend genannten Voraussetzungen erfüllt werden. Sie werden im Anschluss detailliert erläutert. Die Anforderungen werden durch den Autor definiert und sind im Laufe einer ersten Anwendung des Verfahrens im Rahmen der Ausarbeitung von Pussep[133] entstanden.

- Webcrawling
    - Sequentielles Herunterladen der PDF-Dateien von Internetseiten
    - Verlinkung der PDF-Datei darf direkt und indirekt sein
- Nachvollziehbarkeit
    - Speicherung des Download-Link
    - Speicherung des Zeitpunkts des Herunterladens
    - Eindeutige Identifikation einzelner Dokumente ermöglichen
- Metadaten-Speicherung
    - Dateiname

[133] Pussep et al. (2012).

    - Konferenzbezeichnung und –jahr
    - Hashwert des Dateiinhalts
    - Manuell veränderbarer Kommentar
- Datensicherheit
    - Kein Überschreiben von Daten
    - Zulässigkeit redundanter Dateien
    - Manuellen Hinzufügen weiterer Dateien ohne Download ermöglichen
    - Unterscheidung von automatisch heruntergeladenen und manuell hinzugefügten Dateien

Durch **Webcrawling**[134] soll es möglich sein, ein manuelles Herunterladen von PDF-Dateien zu vermeiden – ein manuelles Verfahren kann unter Umständen einen sehr hohen Aufwand verursachen. Bei Angabe einer Internetseite, auf der die jeweiligen PDF-Dateien verlinkt sind, sollen automatisch alle Links zu PDF-Dateien erkannt und zum Herunterladen gespeichert werden.

Nachvollziehbarkeit ist für die spätere Analyse der Ergebnisse und die Identifikation möglicher Fehler relevant. Bei Problemen vom Herunterladen der Dateien – beispielsweise konnte die Datei nicht geladen werden oder der Link führt zu keiner Datei – ist die Dokumentation des Links und der Zeitpunkts des Herunterladens zur Fehleranalyse wichtig. Sollen bestimmte wissenschaftliche Ausarbeitungen ausgewählt werden, um diese genauer auf den Inhalt zu überprüfen – beispielsweise passen Sie zu einem gesuchten Thema – so sind diese eindeutig zu identifizieren. Für Auswertungen der gesamten Domäne oder der Erstellung eines zeitlichen Verlaufs von bestimmten Themen müssen die zu einer Datei zugehörigen **Metadaten** gespeichert werden. Neben Dateiname, Konferenz, dem Jahr der Veröffentlichung soll auch ein Hashwert der Datei und ein optional manuell nachpflegbarer Kommentar gespeichert werden können. Dies geschieht vorzugsweise in einem flexibel auslesbarem Format wie z.B. XML.

[134] Software zur automatisieren Analyse von Internetseiten auf bestimmte Informationen. Vgl. Wikipedia (o. J.): Webcrawler.

Inkonsistenzen sollen im Rahmen der **Datensicherheit** möglichst vermieden werden bzw. identifizierbar sein, auch nach manuellem Bearbeiten der Daten. So dürfen Dateien nicht automatisch überschrieben werden, falls Redundanzen – Dateien könnten beispielsweise doppelt verlinkt sein – entstehen. Durch die Metadaten-Speicherung können die Probleme erkannt werden. Wurden Dateien nicht automatisch heruntergeladen, so sollen diese nachträglich manuell in die Domäne ergänzt und automatisch gekennzeichnet werden können.

3.2.2.2 Prozess (2.a) - Herunterladen

Zum Herunterladen von PDF-Dateien durch einen Webcrawler ist ein Prozess notwendig, der die vorher genannten Voraussetzungen erfüllt und die Struktur der Domäne berücksichtigt. Der Prozess ist folgendermaßen aufgebaut:

1. Angabe aller Seiten, auf denen PDF-Dateien verlinkt sind, in einer CSV-Datei mit Konferenz und Jahr nach dem Schema in Tabelle 6 (Konferenz und Jahr sind keine Pflicht-Felder)
2. Pro Zeile der CSV-Datei
    a. Angegebene Download-Seite auf PDF-Links prüfen
    b. Speicherung der extrahierten PDF-Links in einer Queue[135]
    c. Pro Eintrag in Queue
        i. Zufällige siebenstellige ID im Zahlenraum von 10000000 bis 79999999 berechnen.
        ii. Im Ordner „1_pdf_download“ der Domäne einen Ordner mit der gewählten ID als Namen erstellen. Falls die ID bereits vorhanden ist, fahre mit i fort.
        iii. XML-Datei für Metadaten im ID-Ordner erstellen. Speicherung von:
            1. Download-Link aus Queue
            2. Konferenz-Bezeichnung aus der CSV-Datei „download_links.csv“

[135] Queue (engl.): Warteschlange.

3. Konferenz-Jahr aus der CSV-Datei „download_links.csv“
4. Dateiname der PDF-Datei

iv. Herunterladen der entsprechenden PDF-Datei und Speicherung im ID-Ordner

v. Fortfahren mit c. bis Queue leer

d. Fortfahren mit 2. bis alle Zeilen der CSV-Datei durchlaufen wurden

Tabelle 6 - Aufbau der download_links.csv für den Webcrawler

Download-Seite 1	Konferenz-Bezeichnung 1	Konferenz-Jahr 1
⋮	⋮	⋮
Download-Seite n	Konferenz-Bezeichnung n	Konferenz-Jahr n

Nach Durchführung dieses Prozesses sind im Ordner „1_pdf_download“ alle PDF-Dateien jeweils in einem eigenen Ordner zusammen mit einer XML-Datei abgelegt. Anhand der IDs der einzelnen Ordner lässt sich eine PDF-Datei eindeutig identifizieren.

3.2.2.3 Prozess (2.b) – Normalisieren und Importieren

Unter Umständen gibt es Daten-Quellen, die keine Möglichkeit zum Herunterladen von Dateien anbieten. In diesem Fall können PDF-Dateien im Ordner „2_pdf_import“ hinterlegt werden. Der folgende Prozess wird als Normalisierung bezeichnet und führt die manuell hinzugefügten Dateien mit dem Bestand der heruntergeladenen Dateien zusammen.

1. Für jeden Ordner in „1_pdf_download“
    a. Kopieren des Ordners nach „3_pdf_change“, falls dieser dort noch nicht vorhanden ist.
2. Normalisieren: Für jede PDF-Datei im Ordner „2_pdf_import“
    a. Falls die PDF-Datei bereits eine dazugehörige XML-Datei in einem ID-Ordner besitzt, fahre mit Schritt 2 fort.
    b. Freie ID im Zahlenraum von 80000000 bis 89999999 ermitteln.

c. Erstellung des ID-Ordners in „2_pdf_import“

d. Verschieben der PDF-Datei in den ID-Ordner

e. Erstellung einer XML-Datei im ID-Ordner mit folgenden Meta-Daten

i. Dateiname

ii. Konferenz-Bezeichnung aus dem erstem Ordner-Namen unterhalb von „2_pdf_import“, wenn die Datei in einem Unterordner liegen sollte

iii. Konferenz-Jahr aus dem zweiten Ordner-Namen unterhalb von „2_pdf_import“, wenn die Datei in zwei Unterordnern liegen sollte

iv. Download-Link mit Text „manually imported“ füllen

v. Fortfahren mit 2. bis Queue leer

3. Importieren: Für jeden Ordner in „2_pdf_import“

a. Kopieren des Ordners nach „3_pdf_change“, falls dieser dort noch nicht vorhanden ist.

4. Für jeden Ordner in „3_pdf_change“

a. Erstellung des Hashwerts der PDF-Datei und Speicherung in vorhandener XML-Datei

Nach Durchführung dieses Prozesses liegen alle PDF-Dateien, die heruntergeladen oder manuell hinzugefügt wurden, im Ordner „3_pdf_change“. Jede PDF-Datei liegt dort in einem jeweils eigenen Ordner zusammen mit einer XML-Datei. Anhand der ID der einzelnen Ordner lässt sich eine PDF-Datei eindeutig identifizieren. Der Datenbestand an PDF-Dateien ist damit komplett und steht für nachfolgende Verarbeitungs-Prozesse zur Verfügung.

In Kapitel 3.2.3.2 wird noch ein weiterer Normalisierungs-Prozess für das manuelle Hinzufügen von Text-Dateien vorgestellt. Es können zu den bereits gesammelten Dateien also noch weitere hinzukommen.

3.2.3 Prozess (3) - Daten-Aufbereitung

Nach der Sammlung der PDF-Dateien soll der gesamte Datenbestand zur weiteren Verarbeitung aufbereitet werden. Dabei werden die Inhalte der bisher gesammelten PDF-

Dateien ausgelesen und im Text-Format gespeichert. Diese werden wiederum mit manuell importierten Text-Dateien zusammengeführt und im Anschluss in eine CSV-Datei überführt, die als Dateninput für TMT dient und alle Metadaten und Inhalte der Textdateien enthält. Dieser gesamte Prozess lässt somit in 3 Unterprozesse aufteilen. Sie bilden die entsprechenden Unterkapitel. Danach werden mögliche Probleme diskutiert, die während der Verarbeitung auftreten können.

3.2.3.1 Prozess (3.a) - Konvertieren

Der erste Prozess zur weiteren Datenverarbeitung besteht in der Konvertierung der gespeicherten PDF-Dateien in ein Text-Format. Während dieses Vorganges werden zusätzliche Meta-Daten ausgelesen, die in einer XML-Datei bei den Text-Dateien gespeichert werden. Der Prozess gestaltet sich wie folgt:

1. Durchsuchen des Ordners „3_pdf_change“ nach PDF-Dateien.
    a. Für jede PDF-Datei den Pfad in einer Queue speichern.
2. Für jede PDF-Datei in der Queue
    a. Extraktion des textuellen Inhalts aus der PDF-Datei und Speicherung in einer TXT-Datei in einem Ordner, der die ID als Ordnernamen trägt, innerhalb des Ordners „4_txt_convert“.
    b. Auslesen von Informationen aus den Meta-Daten der PDF-Datei
        i. Erstellungsdatum
        ii. Letztes Änderungsdatum
        iii. Keywords
        iv. Seitenanzahl
        v. Autor
        vi. Titel
        vii. Betreff
        viii. Ersteller (z.B. Adobe Acrobat X)
        ix. Produzent (z.B. Microsoft Word 2010)

c. Suche nach einer XML-Datei am Fundort der PDF-Datei.

d. Auslesen aller bereits gespeicherten Meta-Daten aus der XML-Datei.

e. Zusammenfügen der neuen Meta-Daten aus der PDF und der bereits vorhandenen Meta-Daten. Speicherung in einer XML-Datei im Ordner der zugehörigen Text-Datei.

f. Tritt ein Fehler während der Konvertierung einer Datei auf, wird diese zusammen mit ihrer XML-Datei in „3b_pdf_error“ anstatt in „4_txt_convert“ abgelegt.

g. Fortfahren mit 2. bis Queue leer.

Bei dieser Konvertierung können Probleme auftreten. Unter Umständen kann der Inhalt nicht direkt ausgelesen werden. Verschiedene Test-Läufe haben ergeben, dass Dateien nicht ausgelesen werden können, wenn diese eingescannt wurden oder mit einem Schutz versehen sind.

Bei gescannten Dokumenten gibt es die Möglichkeit, eine OCR[136]-Software einzusetzen. Diese kann Text auf Bildern erkennen und in ein Textformat überführen. Dabei versucht die Software, einzelne Buchstaben anhand von Ähnlichkeiten zu erkennen. Im Rahmen der Auswertung von Pussep[137] wurde Adobe Acrobat Professional 9 verwendet. Die Software kann über eine Stapelverarbeitung mehrere Dateien sequentiell abarbeiten. Sie kann den erkannten Text einer PDF-Datei in derselben Datei zu speichern. Jedoch stößt auch eine OCR-Software an ihre Grenzen, wenn die Bildqualität der gescannten Dokumente nicht sehr hoch ist, Seiten beispielsweise um 90 Grad gedreht sind oder der Text durch den Buchknick gewölbt ist.

Sind PDF-Dateien mit einem Schutz versehen, können diese unter Umständen ebenfalls nicht ausgelesen werden. Es gibt verschiedene Stufen, eine PDF-Datei zu schützen. Hier hilft oft nur das manuelle Kopieren des Texts in eine Text-Datei, das Abtippen des Texts oder das Auslesen des Inhalts durch Drucken und Scannen mit Unterstützung einer OCR-Software.

[136] Optical Character Recognition.

[137] Pussep et al. (2012).

3.2.3.2 Prozess (3.b) – Normalisieren und Importieren

Nach dem Abschluss des letzten Prozesses liegt der Datenbestand in Form von TXT-Dateien vor. Wie beim Herunterladen der Dateien auch, soll es eine Möglichkeit geben, an dieser Stelle Dateien manuell zum aktuellen Bestand hinzufügen zu können. Dieser Prozess ist analog zu dem der PDF-Dateien in Kapitel 3.2.2.3 gestaltet und ist nachfolgend Schritt für Schritt aufgeführt:

1. Für jeden Ordner in „4_txt_convert“
    a. Kopieren des Ordners nach „6_txt_change“, falls dieser dort noch nicht vorhanden ist.
2. Normalisieren: Für jede TXT-Datei im Ordner „5_txt_import“
    a. Falls die TXT-Datei bereits eine dazugehörige XML-Datei in einem ID-Ordner besitzt, fahre mit Schritt 2 fort.
    b. Freie ID im Zahlenraum von 90000000 bis 99999999 ermitteln.
    c. Erstellung des ID-Ordners in „5_txt_import“.
    d. Verschieben der TXT-Datei in den ID-Ordner.
    e. Erstellung einer XML-Datei im ID-Ordner mit folgenden Meta-Daten
        i. Dateiname
        ii. Konferenz-Bezeichnung aus dem erstem Ordner-Namen unterhalb von „5_txt_import“, wenn die Datei in einem Unterordner liegen sollte.
        iii. Konferenz-Jahr aus dem zweiten Ordner-Namen unterhalb von „5_txt_import“, wenn die Datei in zwei Unterordnern liegen sollte.
        iv. Download-Link mit Text „manually imported“ füllen.
3. Importieren: Für jeden Ordner in „5_txt_import“
    a. Kopieren des Ordners nach „6_txt_change“, falls dieser dort noch nicht vorhanden ist.
4. Für jeden Ordner in „6_txt_change“
        i. Auslesen des Inhalts der TXT-Datei.
        ii. Ermittlung von Meta-Daten

1. Analyse des Inhalts zur Identifizierung der geschriebenen Sprache.
2. MD5-Hashwert über die Datei erstellen.

iii. Speicherung der Meta-Daten in der vorhandenen XML-Datei.

Nach Durchführung dieses Prozesses ist der Datenbestand für die Durchführung von Analysen komplett. Die Textdateien können nun in ein für TMT lesbares Format umgewandelt werden.

3.2.3.3 Prozess (3.c) - Konsolidieren

Da alle Dateien für die Durchführung der eigentlichen Analyse bereitliegen, besteht die nächste Notwendigkeit in einer Konvertierung der TXT-Dateien in ein CSV-Format, dass von TMT verarbeitet werden kann. Die Software benötigt für die Durchführung von LDA alle Daten in einer CSV-Datei, die folgende Voraussetzungen erfüllen muss[138]:

- Pro Dokument wird eine Zeile erstellt. Die erste Zeile darf auch die Überschriften enthalten, wenn dies in der Implementierung als Parameter gesetzt wird.
- Eine Spalte muss je Dokument eine eindeutig identifizierbare Nummer (ID) besitzen.
- Eine Spalte wird für die Speicherung des Inhaltes eines Dokuments verwendet.
- In allen Zellen dürfen keine Zeilenumbrüche enthalten sein.

Neben den zur Erfüllung der Voraussetzungen geforderten Daten können weitere Spalten mit Meta-Daten im CSV-Dokument gespeichert werden. Nach der Durchführung von LDA können die Ergebnisse nach einer der Spalten differenziert werden, beispielsweise nach dem zugehörigen Jahr oder der jeweiligen Konferenz. Für die weitere Verarbeitung ist es wichtig, dass in der zweiten Spalte die ID gespeichert ist. Die Spaltennummern der Inhalte und der Meta-Daten für eine potenzielle Differenzierung werden direkt in der zu implementierenden Software konfiguriert.

In Tabelle 7 ist das vom Autor definierte Format für die CSV-Datei aufgeführt. Die Meta-Daten umfassen folgende Informationen: Dateiname, Dateigröße, Download-Link, Seitenanzahl,

[138] Vgl. The Stanford Natural Language Processing Group (2011).

Titel, Autor, Betreff, Stichwörter, Ersteller, Produzent, Erstellungsdatum, Änderungsdatum, Sprache, Konferenzname, Konferenzjahr, Download-Zeitpunkt.

Tabelle 7 - Format der CSV-Datei für LDA

rownumber	**id**	**<Metadatum Überschrift 1>**	**...**	**<Metadatum Überschrift m>**	**content**
1	ID 1	Metadatum 1,1	...	Metadatum 1,m	Inhalt 1
⋮	⋮	⋮	⋱	⋮	⋮
n	ID n	Metadatum n,1	...	Metadatum n,m	Inhalt n

Der Prozess zur Erstellung der CSV-Datei für TMT gestaltet sich folgendermaßen:

1. Für jeden Ordner in „6_txt_change“
    a. Speicherung der jeweiligen TXT-Datei in einer Queue.
2. Erstelle im Ordner „7_lda_algorithm“ einen Unterordner nach dem Schema „<jahr><monat><tag>_<stunde><minute><Sekunde>“[139]. In diesem Ordner wird die Datei „converted_file.csv“ erstellt.
3. Erstellung der Überschriften in der Datei „converted_file.csv“.
4. Für jede TXT-Datei in der Queue
    a. Auslesen des Datei-Inhaltes.
    b. Auslesen der Ordnernamens, der die TXT-Datei enthält, für die ID.
    c. Auslesen der Meta-Daten der am Speicherort der TXT-Datei vorhandenen XML-Datei.
    d. Tritt ein Fehler während der Konsolidierung einer Datei auf, wird diese zusammen mit ihrer XML-Datei in „6b_txt_error“ abgelegt anstatt in der CSV-Datei abgespeichert.
    e. Erstelle eine neue Zeile in der CSV-Datei

[139] Beispielsweise „20120321_145502“ für den 21.03.2012 um 14.55 Uhr

i. Speicherung der fortlaufenden Zeilennummer in Spalte 1.

ii. Speicherung der ID in Spalte 2.

iii. Speicherung aller Metadaten - pro Spalte ein Merkmal der Meta-Daten.

iv. Speicherung des gesamten Inhalts der Textdatei in der letzten Spalte.

f. Fortfahren mit 3. bis Queue leer.

5. Abschluss der CSV-Datei (Speicherung im Format „ISO-8859-1“).

Nachdem der Speicherung der CSV-Datei abgeschlossen ist, kann TMT die Daten mit LDA analysieren.

3.2.4 Prozess (4) - Latent Dirichlet Allocation

Da LDA für unterschiedliche Anwendungszwecke eingesetzt werden kann, wird dieser Prozess in 4 weitere Unterprozesse gegliedert:

(a) Anzahl Topics berechnen

(b) Topic-Modell erstellen

(c) Topic-Modell anwenden

(d) Topic-Modells differenzieren

Bevor das eigentliche Topic-Modell mit den dazugehörigen Ergebnissen berechnet werden kann, ist es ratsam, eine möglichst optimale Anzahl an Topics zu schätzen (a). Nachdem eine gewünschte Anzahl an Topics ermittelt wurde, kann das eigentliche Topic-Modell anhand eines LDA-Approximationsverfahrens berechnet werden (b). Ist ein Modell vorhanden, kann dieses angewendet (c) werden, um die gewünschten Ergebnisse der Analyse zu erhalten. Zusätzlich kann das Modell auch nach einer weiteren Spalte in der CSV-Datei differenziert (d) werden, um z.B. eine Zeitreihe über alle Topics zu erhalten.

Die für die Prozesse notwendigen Einstellungen sind in Tabelle 8 aufgeführt. In der ersten Zeile sind die einzelnen Prozesse aufgelistet, während in der ersten Spalte die einzelnen Einstellungen aufgeführt sind und ob sie von LDA oder TMT benötigt werden. Ein „x“ in einer Zelle bedeutet, dass die Einstellungen von dem Prozess benötigt wird.

Tabelle 8 - Nötige Einstellungen einzelner LDA-Prozesse

	(1) Perplexity berechnen	(2) Modell erstellen	(3) Modell anwenden	(4) Modell differenzieren
LDA				
Anzahl Topics	(als Liste)	x		
Topic Smoothing	x	x		
Term Smoothing	x	x		
Approximatives Verfahren	x	x		
Anzahl an Iterationen	x	x		
TMT				
Auswahl der CSV-Datei	x	x	x	x
Inhalts-Spalte auswählen	x	x	x	x
Übungsanteil einstellen	x			
Auswahl berechnetes Modell			x	x

Die Einstellungen für LDA wurden bereits in 2.3.1 ausführlich diskutiert. An **approximativen Verfahren** stehen in TMT „CVB0LDA“ und „GibbsLDA“ zur Verfügung. Standardmäßig wird „CVB0LDA“ aus Vorteilen in der Berechnungszeit genutzt. Das Verfahren nutzt alle Prozessor-Kerne und der Log-Likelihood-Schätzer konvergiert über die Iterationen schneller als bei „GibbsLDA“ und kann damit früher abgebrochen werden kann. Andererseits verbraucht das letztgenannte Verfahren weniger Arbeitsspeicher während der Berechnung und ermöglicht damit die Berechnung großer Datenbestände, die mit „CVB0LDA“ nicht möglich wären.[140]

Die Einstellungen für TMT sind bei allen Prozessen sehr ähnlich. Die Software soll unterschiedliche CSV-Dateien verarbeiten können, auch wenn diese mit einem anderen Programm erstellt wurden. Durch die Zeitstempel der erstellten CSV-Dateien im Ordner „7_lda_algorithm“ sind unter Umständen mehrere CSV-Dateien in der Domäne vorhanden. Daher muss die gewünschte **CSV-Datei ausgewählt** werden.

[140] The Stanford Natural Language Processing Group (2011).

Um unterschiedliche CSV-Dateien verarbeiten zu können, muss in SIERRA die zu analysierende **Spalte mit dem Inhalt** ausgewählt werden.

Für die Perplexity-Berechnung ist es wichtig, dass ein **Übungsanteil eingestellt** wird. Dieser Wert reicht von 0 bis 1 und gibt es den Anteil jedes Dokuments an, auf dem das LDA-Modell erstellt wird, um danach das Modell auf den verbleibenden Teil des Textes anzuwenden. Danach wird der jeweilige Perplexity-Wert[141] der zweiten Berechnung ermittelt. Dieser Vorgang wird für jede zu testende Topic-Anzahl wiederholt.

Bei den Prozessen (c) und (d) werden ausschließlich Berechnungen vorgenommen, die auf dem in (b) **berechneten Modell** basieren. Daher muss das für die Auswertungen gewünschte zugrunde liegende Modell ausgewählt werden.

Bevor die Prozesse im Einzelnen vorgestellt werden, soll die Datenvorbereitung innerhalb von TMT aufgefasst werden. Diese Einstellungen werden im Programmiercode der Software enthalten sein.

Die Inhalte werden schrittweise aufeinander aufbauend verarbeitet:

1. Leerzeichen und Punktuation entfernen.
2. Alle buchstaben „verkleinern“ (lowercase).
3. Non-words und non-numbers entfernen.
4. Porterstemmer[142] anwenden.
5. Wörter mit weniger als 3 Zeichen entfernen.
6. Stoppwörter entfernen (in TMT integrierte Liste[143] und eine eigene definierte Liste aus „stopwords.txt“).
7. Entfernen aller Wörter, die in weniger als 4 Dokumenten enthalten sind.
8. Entfernen der 0 am häufigsten vorkommenden Wörtern.
9. Dokumente mit weniger als 5 Wörtern – die nicht herausgefiltert wurden – entfernen.

[141] Siehe Kapitel 2.3.2.

[142] Siehe Kapitel 2.3.1.

[143] Siehe Anhang A.

Nach der Erläuterung der wichtigen Einstellungen je Prozess und der Einschränkung des Datenbestandes werden die Unterprozesse in den folgenden Unterkapiteln in der Reihenfolge ihrer Vorstellung erklärt.

3.2.4.1 Prozess (4.a) – Anzahl Topics berechnen

Zur Berechnung einer annähernd optimalen Anzahl an Topics für die eigentliche LDA-Berechnung in Prozess (4.b), muss das Modell für jede zu untersuchende Topic-Anzahl berechnet werden. Der Perplexity-Wert hilft, diesen Wert zu ermitteln. Wie in Kapitel 2.3.2 bereits erläutert wurde, soll der Wert über eine steigende Anzahl an Topics mindestens konvergieren. Im Optimum erhält man ein Minimum in der Menge der berechneten Werte. Letztes ist jedoch durch meist große Datenbestände und begrenzter Rechenleistung kaum zu bewerkstelligen, eine deutliche Konvergenz ist daher ausreichend.

Nach dem Einstellen der Parameter aus Tabelle 8 kann der Prozess direkt gestartet werden. Sobald die Berechnung beendet ist, werden die Perplexity-Werte je Topic-Anzahl ausgegeben. Die Interpretation der Ergebnisse und damit die Festlegung einer Anzahl an Topics erfolgt durch den Benutzer.

Während der Berechnung wird im Ordner der CSV-Datei eine Archiv-Datei erstellt, die beispielsweise den Namen „converted_file.csv.term-counts.cache.a209f489.gz“ trägt. Sie wird von TMT als Zwischenspeicher für Berechnungen erstellt. In dem Archiv ist das gesamte Vokabular des Datenbestandes mit der absoluten Häufigkeit je Wort enthalten.

3.2.4.2 Prozess (4.b) – Modell erstellen

Dieser Prozess bildet das Kernstück von TMT. Anhand der Daten aus Kapitel 3.1, der Parameter aus Tabelle 8 und der Anzahl an Topics aus Prozess (4.a) in Kapitel 3.2.4.1 kann das gewählte LDA-Approximationsverfahren angewendet werden. Das Ergebnis ist ein Topic-Modell, das auf die gleichen oder auch andere Daten in (4.c) angewendet werden kann und damit die strukturelle Analyse der Daten ermöglicht.

Während der Berechnung erstellt TMT unterschiedliche Dateien. Neben dem Archiv, das bereits in Prozess (4.a) genutzt wurde, wird ein Unterordner im Verzeichnis der CSV-Datei „converted_file.csv“ erstellt. Dieser kann beispielsweise den Namen „lda-a76da864-30-

d4249a60“ tragen. Der Name setzt sich aus einer Prüfziffer der Daten und der Einstellungen zusammen. TMT legt in diesem Ordner mehrere Dateien und Ordner ab, die nachfolgend erklärt werden:

- document-topic-distributions.csv
- description.txt

In der Datei **„document-topic-distributions.csv“** ist für jedes Dokument des Datenbestandes eine Zeile vorhanden. In der ersten Spalte ist die ID des jeweiligen Dokuments hinterlegt. Die restlichen Spalten bilden die Verteilung von insgesamt K Topics über jeweils ein Dokument. Die Summe aller Topic-Anteile ist 1. Der schematische Aufbau der Datei ist in Tabelle 9 dargestellt.

In der Datei **„description.txt“** sind die gesamten Einstellungen hinterlegt, die zur Berechnung des Modells angegeben wurden.

Tabelle 9 - Schematischer Aufbau der Datei document-topic-distributions.csv

ID 1	Topic-Anteil 1,1	...	Topic-Anteil 1, K
⋮	⋮	⋱	⋮
ID n	Topic Anteil n,1	...	Topic Anteil n,K

Innerhalb des Modell-Ordners wird während der Berechnung alle 50 Iterationen ein Ordner mit der entsprechenden Iterationsnummer, aufgefüllt mit Nullen auf 5 Stellen, erstellt. Jede dieser Iterationen beinhaltet ein komplettes Modell. Innerhalb des Ordners werden die nachfolgend aufgezählten Dateien von TMT abgelegt. Die Erläuterung zu jeder Datei ist direkt hinter dem Dateinamen vermerkt[144]:

- description.txt – Einstellungen für die Parameter.
- log-probability-estimate.txt – berechneter Log-Likelihood-Schätzer zur Bewertung der Iterationen[145].

[144] The Stanford Natural Language Processing Group (2011).

[145] Siehe Kapitel 2.3.2.

- params.txt – Einstellungen für die LDA-Berechnung.
- summary.txt – Auflistung der 20 Top-Wörter je Topic, bewertet nach Relevanz und der kumulierten Gesamt-Relevanz pro Topic.
- term-index.txt – Das gesamte Vokabular der Daten.
- tokenizer.txt – Einstellungen für den Tokenizer[146].
- topic-term-distribitions.csv.gz – gepacktes Archiv mit allen Topics, den zugehörigen Wörtern und deren absolute Häufigkeit.

Nach Abschluss dieses Prozesses liegt das berechnete Topic-Modell vor. Der Ordner mit der höchsten Iterationsnummer beinhaltet das beste Modell und bildet die Grundlage für die folgenden Prozesse.

3.2.4.3 Prozess (4.c) – Modell anwenden

Nach der Berechnung des Topic-Modells kann dieses zur Generierung von Ergebnissen genutzt werden. In der Regel wird es auf den gleichen Daten angewendet, die bereits zur Berechnung des Modells genutzt wurden. Dies ist nicht zwingend. Es kann genauso gut auf andere Daten angewendet werden. Die Topics verändern sich in jedem Fall nicht. Jedes Dokument wird lediglich auf die inhaltliche Topic-Struktur analysiert.

Für die Anwendung des Modells ist es nicht weiter notwendig, LDA-zugehörige Parameter anzugeben. Lediglich der Pfad des berechneten Modells muss für TMT angegeben werden.

Nachdem der Prozess abgeschlossen ist, sind folgende Dokumente im LDA-Ordner zu finden:

- converted_file-document-topic-distributions.csv
- converted_file-usage.csv
- converted_file-top-terms.csv

Die Datei **„converted_file-document-topic-distributions.csv“** ist analog zu Tabelle 9 und der Beschreibung in Kapitel 2.2.1. Sie beschreibt die Verteilung der Topics über jedes Dokument.

[146] Einstellungen für die Verarbeitung einzelner Wörter. Vgl. The Stanford Natural Language Processing Group (2011).

In der Datei **„converted_file-usage.csv“** sind die absoluten Häufigkeiten von Dokumenten und Wörtern festgehalten, die einem jeweiligen Topic zugeordnet wurden. Hier ist zu beachten, dass die absolute Häufigkeit ein Dezimal-Wert ist, da ein Wort eine Verteilung über alle Topics besitzt. Die Summe über die Spalte „Documents“ bzw. „Words“ ergibt die Gesamtanzahl an verfügbaren Dokumenten bzw. Wörtern. Das Schema ist in Tabelle 10 abgebildet.

Tabelle 10 - Schematischer Aufbau der Datei converted_file-usage.csv

Topic	**Documents**	**Words**
Topic 1	Absolute Häufigkeit Dokumente 1	Absolute Häufigkeit Wörter 1
⋮	⋮	⋮
Topic K	Absolute Häufigkeit Dokumente K	Absolute Häufigkeit Wörter K

Eine Auflistung der 50 am häufigsten vorkommenden Wörter je Topic sind in der Datei **„converted_file-top-terms.csv“** zu finden. Das Schema ist in Tabelle 11 zu sehen.

Tabelle 11 - Schematischer Aufbau der Datei converted_file-top-terms.csv

Topic	**Top Terms**		
Topic 1	Wort 1,1	...	Wort 1, 50
⋮	⋮	⋱	⋮
Topic K	Wort K,1	...	Wort K,50

Nach dem Abschluss dieses Prozesses liegen bereits verwertbare Ergebnisse vor, die für eine weiterführende Analyse verwendet werden können. Erklärungen, wie die Ergebnisse interpretiert und ausgewertet werden können, sind in Kapitel 4.4 zu finden.

3.2.4.4 Prozess (4.d) – Modell differenzieren

Der letzte für LDA relevante Prozess kann Daten anhand eines vorliegenden Modells differenziert berechnen, so dass detailliertere Analysen angestellt werden können. Z.B. bietet

sich so die Möglichkeit, Zeitreihen zu erstellen. Für diesen Prozess muss in den Einstellungen der Software eine Spalte festgelegt werden, nach der die Daten differenziert werden sollen. Er ist unabhängig von Prozess (4.c), erstellt wie dieser aber ähnliche Ergebnisse in Form der folgenden Dateien:

- converted_file-sliced-usage.csv
- converted_file-sliced-top-terms.csv

Der schematische Aufbau der beiden Dateien ist analog zu den Tabellen aus Kapitel 2.2.1 mit dem Unterschied, dass jedes Topic nach dem ausgewählten Merkmal in der Spalte „Group ID“ differenziert wird, z.B. nach Jahren.

Tabelle 12 - Schematischer Aufbau der Datei converted_file-sliced-usage.csv

Topic	**Group ID**	**Documents**	**Words**
Topic 1	Merkmal 1	Absolute Häufigkeit Dokumente 1	Absolute Häufigkeit Wörter 1
⋮	⋮	⋮	⋮
Topic K	Merkmal K	Absolute Häufigkeit Dokumente K	Absolute Häufigkeit Wörter K

Tabelle 13 - Schematischer Aufbau der Datei converted_file-sliced-top-terms.csv

Topic	**Group ID**	**Top Terms**		
Topic 1	Merkmal 1	Wort 1,1	...	Wort 1, 50
⋮	⋮	⋮	⋱	⋮
Topic K	Merkmal K	Wort K,1	...	Wort K,50

Die vorliegenden Dateien sind hilfreich, um weiterführende Analysen anstellen zu können. Erklärungen, wie die Ergebnisse interpretiert und ausgewertet werden können, sind in Kapitel 4.4 zu finden.

3.3 Software

Die zu implementierende Software trägt die vom Autor gewählte Bezeichnung „scientific information extraction for rigorous relevance analysis“ (SIERRA). Sie basiert auf den definierten Prozessen und soll eine grafische Oberfläche besitzen, die einfach zu bedienen ist. Die Berechnung der Ergebnisse erfolgt durch die Einbindung der „Stanford Topic Modeling Toolbox 0.4.0“ (TMT) und verwendet mehrere Approximationsverfahren von „Latent Dirichlet Allocation“ (LDA).

Nach der Anwendung von SIERRA stehen verschiedene Auswertungen zur Verfügung. Wie diese interpretiert werden können, wird in Kapitel 4.4 diskutiert.

3.3.1 Systemanforderungen

Damit SIERRA grundsätzlich funktioniert, ist ein installiertes Java-Paket ab Version 1.6 notwendig. Die Funktionalitäten wurden mit den Betriebssystemen Mac OS X 10.7, Windows 7 Professional und Windows Server 2008 R2 erfolgreich getestet. Aus den Erfahrungen mit TMT und SIERRA sind folgende Empfehlungen für eine Betriebsumgebung entstanden:

- Arbeitsspeicher: Mehr als 4 GB, da bei einer Verarbeitung von mehreren tausend Dokumenten die Berechnung vom Approximationsverfahren „CVB0LDA“ abgebrochen wird, falls nicht genügend Arbeitsspeicher zur Verfügung steht. Alternativ kann „GibbsLDA“ als Verfahren verwendet werden, da dieses weniger Arbeitsspeicher als „CVB0LDA“ benötigt. Mit einer steigenden Anzahl an Topics und Dokumenten steigt der Bedarf an Arbeitsspeicher.
- Prozessor: Mindestens ein 64bit-Prozessor, da mehr als 4GB Arbeitsspeicher adressiert werden können. Mehr als 1 Kern ist empfehlenswert, da diese von TMT vollständig genutzt werden können. Ein Prozessor mit 4 Kernen hatte in mehreren Testläufen Fehler erzeugt, jedoch konnten die Fehler nicht eindeutig auf die Kerne zurückgeführt werden.
- Festplattenspeicher: Durch die voneinander abgegrenzten Prozesse in SIERRA wird jede PDF-Datei und jede Text-Datei doppelt gehalten. In der für LDA benötigten CSV-Datei sind ebenfalls nochmal alle Inhalte der Text-Datei vorhanden. Für einen

kompletten Durchgang mit LDA sollte mindestens das 5-fache der zu untersuchenden Datenmenge zur Verfügung stehen.

3.3.2 Starten der Software

Zum Starten kann die Datei „sierra.jar“ direkt geöffnet werden. Jedoch reserviert Java mit standardmäßig ca. 300-400 MB reservierten Arbeitsspeichers für die Anwendung des LDA-Algorithmus bedeutend zu wenig Speicher. Daher gilt die Empfehlung, folgenden exemplarisch dargestellten Startbefehl über die Eingabeaufforderung oder Konsole auszuführen:

```
java –Xmx16000m –Xms16000m –jar sierra.jar
```

Die Parameter „–Xmx“ und „–Xms“ beschreiben dabei den reservierten Speicherplatz durch die Anwendung. „16000m“ bedeutet, dass 16000 Megabyte an Arbeitsspeicher reserviert werden. Dieser Wert sollte auf den frei verfügbaren Arbeitsspeicher angepasst werden. Durch die Festplatten-Auslagerung des Betriebssystems ist es durchaus möglich, mehr Arbeitsspeicher, als physisch vorhanden ist, zu reservieren. Durch die Datenmenge, die der LDA-Algorithmus im Arbeitsspeicher hält, empfiehlt sich diese Vorgehensweise nicht. Die Laufzeit von speicherintensiven Anwendungen erhöht sich durch die Auslagerung des Arbeitsspeichers.

3.3.3 Vorstellung der grafischen Oberfläche

Nach dem Start von SIERRA öffnet sich ein Fenster. Das Erscheinungsbild unterscheidet sich je nach Betriebssystem leicht. Die einzelnen Elemente zur Bedienung der grafischen Oberfläche (GUI) sind in Abbildung 9 zu sehen.

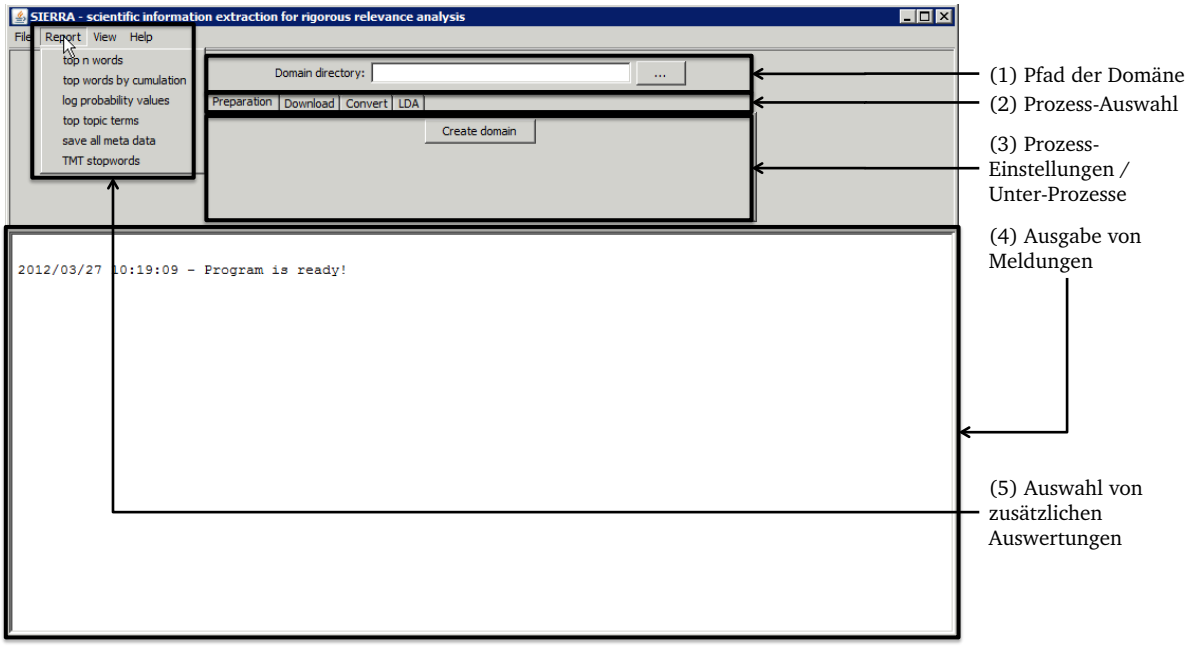


Abbildung 9 - Grafische Oberfläche von SIERRA mit Beschreibungen der Bedienelemente

Im oberen Teil (1) wird das Verzeichnis der Domäne angegeben. Es muss sich um einen lokalen Ordner handeln. Ein Klick auf den Button „...“ öffnet den Auswahldialog des Verzeichnisses für die Domäne. In Teil (2) sind die Haupt-Prozesse abgebildet. Teil (3) bietet alle zum Prozess zugehörigen Unterprozesse und Einstellungsmöglichkeiten an. Die Ausgabe von einzelnen Meldungen erfolgt in Teil (4) mit der Angabe des jeweiligen Datums und der Uhrzeit. Über das Menü kann eingestellt werden, ob der Ausgabebereich bei jeder neuen Meldung mitscrollen soll.

3.3.4 Benutzte Bibliotheken

Für die Entwicklung von SIERRA wurden neben TMT weitere Bibliotheken eingebunden, die nachfolgend im Einzelnen aufgelistet sind. Im Anschluss wird deren Verwendung kurz erklärt.

- TMT 0.4.0
- Scala 2.9.1
- Jlangdetect europarl 0.3 und jlangdetect 0.3
- Log4j
- pdfbox 1.6.0

- OpenCSV 2.3
- Httpclient 4.1.2
- commons-codec 1.4
- commons-logging 1.1.1
- commons-lang 3.1

Die Stanford Topic Modeling Toolbox (**TMT**)[147] bildet die Grundlage für die Berechnungen mittels LDA im Rahmen der Entwicklung der Software. Sie ist an der Stanford Natural Language Processing Group entstanden und erlaubt es, auf einfache Weise LDA anzuwenden. Alle Parameter der Berechnung können manuell beeinflusst werden. Eine detaillierte Beschreibung ist in Kapitel 2.3.1 zu finden. TMT steht unter der Lizenz GPLv2[148].

Scala ist eine skalierbare Programmiersprache, die flexibel für unterschiedliche Anwendungszwecke eingesetzt werden kann.[149] Im Rahmen von SIERRA wird Scala für die Steuerung der Schnittstelle zur TMT benötigt. Die Sprache kann auf einer Java Virtual Machine ausgeführt werden und steht unter einer eigenen Lizenz[150], die der BSD-Lizenz sehr ähnlich ist[151].

Für LDA sollen möglichst nur englischsprachige Artikel[152] verwendet werden. Zur Gewährleistung dieser Voraussetzung wird **jlangdetect** eingesetzt. Mithilfe von N-Grammen[153] wird der zu untersuchende Text in einzelne Buchstaben und Buchstabenketten zerlegt, um anhand von linguistischen Wahrscheinlichkeitsverteilungen die geschriebene Sprache zu erkennen. Die Bibliothek steht unter der Apache-2.0-Lizenz.[154]

---

147 The Stanford Natural Language Processing Group (2011).

148 Free Software Foundation (1991).

149 Wikipedia (o. J.): Scala (Programmiersprache).

150 École Polytechnique Fédérale de Lausanne (2012).

151 Wikipedia (o. J.): Scala (Programmiersprache).

152 Siehe Kapitel 3.1.3.

153 Vgl. Niesler; Woodland (1996), S. 164.

154 The Apache Software Foundation (2004).

Apache **log4j**[155] ist ein Framework zur Protokollierung von Ereignissen und der Generierung von Ausgaben.

Das Auslesen der Inhalte von PDF-Dateien übernimmt Apache **PDFBox**[156]. Zusätzlich können auch Meta-Daten und nach Bedarf auch Tabellen, Bilder und weitere Objekte extrahiert werden.

Zur sequentiellen Verarbeitung mehrerer Download-Seiten wird eine CSV-Datei eingelesen. Das Einlesen von Dateien bietet Java bereits umfassend an. CSV-Dateien können über die Bibliothek **OpenCSV** jedoch komfortabler gelesen und gespeichert werden. Die Bibliothek wird in der Apache 2.0 Lizenz angeboten.[157]

Zur Interaktion mit Internetseiten über das HTTP-Protokoll bietet **Apache HttpClient** zahlreiche Funktionen an. Die Bibliothek wird zum Einlesen der einzelnen Internetseiten, dem Extrahieren einzelner Links und der Identifikation von PDF-Dateien genutzt. Zum Zeitpunkt der Fertigstellung dieser Thesis hat Apache bekannt gegeben, dass Apache HttpClient nicht weiter entwickelt wird. Die Weiterentwicklung wird unter dem Projekt Apache HttpComponents stattfinden.

Eine sinnvolle Ergänzung zu den in Java verfügbaren Funktionen bieten die von Apache bereitgestellten Bibliotheken „**commons-codec**“[158], „**commons-logging**“[159] und „**commons-lang**“[160]. Sie können für Vereinfachungen im Code eingesetzt werden, wie z.B. das Verbinden von String-Arrays oder ein vereinfachtes Event-Logging.

Alle verwendeten Apache-Projekte stehen unter der Apache 2.0-Lizenz[161].

---

155 The Apache Software Foundation (2011d).

156 Ders. (2012).

157 Ders. (2004).

158 Ders. (2011b).

159 Ders. (2011a).

160 Ders. (2011c).

161 Ders. (2004).

3.3.5 Ergebnisse

Neben den automatisch von TMT erstellten Dateien[162] können weitere in SIERRA implementierte Berichte generiert werden. Diese sind in der Oberfläche über den Menüpunkt „Reports“ zu erreichen:

- top n words
- top words by cumulation
- TMT stopwords
- log probability values
- top topic terms
- save all meta data

Über den Bericht **„top n words“** können die n am häufigsten vorkommenden Wörter im Korpus ausgegeben werden. Die Liste ist bereits um die in TMT integrierten Stoppwörter bereinigt. Sollen bestimmte Wörter bei der Berechnung von LDA nicht verwendet werden, können diese in der Datei „stopwords.txt“ eingetragen werden. Es empfiehlt sich, häufig vorkommende Wörter auszuschließen, damit das Vokabular kleiner wird[163]. Der Bericht **„top words by cumulation“** dient dem gleichen Zweck wie „top n words“ mit dem Unterschied, dass z.B. alle der Wörter angezeigt werden, die 10 % der Gesamthäufigkeit aller Wörter ausmachen. Der Prozentanteil kann frei eingestellt werden und wird als Dezimalwert angegeben.

Der Bericht **„TMT stopwords“** gibt alle Stoppwörter aus, die in TMT bereits integriert sind. Aktuell besteht die Liste aus rein englischen Wörtern.

Sollen die Log-Likelihood-Schätzer des berechneten Modells für jede bereits gespeicherte Iteration ausgegeben werden, empfiehlt sich der Bericht **„log probability values“**. Das Delta zwischen den Iterationen sollte mit steigender Anzahl an Iterationen möglichst klein werden, d.h. der Schätzer soll deutlich konvergieren.[164]

---

[162] Siehe Kapitel 3.2.4.

[163] Siehe Kapitel 2.3.1.

[164] Siehe Kapitel 2.3.2.

Mithilfe des Berichts **„top topic terms“** wird in dem Ordner mit der höchsten Iterationsnummer die Datei „summary.txt“ analysiert und unter dem Dateinamen „summary.csv“ gespeichert. Je Topic beinhaltet sie 10 % der am häufigsten vorkommenden Wörter, jedoch mindestens 5 und maximal 10 Wörter. Pro Zeile ist ein Topic gespeichert. In der zweiten Spalte ist der kumulierte numerische Relevanz-Wert enthalten, der auch in der „summary.txt“ zu jedem Topic hinterlegt ist. Je nach Einstellung kann ein Modell eine hohe Anzahl an Topics besitzen. Um möglicherweise für die gewünschte Untersuchung relevante Topics zu identifizieren soll dieser Bericht dienen. Er bietet eine kurze Übersicht über alle Topics in tabellarischer Form.

Über den Bericht **„save all meta data“** wird in jedem Ordner direkt unter der Domäne eine CSV-Datei erstellt, die alle IDs und Metadaten der in dem jeweiligen Ordner gespeicherten Dateien enthält. Der Dateiname ist mit einem Zeitstempel versehen und setzt sich nach dem Schema „<jahr><monat><tag>_<stunde><minute><sekunde>_xml_summary.csv“ zusammen.[165]

Für eine Anwendung von LDA sind die Daten definiert worden. Die datenverarbeitenden Prozesse wurden mit den zu erstellenden Dateien detailliert dargestellt. SIERRA implementiert die vorgestellten Prozesse und kann nun den Korpus analysieren.

[165] Beispielsweise „20120322_095451_xml_summary.csv“.

4 Anwendung

Nachdem in den vorherigen Kapitel die nötigen Grundlagen für eine Erstellung von Analysen über einen Korpus vermittelt wurden, soll das gesamte Verfahren mit Unterstützung von LDA praktisch durchgeführt werden. Dafür steht der in Kapitel 3.1 vorgestellte Datenbestand zur Verfügung. Mit den in SIERRA implementierten Prozessen werden die PDF-Dateien im Datenbestand in Text-Dateien konvertiert, um anschließend in einer CSV-Datei in tabellarischer Form gespeichert zu werden. Die Tabelle enthält außerdem weitere Meta-Daten, die während der gesamten Verarbeitung der einzelnen Dateien extrahiert werden konnten. Die CSV-Datei kann dann von TMT für die Erstellung eines Topic-Modells durch LDA verarbeitet werden. Das Modell dient als Grundlage für weitere Auswertungen, wie z.B. die Anwendung des Modells auf den bereits bestehenden oder einen neuen Datenbestand. Alternativ kann der Datenbestand auf ein Merkmal differenziert werden, um z.B. Zeitreihen zu erstellen.

In diesem Kapitel sollen zuerst die technischen Aspekte wie Laufzeiten und die Ausstattung des Test-Computers vorgestellt werden. Danach folgt eine Analyse des Datenbestandes über die einzelnen Prozesse und welche Probleme dabei auftauchen können. Am Schluss wird eine inhaltliche Beispiel-Auswertung vorgestellt, wie die Ergebnisse zu interpretieren sind und wie die Untersuchungen vertieft werden können.

4.1 Technische Ausstattung

Die gesamte Berechnung wird auf einem Server der „Forschungsgemeinschaft Frankfurt Cloud“[166] durchgeführt. Dabei handelt es sich um ein gemeinsames Projekt zur Bereitstellung von Rechen-Ressourcen für „Cloud Computing“, initiiert von der Frankfurter Goethe-Universität und mehreren Unternehmen. Diese Ressourcen stehen auch den Universitäten in Hessen zur Verfügung. Für die Durchführung von LDA wird eine virtuelle Maschine mit folgenden Spezifikationen genutzt:

- Prozessor: AMD Opteron 6172 2.10 GHz mit 2 Kernen
- Arbeitsspeicher: 16 GB

[166] Jochum (2012).

- Festplatte: 70 GB
- Betriebssystem: Windows Server 2008 R2 Datacenter

Für die Berechnung wurden mehrere Konfigurationen zum Betrieb von SIERRA getestet. Dabei sind mehrere Probleme aufgetaucht, die Prozessor, Arbeitsspeicher und Festplatte betreffen.

Bei der Durchführung von LDA mit dem Näherungsverfahren „CVB0LDA“ belastet TMT alle Prozessor-Kerne gleichmäßig. Je mehr Kerne vorhanden sind, desto schneller können die Berechnungen durchgeführt werden. Bei einem Test mit einem Prozessor mit 4 Kernen sind bei der Überprüfung der Ergebnisse in der Datei „document-topic-distributions.csv“ Wahrscheinlichkeitswerte gefunden worden, die negativ oder größer als 1 sind. Jedoch sollte die Summe aller Wahrscheinlichkeiten je Dokument über alle Topics genau 1 ergeben. Das Verhalten ist nicht eindeutig auf die Anzahl der Prozessor-Kerne zurückführbar, ist aber bei 2 Kernen nicht aufgetreten. Ein Test mit dem Gibbs-Verfahren war auf dem 4-Kern-Prozessor zeitlich nicht durchführbar.

Während der Berechnung eines Topic-Modells lagert LDA den gesamten Korpus und weitere Daten in den Arbeitsspeicher aus. Bei ca. 13.500 Dokumenten und 300 Topics wurden bereits die gesamten zur Verfügung stehenden 16 GB an Arbeitsspeicher genutzt. Rechner mit weniger Speicher waren nicht in der Lage, LDA für den Korpus durchzuführen. Alternativ kann das Gibbs-Verfahren genutzt werden, das vergleichsweise wenig Speicher benötigt, aber lange Laufzeiten benötigt.

Zuerst standen dem Rechnern rund 30 GB an Festplattenspeicher zur Verfügung. Dieser Wert wurde später auf 70 GB erhöht. Von diesem müssen rund 10-15 GB für die Belegung des Betriebssystems abgezogen werden. In Tabelle 14 ist der belegte Speicherplatz für die Domäne dieser Anwendung dargestellt. Eine ausreichend große Festplatte ist also empfehlenswert.

Tabelle 14 - Benötigter Speicherplatz in GB nach Ordner in der Domäne, gerundet auf 2 Dezimalstellen

Domänen-Ordner	**Speicherplatz (in GB)**
1_pdf_download	3,78
2_pdf_import	0,01
3_pdf_change	3,79
3b_pdf_error	0,01
4_txt_convert	0,55
5_txt_import	0,00
6_txt_change	0,55
6b_txt_error	0,00
7_lda_algorithm	4,54
Summe	**13,23**

4.2 Verarbeitung

Vor der Analyse der Daten durch LDA soll die Aufbereitung der Dokumente detaillierter betrachtet werden. Die wichtigsten Konferenzen der Wirtschaftsinformatik, die in der AIS organisiert sind, bieten eine gemeinsame Online-Bibliothek zum Herunterladen ihrer Beiträge an. Es handelt sich um ca. 13700 Dokumente, die zum Zeitpunkt der Erstellung der Thesis verfügbar sind. Eine detaillierte Beschreibung der Daten ist in Kapitel 3.1 zu finden.

Die Webadressen der Unterseiten der Konferenzen je Jahr in der AISeL werden manuell in der Datei „download_links.csv“ eingefügt. In jeder neuen Zeile steht eine Webadresse zusammen mit der dazugehörigen Konferenz und dem Jahr. So kommen insgesamt 202 verschiedene Webadressen für 6 Konferenzen über 19 Jahre zusammen.

Insgesamt werden durch den Einsatz des Crawlers „DownThemAll“[167] 13689 PDF-Dokumente auf den Seiten der AISeL identifiziert. Der in SIERRA implementierte Crawler hat 13671 PDF-Dateien erkannt. Die Differenz von 18 Dateien teilen sich wie folgt auf: 8 Dateien sind auf den Seiten der AISeL falsch verlinkt[168] und 10 Dateien werden von SIERRA nicht identifiziert. Nach einem Abgleich der Dateien werden die 10 fehlenden Dateien manuell heruntergeladen

[167] Siehe Kapitel 3.2.2.1.

[168] Es wird der Fehler „404“ angezeigt. Dieser ist als „toter Link“ definiert. Vgl. World Wide Web Consortium (1999).

und in den Datenbestand importiert, so dass insgesamt 13681 PDF-Dateien vorhanden sind. Für die Analyse aller Webadressen und das Herunterladen benötigt SIERRA aus dem Netz der Goethe-Universität ca. 33 Stunden.

Für die Konvertierung der PDF-Dateien zu TXT-Dateien werden 3 Stunden benötigt. Insgesamt können 13660 Dateien konvertiert werden. Die Differenz zum vorherigen Prozess beträgt 21 Dateien. Von denen sind 18 Dateien korrupt, d.h. sie sind beschädigt. 2 Dateien sind bereits auf den Seiten der AISeL korrupt, bei den restlichen 16 Dateien sind Schäden während der Übertragung oder Speicherung aufgetreten. 3 Dateien sind gescannte Dateien und können nicht ausgelesen werden. Die beschädigten 16 Dateien werden anhand der in den XML-Dateien gespeicherten Links manuell heruntergeladen, dem Datenbestand manuell hinzugefügt und durch ein erneutes Ausführen des Konvertierungsprozesses in Text-Dateien umgewandelt.

Durch das Konsolidieren der TXT-Dateien in einer CSV-Datei werden ausschließlich englischsprachige Dokumente verarbeitet. Insgesamt werden dadurch 156 Dateien aussortiert. Die CSV-Datei beinhaltet insgesamt 13520 Dateien, die von LDA weiter verarbeitet werden. Der Prozess benötigt ca. 5 Minuten.

Die nachfolgenden LDA-Prozesse werden durch TMT ausgeführt und im nächsten Unterkapitel detaillierter vorgestellt. Die Gesamtlaufzeit aller Prozesse beträgt ca. 150 Stunden und ist in Tabelle 15 aufgeschlüsselt. Es ist zu beachten, dass die letzten beiden Prozesse parallel ausgeführt werden können, da der Prozessor nicht voll ausgelastet ist. Es empfiehlt sich für diesen Zweck SIERRA zweimal zu öffnen.

Tabelle 15 - Gesamtlaufzeit alle Prozesse

Prozess	**Laufzeit in Stunden**	**Hinweise**
Download	33	
Dateien zusammenführen	0	Dauert ca. 5 Minuten.
Konvertieren	3	
Konsolidieren	0	Dauert ca. 10 Minuten.
CSV erstellen	0	Dauert ca. 5 Minuten.
LDA Topic-Berechnung	57	
LDA Modellberechnung	34	
LDA anwenden	23	Die beiden Prozesse laufen parallel.
LDA differenzieren	23	
Summe	**150**	

4.3 Anwendung von Latent Dirichlet Allocation

Für die Ausführung von LDA sind Entscheidungen für einzelne Einstellungen zu treffen[169]. Diese decken die Anzahl an Topics, die Wahl der Hyperparameter und die Anzahl der Iterationen ab. Für die Wahl dieser Werte wurden bereits Empfehlungen abgegeben, optimale Einstellungen gibt es jedoch oft nicht. In den nachfolgenden Unterkapiteln werden die Entscheidungen der einzelnen Werte dargelegt und erläutert.

Durch die zeitintensive Berechnung der Ergebnisse und die darauffolgenden manuellen Auswertungen beruhen diese in den folgenden Unterkapiteln auf einem früher erstellten Modell mit 13415 Dokumenten. Erst bei der Fertigstellung der Auswertungen sind unvorhergesehene Fehler in SIERRA identifiziert worden. Die Ergebnisse wurden auf dem neuen Datenbestand von 13520 Dokumenten neu berechnet und mit den vorherigen Ergebnissen inhaltlich verglichen. Der ursprüngliche Korpus ist damit um 0,008 % gewachsen. Es zeigen sich nur sehr geringe Abweichungen, die sich nicht auf die gewonnenen Erkenntnisse auswirken. Die Topics besitzen in dem Fall andere Nummern, lassen sich inhaltlich jedoch eindeutig zuordnen. Die Fehler in SIERRA konnten behoben werden.

[169] Siehe Kapitel 2.3.1.

4.3.1 Bestimmung der Anzahl an Topics

Vor der Erstellung eines Topic-Modells muss eine Anzahl an Topics gewählt werden[170]. Da ein Optimum nicht direkt berechnet werden kann, ist es nötig, Topic-Modelle für verschiedene Anzahlen an Topics zu erstellen, um sie danach zu testen. Ein Maß zum Testen ist der Perplexity-Wert[171]. Für die Berechnung wird dieser Wert in 30er-Schritten zwischen 30 und 300 Topics getestet und ist in Abbildung 10 zu sehen. Obwohl kein Minimum aufgrund des begrenzten Arbeitsspeichers erreicht werden konnte, ist ein deutliches Konvergieren der Kurve zu erkennen. Ein abnehmender und konvergierender Perplexity-Wert reicht für die Wahl einer Topic-Anzahl aus[172]. Für diese Berechnung werden 500 Iterationen eingestellt, da sich das Modell über die Iterationen stabilisiert. Für beide Hyperparameter[173] wird jeweils der Wert 0,1 gewählt.

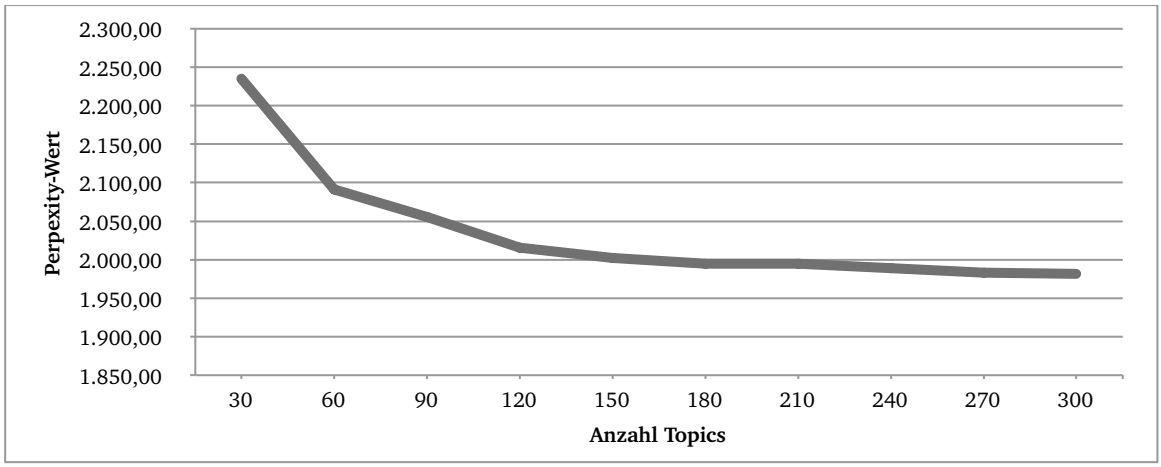


Abbildung 10 - Perplexity-Werte zur Berechnung der Topics in 30er-Schritten. Getestet von 30 bis 300 Topics.

Dieser Vorgang hat insgesamt 57 Stunden gedauert. Es wird die Entscheidung getroffen, dass für weitere Berechnungen 150 Topics als Parameter genutzt werden sollen, da die Kurve ab diesem Punkt abflacht und zu konvergieren scheint.

---

170 Siehe Kapitel 2.3.1.

171 Siehe Kapitel 2.3.2.

172 Siehe Kapitel 2.3.2.

173 Siehe Kapitel 2.1.2.

4.3.2 Berechnung des Topic-Modells

Mit der Auswahl einer Topic-Anzahl kann das Topic-Modell berechnet werden, auf dessen Grundlage weitere Berechnungen erfolgen können. Für die Hyperparameter[174] wird wie bei der Berechnung der Topic-Anzahl jeweils der Wert 0,1 gewählt. Für die Anzahl an Iterationen wird 1500 eingestellt. Alle 50 Iterationen berechnet TMT einen Log-Likelihood-Schätzer und speichert diesen in der Datei „log-probability-estimate.txt“. Mit dem Schätzer kann berechnet werden, ob bereits genügend Iterationen für das Modell durchgeführt wurden. In Abbildung 11 sind die einzelnen Schätzer-Werte in einem Diagramm im Bereich von 0 bis 250 Iterationen abgetragen, da die Werte ab diesem Punkt konvergieren. Deutlich zu erkennen ist ein abflachen der Kurve ab Iteration 100 mit einer folgenden Konvergenz um den Wert -275.000.000. Dies deutet darauf hin, dass 1500 Iterationen für die Berechnung des Modells nicht nötig sind.

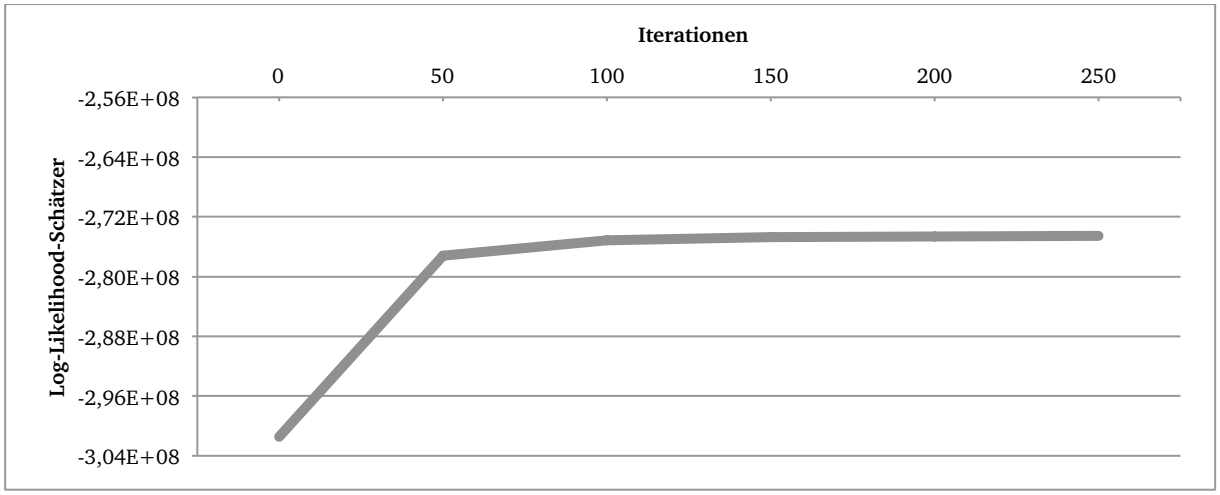


Abbildung 11 - Log-Likelihood-Schätzer im Bereich von 0 bis 250 Iterationen

Für einen Überblick über die Veränderungen einzelner Topics zwischen den Iterationen sind in Tabelle 16 zwei Topics als Beispiele dargestellt. Je Topic sind die 10 relevantesten Wörter über die Iterationen 0, 100, 300 und 1500 aufgeführt. In Iteration 0 beginnt LDA mit einer Ansammlung an zufällig ausgewählten Wörtern. Erst über die Iterationen bilden sich Topics heraus, die eventuell einem übergeordneten Thema zugeordnet werden können. Für Topic

[174] Siehe Kapitel 2.3.1.

065 könnte „Mobile Technologien“ und für Topic 070 „Finanzmärkte“ als Oberbegriff gewählt werden. Diese Zusammensetzung der Topics ist je Iteration in der Datei „summary.txt“ zu finden.

Tabelle 16 - Zwei beispielhafte Topics über mehrere Iterationen

Topic / Iteration	0	100	300	1500
Topic 065	inform thi research manag model process technolog studi develop busi	mobil devic wireless phone technolog applic locat comput access person	mobil devic wireless phone technolog applic comput locat access network	mobil devic wireless phone technolog applic comput locat network access
Topic 070	inform thi research manag model process studi develop technolog busi	cost order trade resourc thi optim benchmark time asp alloc	cost trade market stock order price transact financi benchmark investor	market trade stock event announc price investor return order financi

Über Iteration 100 hinaus stabilisieren sich die einzelnen Wörter in ihren Positionen. Eine Bewertung des Modells könnte in diesem Beispiel aber bereits bei 100 Topics durchgeführt werden. Bei einer Laufzeit von 34 Stunden über 1500 Iterationen könnte die Berechnung mit weniger Iterationen weit schneller durchgeführt werden.

4.3.3 Anwendung des Modells

Nachdem das Modell erstellt wurde, kann es auf Daten angewendet werden, um die Topic-Struktur zu ermitteln. Im Fall dieses Beispiels wird das Modell auf den gleichen Korpus angewendet. Es besteht die Möglichkeit, die Prozesse (4.c) und (4.d) auszuführen, um Analysen über das Modell und dessen Einschränkung nach einem weiteren Merkmal zu erstellen. Für dieses Beispiel wird eine Differenzierung der Daten nach dem Jahr der

Konferenz vorgenommen. Die beiden Prozesse für die Erstellung der Auswertungen werden parallel gestartet, da sie den Prozessor und den Arbeitsspeicher nur zum Teil belasten. Je Prozess werden dabei 23 Stunden bis zur Fertigstellung benötigt. Die Ergebnisse und wie diese zu interpretieren sind werden im nachfolgenden Kapitel diskutiert.

4.4 Interpretation der Ergebnisse

Die für eine ausführliche Analyse nötigen Daten sind erstellt und können interpretiert werden. Dafür stehen die in Kapitel 3.3.5 vorgestellten Dateien zur Verfügung. Anhand der Dateien werden mehrere Auswertungen darüber erstellt, wie die Daten interpretiert und für weitere Analysen genutzt werden können. Das Szenario besteht aus der vorgestellten Identifikation inhaltlich ähnlicher Dokumente und die Visualisierung von Topics und der Topic-Struktur einzelner Dokumente.[175] Durch die Funktionen von TMT bietet sich auch die Erstellung von Zeitreihen für Topics und deren Relevanz an.

4.4.1 Visualisierung von Topics

Eine Analyse über die von LDA im Korpus entdeckten Themen gestaltet sich anhand der von TMT erstellten Dateien relativ einfach. Die Datei „converted_file-top-terms.csv“ wurde bereits in Kapitel 3.2.4.4 vorgestellt und bietet einen Überblick über die 50 wichtigsten Wörter je Topic in Tabellenform. Ein Topic stellt dabei eine Menge von Wörtern dar. Nachfolgend sind die 10 wichtigsten Wörter von 2 Topics beispielhaft aus dem berechneten Modell zu sehen:

{ mobil, devic, wireless, phone, technolog, applic, comput, locat, network, access }

{ market, trade, stock, event, announc, price, investor, return, order, financi }

Die Topics weisen inhaltlich einen starken Zusammenhang auf und können subjektiv interpretiert werden. Das erste Topic handelt von mobilen Technologien. Das zweite Topic befasst sich mit Finanzmärkten. Eine Auflistung aller Topics ist in Anhang B zu finden.

[175] Siehe Kapitel 2.

4.4.2 Visualisierung und Berechnung der Ähnlichkeit von Dokumenten

Für die Identifikation ähnlicher Dokumente wurde in Kapitel 2.3.2 bereits eine Formel definiert. Dieser Wert kann immer nur zwischen zwei Dokumenten berechnet werden. Für eine Berechnung ist zuerst die Tabelle „converted_file-document-topic-distributions.csv“ zu öffnen. Diese enthält für jedes Dokument die Topic-Verteilung. Für ein Beispiel wird die Tabelle nach Topic 119 absteigend sortiert, um Dokumente mit einem starken Bezug zu diesem Topic zu identifizieren. Danach werden die Werte der ersten 10 Dokumente in einer neuen Tabelle mit der genannten Formel berechnet. Das Ergebnis ist eine Matrix, die in Tabelle 17 abgebildet ist. Die Spalten und Reihenüberschriften bestehen aus der IDs der jeweiligen Dokumente. Die markierten Zellen stellen den höchsten und den niedrigsten Wert in der Matrix dar.

Tabelle 17 - Distanzen zwischen mehreren Dokumenten. Die höchste und die niedrigste Distanz wurden markiert.

	10703625	**11406665**	**11768857**	**12589828**	**15217749**	**15378580**	**15681402**	**16534282**	**18672055**	**18908704**
10703625	0									
11406665	0,63	0								
11768857	0,49	0,74	0							
12589828	0,58	0,81	0,38	0						
15217749	0,64	0,88	0,53	0,43	0					
15378580	0,60	0,90	0,39	0,47	0,69	0				
15681402	0,39	0,87	0,40	0,48	0,41	0,61	0			
16534282	0,53	0,68	0,54	0,65	0,65	0,56	0,48	0		
18672055	0,78	0,68	0,55	0,49	0,62	0,48	0,71	0,41	0	
18908704	0,55	1,05	0,36	0,60	0,41	0,54	0,32	0,65	0,66	0

An den markierten Zellen sind drei Dokumente beteiligt, die nachfolgend mit ihrem Titel aufgeführt werden:

- 11406665: Profile of the Social Network in Photo Sharing Systems[176]

[176] Kazienko et al. (2008).

- 15681402: Using Online Social Networking to Enhance Social Connectedness and Social Support for the Elderly[177]
- 18908704: Social media, social capital, and seniors: The impact of Facebook on bonding and bridging social capital of individuals over 65.[178]

Die beiden zuletzt genannten Dokumente weisen die geringste Distanz und damit höchste Ähnlichkeit auf. Wie schon die Titel vermuten lassen, handelt es sich um Publikationen, die sich mit sozialen Netzwerken und älteren Individuen auseinandersetzen. Der größte Anteil ihrer Ähnlichkeit kommt neben der Zuordnung zu Topic 119 aus Topic 60.

Die Dokumente 11406665 und 18908704 haben ausschließlich die Zuordnung zu Topic 119 gemein und unterscheiden sich in den restlichen Zugehörigkeiten sehr. Das Dokument 11406665 weist eine Nähe zu Topic 16 auf, das ebenfalls den sozialen Netzwerken zugeordnet werden kann. Die genannten Topics sind zur Übersichtlichkeit nochmal nachfolgend aufgeführt:

Tabelle 18 - Topics im Zusammenhang mit sozialen Netzwerken. Herleitung aus der Ähnlichkeit von Dokumenten.

Topic 016	{ network, structur, node, social, central, connect, link, analysi, individu, degre }
Topic 060	{ commun, member, onlin, particip, social, discuss, forum, post, support, virtual }
Topic 119	{ social, capit, network, blog, facebook, friend, onlin, peopl, post, twitter }

Bei der Untersuchung weiterer Topics wurde die Erfahrung gemacht, dass bei der Ähnlichkeit von Dokumenten immer ein Blick in die Dokumente geworfen werden sollte. In den Beiträgen der Konferenzen sind auch Zusammenfassungen und Ankündigungen von Präsentationen enthalten. Teilweise auch Agenden von Arbeitsgruppen. Diese Dokumente waren alle thematisch sehr ähnlich. Es handelt sich nur nicht um wissenschaftliche Publikationen. Das Ergebnis zeigt, dass anhand des Ähnlichkeitsmaßes thematisch ähnliche Dokumente identifiziert werden können. Jedoch sollte auch bedacht werden, dass gerade bei großen Datenmengen ein zweiter Blick nötig ist. Die Zahlen allein können täuschen.

[177] Goswami et al. (2008).

[178] Erickson (2011).

Rein experimentell wurde in der gesamten Dokument-Topic-Verteilung das Ähnlichkeitsmaß nicht für die Distanz von Dokumenten, sondern für Topics angewendet. Die Ergebnisse haben jedoch gezeigt, dass sehr allgemeine Topics – die unter anderem viele Stoppwörter enthalten – und sinnlose Topics – diese beinhalten oft nur Jahreszahlen – den subjektiv qualitativ hochwertig empfundenen Topics zugeordnet werden und in Summe die eigentlich zusammenhängenden Topics übertreffen. Auf diese Weise kann das Maß also nicht angewendet werden.

4.4.3 Bestimmung relevanter Topics

Die Bestimmung relevanter Topics gestaltet sich schwierig und kann nicht eindeutig durch ein Verfahren bestimmt werden. Vielmehr muss ein Verfahren definiert werden, dass aus einer subjektiven Betrachtung heraus gute Ergebnisse liefert. Pussep[179] reduziert den Datenbestand nach einer ersten Berechnung durch Aussortieren von Topics, die offensichtlich keinen Sinn ergeben bzw. als nicht relevant erscheinen. Die Topics werden durch die Meinung von Experten in definierte Ober-Themen eingeteilt, soweit diese zutreffend sind. Die übrig gebliebenen Dokumente werden ein zweites Mal bewertet und analysiert. Für die Bewertung der Relevanz soll weniger ein Verfahren, sondern mehr ein Regelwerk entstehen, wie Topics bewertet werden können.

Um die im Korpus enthaltenen relevantesten Topics zu identifizieren, können mehrere Maße herangezogen werden. In der von SIERRA als Bericht erstellbare Datei „summary.csv“[180] werden alle Topics mit ihren Relevanz-Werten[181] und den 20 relevantesten Wörtern je Topic ausgegeben. Nach dem Relevanz-Wert absteigend sortiert erhält man eine Rangreihenfolge von Topics. Genauso gut kann auch die Datei „converted_file-usage.csv“[182] genutzt werden, wie aus Tabelle 19 ersichtlich wird. In dieser sind die absoluten Häufigkeiten von Dokumenten und Wörtern enthalten, die den jeweiligen Topics zuzuordnen sind. Diese können absteigend nach den Häufigkeiten der Dokumente oder Wörter sortiert werden. In Tabelle 19 sind die in allen Sortierungen 10 obersten Topics abgebildet. Es ist zu erkennen,

[179] Pussep et al. (2012). S. 7.

[180] Siehe Kapitel 3.3.5.

[181] Siehe Kapitel 2.3.2.

[182] Siehe Tabelle 10.

dass sich die Rangreihenfolgen nur leicht unterscheiden. Die Spalten der Wörter und der Relevanz sind in diesem Beispiel sogar identisch. Die Relevanz darf nicht mit der Wahrscheinlichkeitsverteilung von Dokumenten und Topics verwechselt werden.[183]

Tabelle 19 - Rangreihenfolgen der Topics nach Relevanz[184] und der absoluten Häufigkeit zugeordneter Dokumente und Wörter[185], absteigend sortiert.

Platzierung	**documents**	**words**	**relevance**
1	Topic 131	Topic 131	Topic 131
2	Topic 101	Topic 101	Topic 101
3	Topic 018	Topic 140	Topic 140
4	Topic 140	Topic 007	Topic 007
5	Topic 007	Topic 033	Topic 033
6	Topic 033	Topic 036	Topic 036
7	Topic 036	Topic 027	Topic 027
8	Topic 027	Topic 080	Topic 080
9	Topic 117	Topic 117	Topic 117
10	Topic 080	Topic 120	Topic 120

Die Berechnung des Modells wurde ohne die Definition eigener Stoppwörter durchgeführt. Wie in Kapitel 2.3.1 bereits erläutert, ist zu erwarten, dass viele Topics diese Wörter enthalten werden und dadurch die Aussagekraft insgesamt sinken wird. In Anlage C ist eine Liste der 50 häufigsten Wörter im Korpus dargestellt, die zugleich auch 20 % der Gesamthäufigkeiten aller Wörter entsprechen. Bei einer absteigenden Sortierung der Topics nach den Häufigkeiten der zugeordneten Wörter ergibt sich eine Matrix, wie sie in Tabelle 20 abgebildet ist. Ist ein Wort in der Matrix auch ein Wort der 50 häufigsten Wörter, ist dieses mit einer Schattierung markiert.

183 Siehe Kapitel 2.3.2.

184 Siehe Kapitel 2.3.2.

185 Siehe Schema der Tabelle 10. Dort ist der Aufbau der Tabelle über die absolute Häufigkeit von Wörtern und Dokumenten je Topic abgebildet.

Tabelle 20 - Auszug aus der Topic-Liste mit den am häufigsten zugeordneten Wörtern, die markiert sind, falls diese zu den 50 häufigsten Wörtern gehören.

Rang	Topic	Word 1	Word 2	Word 3	Word 4	Word 5
1	131	thi	inform	differ	framework	approach
2	101	thi	mai	inform	provid	need
3	140	effect	studi	research	thi	inform
4	007	thei	thi	work	peopl	time
5	033	manag	implement	support	process	inform
6	036	2006	2005	2004	research	2007
7	027	time	number	thi	result	figur
8	080	studi	research	data	interview	thi
9	117	manag	1999	1998	1997	1996
10	120	technolog	social	practic	structur	context
⋮	⋮	⋮	⋮	⋮	⋮	⋮
65	Topic 129	decis	make	decision-mak	support	choic
66	Topic 055	custom	bank	crm	relationship	market
67	Topic 077	group	member	support	process	discuss
68	Topic 086	qualiti	satisfact	expect	dimens	perceiv
69	Topic 014	cultur	differ	nation	dimens	countri
70	Topic 026	secur	comput	attack	threat	polici
71	Topic 010	integr	applic	infrastructur	technolog	support
72	Topic 019	media	commun	messag	email	rich
73	Topic 081	design	artifact	pattern	scienc	principl
74	Topic 023	user	featur	function	end-us	end
75	Topic 039	internet	digit	access	divid	home
76	Topic 051	usabl	visual	interfac	design	imag
77	Topic 148	2009	2010	2008	2011	thirti
78	Topic 052	sourc	open	develop	oss	softwar
79	Topic 053	agent	auction	bid	negoti	price
80	Topic 147	skill	profession	train	job	worker
81	Topic 008	busi	oper	enterpris	unit	custom
82	Topic 006	power	polit	discours	societi	empower
83	Topic 038	review	seller	rate	reput	onlin
84	Topic 089	search	document	queri	retriev	engin
⋮	⋮	⋮	⋮	⋮	⋮	⋮
141	Topic 063	decept	detect	spam	0.0	phish
142	Topic 110	canada	2008	america	toronto	august
143	Topic 099	ninth	2008	pari	twenti	intern
144	Topic 105	stakehold	polici	broadband	vote	nonprofit
145	Topic 113	interact	convers	involv	mediat	human
146	Topic 115	client	consult	advisor	advic	advisori
147	Topic 128	flow	question	answer	experi	ask
148	Topic 041	page	european	18th	fig	delet
149	Topic 062	reus	taxonomi	repositori	asset	domain
150	Topic 084	intranet	sensemak	weick	extranet	corpor

Wie in der Tabelle deutlich zu sehen ist, nimmt der Anteil der häufigsten Wörter mit abnehmender Platzierung der Topics ab. Bei den ersten 10 Topics sind viele Wörter markiert. Diese Topics haben der subjektiven Wahrnehmung nach wenig Aussagekraft und sind nicht zu gebrauchen. Genauso die letzten Topics und weitere über die Tabelle verteilt. Diese besitzen oft Begriffe, die in den Kopf- und Fußzeilen verwendet werden und keine besonderen Zusammenhänge aufweisen. Die Relevanz-Werte in der „summary.csv“ sind bei diesen Topics relativ niedrig. Insgesamt können anhand dieser Betrachtung 31 Topics aussortiert werden.

Ebenfalls in Tabelle 20 abgebildet ist ein Auszug aus den mittleren Platzierungen. Diese besitzen im Vergleich zu den ersten 10 Topics wenige der 50 häufigsten Wörter. Bei subjektiver Betrachtung der Wörter machen die meisten dieser Topics Sinn und könnten detaillierter untersucht werden.

Tabelle 21 – Die 10 relevantesten Topics mit den häufigsten Dokumenten nach Aussortieren nicht-relevanter Topics.

Rang	**Topic**	**Words**
11	002	{ compani, market, industri, product, new, busi, technolog, competit, strategi, corpor }
12	092	{ perceiv, intent, technolog, accept, model, usag, behavior, attitud, davi, eas }
13	000	{ construct, measur, item, valid, model, indic, reliabl, thi, research, structur }
14	013	{ thi, access, server, databas, data, applic, file, provid, architectur, request }
15	090	{ knowledg, manag, transfer, learn, process, share, tacit, creation, practic, individu }
16	100	{ survei, respond, tabl, result, studi, thi, differ, respons, mean, signific }
17	138	{ price, market, cost, thi, econom, profit, consum, product, optim, good }
18	130	{ learn, student, educ, cours, technolog, e-learn, onlin, teach, instructor, environ }
19	145	{ student, program, cours, univers, educ, faculti, curriculum, teach, school, academ }
20	034	{ inform, technolog, provid, confer, need, access, sourc, avail, electron, seek }

In Tabelle 21 sind nach Aussortieren die 10 Topics mit den höchsten Dokumentenhäufigkeiten zu sehen. Dies sind die im Korpus am häufigsten diskutierten Themen. Bei einer genauen Betrachtung der Wörter stehen über die 19 Jahre besonders Themen im Vordergrund, die sich mit Ökonomie, Modellen, Datenhaltung, Bildung und Informationstechnik beschäftigen.

In allen Topics fallen besonders die Jahres- und andere Zahlen auf, die keine besondere Aussagekraft besitzen. Es wird deutlich, dass ein Modell unter Ausschluss von häufig auftretenden Wörtern und bestimmten Zahlen berechnet werden sollte, um möglicherweise bessere Ergebnisse zu erhalten. Unter Umständen reicht dann eine niedrigere Anzahl an

Topics. Des Weiteren scheinen gerade Topics „aus der Mitte“ der Rangreihenfolge Sinn zu ergeben.

4.4.4 Topics im zeitlichen Verlauf

Besonders die Veränderung bestimmter Themen über einen zeitlichen Verlauf erscheinen interessant. So lassen sich Topics über einen längeren Zeitraum in ihrer Entwicklung beobachten. Es kann bewertet werden, ob das Interesse der Autoren an bestimmten Themen in einem Korpus konstant, zu- oder abnehmend ist oder gar nur einen kurzweiligen Hype darstellt. Die Veränderung diskutierter Themen von Konferenzen soll zuerst anhand ausgewählter Beispiele gezeigt werden. Das Beispiel zeigt außerdem, wie Topics zusammengefasst werden können. Danach werden Themen gezeigt, die im zeitlichen Verlauf ein besonders starkes Wachstum bzw. starken Verlust aufweisen.

Eine Möglichkeit zur Analyse stellt die Beobachtung einzelner Topics dar. In Anhang D sind zwei Beispiele für die Veränderungen der zugeordneten Wörter innerhalb eines Topics über die Jahre 1993 bis 2011 aufbereitet. Das eine Topic beschäftigt sich weitgehend mit Sicherheitsthemen. Während in den neunziger Jahren der Fokus eher auf Firewalls, Passwörtern und Eindringlingen liegt, gewinnen ab dem Jahr 2000 Wörter an Bedeutung, die sich mit Schwachstellen, Attacken und Bedrohungen beschäftigen. Ein weiteres Topic zeigt die Veränderung eines Topics von den klassischen Print-Medien über Web 2.0 hin zu social media, dass sich mit „Content“ umschreiben ließe.

Die Betrachtung von quantitativen Veränderungen von Topics über mehrere Jahre soll beispielhaft für das Thema „Internet“ vorgenommen werden. Dafür wird die Datei „summary.csv“ nach potenziell passenden Topics durchsucht. Zur Validierung der Topics dient die „converted_file-top-terms.csv“, in der die wichtigsten 50 Wörter je Topic aufgeführt sind. Nach der Identifikation von 10 relevanten Topics erhalten sie ein Label als Oberbegriff, um thematisch verwandte Topics zusammenzufassen. Dies ist in Tabelle 22 zu sehen.

Tabelle 22 - Ausgewählte Topics zum Thema "Internet"

Label	Topics	Wörter				
collaboration	Topic 019	media	commun	messag	email	rich
	Topic 043	content	2.0	wiki	web	media
	Topic 142	web	site	websit	page	content
eCommerce	Topic 024	electron	e-commerc	sme	commerc	e-busi
	Topic 068	consum	product	onlin	purchas	shop
connectivity	Topic 039	internet	digit	access	divid	home
	Topic 065	mobil	devic	wireless	phone	technolog
social web	Topic 060	commun	member	onlin	particip	social
	Topic 119	social	capit	network	blog	facebook
cloud computing	Topic 064	comput	cloud	resourc	grid	peer

Über die Datei „converted_file-sliced-usage.csv“ kann ein zeitlicher Verlauf leicht erstellt werden. Dafür dient die Spalte „Documents“. Sie enthält die zugeordneten Dokumente je Topic und Jahr. Da es sich um absolute Häufigkeiten handelt, können diese Werte einfach addiert werden. Abbildung 12 zeigt den Verlauf der Labels von 1993 bis 2011. Deutlich zu erkennen ist der stetige Anstieg an Ausarbeitungen über soziale Netzwerke. Die anderen Labels nehmen ebenfalls zu, verzeichnen aber keine so starken Anstiege. Bei einer Bereinigung der Daten um das allgemeine Datenwachstum der analysierten Dateien verzeichnen die Internet-Themen immer noch einen positiven Anstieg. Der relative Anteil an Internet-Themen am Bestand von Veröffentlichungen nimmt folglich zu.

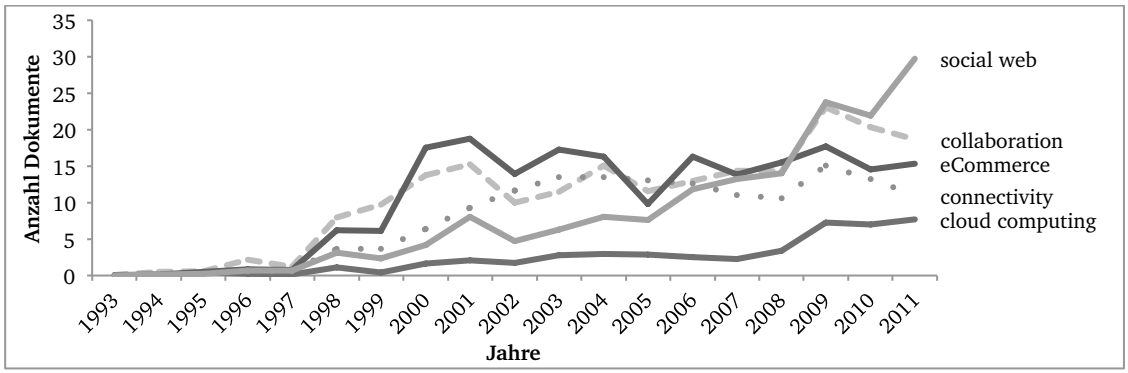


Abbildung 12 - Anzahl Dokumente für ausgewählte Topics von 1993 bis 2011

Im Bereich der IS können sich Rahmenbedingungen über wenige Jahre ändern und dadurch neue Anwendungszwecke entstehen lassen.[186] Bei der Erstellung einer Zeitreihe über die Entwicklungen in der Wirtschaftsinformatik sollen daher unterschiedliche Zeiträume untersucht werden. Möglicherweise gibt es oft diskutierte Themen, die erst vor 5 Jahren aufgekommen sind, aber vor 10 Jahren noch nicht diskutiert wurden. Insgesamt stehen für die Untersuchungen Dateien aus 19 Jahren von 1993 bis 2011 zur Verfügung. Vor dem Jahr 2000 sind pro Jahr nur Beiträge von 2 Konferenzen im Datenbestand. Insgesamt machen diese Konferenzen mit 1465 Dokumenten 10 % vom Korpus aus. Ab dem Jahr 2000 sind regelmäßige Beiträge von mindestens 4 der 6 Konferenzen enthalten. Daher soll die Untersuchung frühestens ab dem Jahr 2000 erfolgen.

Zur Abbildung mittel- und langfristiger Entwicklungen in der Wirtschaftsinformatik wird der Verlauf der letzten 5 bzw. 10 Jahre betrachtet. Dazu wird die absolute Häufigkeit an Dokumenten je Topic im Jahr 2001 bzw. 2006 als Basisjahr aus der Tabelle „converted_file-sliced-usage.csv“ genommen. Die Basiswerte werden mit denen aus dem Jahr 2011 verglichen. Die 5 Topics mit dem stärksten Wachstum bzw. höchsten Verlust werden in jeweils einem Diagramm dargestellt. Für eine inhaltliche Einordnung der Topics sind diese unter dem Diagramm abgebildet.

In Abbildung 13 und Abbildung 14 sind die Entwicklungen der letzten 5 Jahre von 2006 bis 2011 zu sehen. Das stärkste Wachstum zeigen die Topics 119 und 76. Diese befassen sich inhaltlich mit sozialen Netzwerken und der Ökologie von Technik.

Im Rahmen der Wahrnehmung aktueller Nachrichten sind diese Entwicklungen durchaus plausibel. So sind soziale Netzwerke in den letzten Jahren sehr populär geworden und werden im Korpus von vielen unterschiedlichen Blickwinkeln betrachtet.

Die Reduzierung der Auswirkungen von Hard- und Software auf die Umwelt wird oft als „Green IT“[187] bezeichnet. Insbesondere in den letzten Jahren werden die Bemühungen in diesem Bereich verstärkt, auch zur Besserung des Image eines Unternehmens. Im Jahr 2008 hatte sich die CeBit dem Thema gewidmet, wenn auch nur oberflächlich.[188]

---

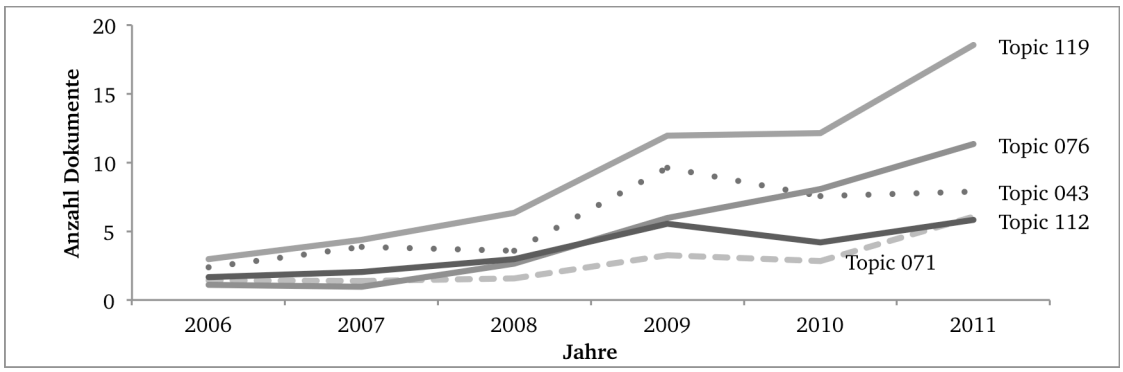
186 Siehe Kapitel 1.2.

187 Vgl. Watson et al. (2008), S. 2.

188 Vgl. Kremp (2008).

Weitere Themen der letzten 5 Jahre beschäftigen sich mit dem oft genannten Begriff „Web 2.0“, der Medizintechnik und der „Crowd“. Dabei ist zu beobachten, dass die Topics 43 und 112 bereits wieder leicht abgenommen haben. Dies ist möglicherweise darin begründet, dass der Begriff „Web 2.0“ eher ein Hype war und inzwischen eher mit „Social Media“ beschrieben werden kann.[189] Das Interesse an „Crowd“-Themen scheint stetig zu wachsen. Speziell das Crowdsourcing beschäftigt sich mit der „[...] Auslagerung von Unternehmensaufgaben und –strukturen [...] auf die Intelligenz einer Masse von Freizeitarbeitern im Internet“[190].

Die höchsten Verluste der letzten 5 Jahre kennzeichnen ein abnehmendes Interesse an Themen wie Outsourcing, Open-Source, Usability, E-Business und dem XML-Format. Dies könnte darauf zurückgeführt werden, dass diese Themen zum einen nicht mehr relativ jung sind und zum anderen bereits viel diskutiert wurden, so dass sie in der Wirtschaft als normal angesehen werden. Beispielsweise sind in Unternehmen Open-Source, XML und Outsourcing inzwischen weit verbreitete Themen. [191]



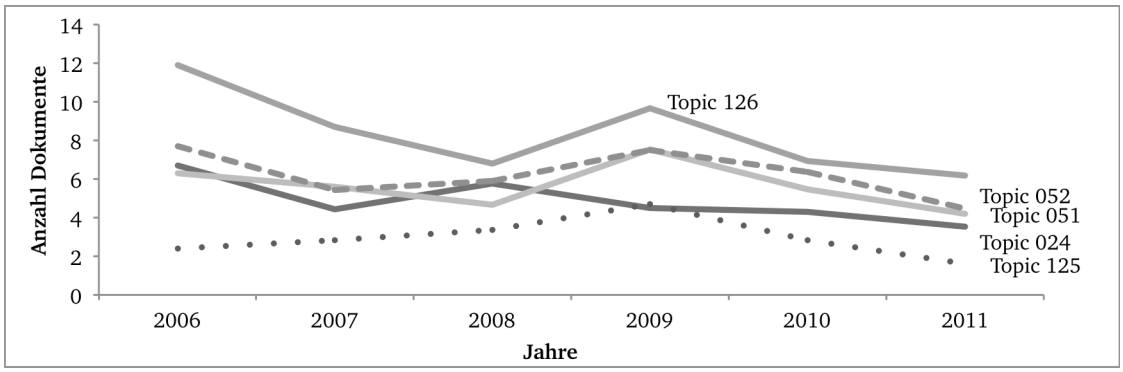
119	76	43	112	71
social	green	content	medic	creativ
capit	environment	2.0	physician	idea
network	sustain	wiki	patient	gener
blog	energi	web	emr	product
facebook	consumpt	media	hospit	contest
friend	electr	new	drug	crowd
onlin	practic	wikipedia	clinic	crowdsourc
peopl	2008	technolog	error	solver
post	ecolog	newspap	record	innov
twitter	initi	video	care	collect

Abbildung 13 - 5 Topics mit den stärksten Zuwächsen 2006-2011. Darunter sind die 10 relevantesten Wörter je Topic abgebildet.

189 Vgl. Schürig (2010).

190 Vgl. Wikipedia (o. J.): Crowdsourcing.

191 Meinung des Autors.



126	52	51	24	125
outsourc	sourc	usabl	electron	standard
vendor	open	visual	e-commerc	xml
contract	develop	interfac	sme	document
offshor	oss	design	commerc	interoper
relationship	softwar	imag	e-busi	element
manag	code	displai	marketplac	xbrl
cost	project	screen	b2b	format
client	linux	color	market	exchang
servic	contribut	present	small	data
sourc	licens	navig	busi	schema

Abbildung 14 - 5 Topics mit den höchsten Verlusten 2006-2011. Darunter sind die 10 relevantesten Wörter je Topic abgebildet.

Bei einer Betrachtung der letzten 10 Jahre verändern sich die Themen leicht. In Abbildung 15 sind analog zu Abbildung 13 die Entwicklungen der letzten 10 Jahre festgehalten. Einzig das Topic 135 taucht in dieser Grafik als neu auf. Die Grafik zeigt eine erhöhte Veröffentlichungsanzahl für Themen der agilen Software-Entwicklung.

Die Themen mit den höchsten Verlusten der letzten 10 Jahre zeigen in Abbildung 16 größtenteils andere Themen als in Abbildung 14. Es sind Themen, die sich speziell mit Internet-Technologien auseinandersetzen. Topic 54 besteht aus Begriffen, die rein subjektiv mit der Verkaufsplattform „eBay“ in Verbindung gebracht werden. Die Anzahl der Veröffentlichungen scheint bei diesem Thema zu schwanken. Topic 199 könnte mit Datenbanken in Verbindung gebracht werden. Das Interesse an diesem Thema ist wichtig, da bei den steigenden Datenmengen[192] die Datenhaltung Schritt halten muss. Vor 10 Jahren scheint das Interesse aber leicht größer gewesen zu sein. Topic 142 und 39 nennen Begriffe

[192] Siehe 3.1.3.

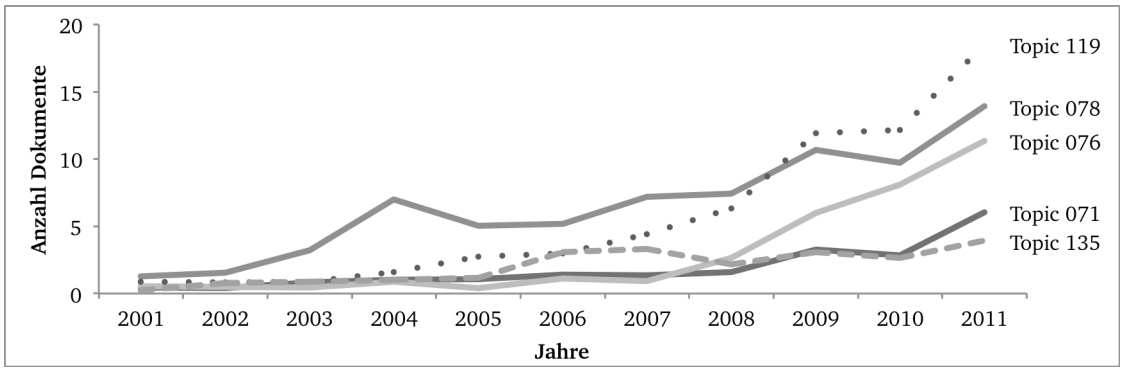
zum Thema Internet, die im normalen Sprachgebrauch inzwischen seltener verwendet werden.[193]

Die Betrachtung der Auswertungen lässt Rückschlüsse auf eine mögliche Weiterentwicklung zu. Das Thema Internet kann mit vielen Topics assoziiert werden. Selbst im Bereich der Sicherheit verändert sich das Thema selbst von der Bedrohung durch Schädlinge zu der Bedrohung durch Angriffe aus dem Netz. Das Internet hat kein eigenes Topic, das als Internet-Topic beschrieben werden könnte. Es untergliedert sich in viele verschiedene Topics und ist Teil von vielen Topics. Nicht nur die physische, sondern auch die soziale Vernetzung findet stärkeres Interesse. In den wissenschaftlichen Publikationen setzen sich unterschiedliche Themengebiete mit der weltweiten Vernetzung auseinander.

Die Ziele dieser Thesis konnten erfüllt werden. Mit Unterstützung einer Software können Trends in der Software-Industrie und häufig diskutierte Problemstellungen identifiziert werden. Damit ist es möglich, die Entwicklungen der wichtigsten IS-Konferenzen zu überblicken. Interessanterweise sind die Trends sehr ähnlich zu den Ergebnissen von Pussep[194].

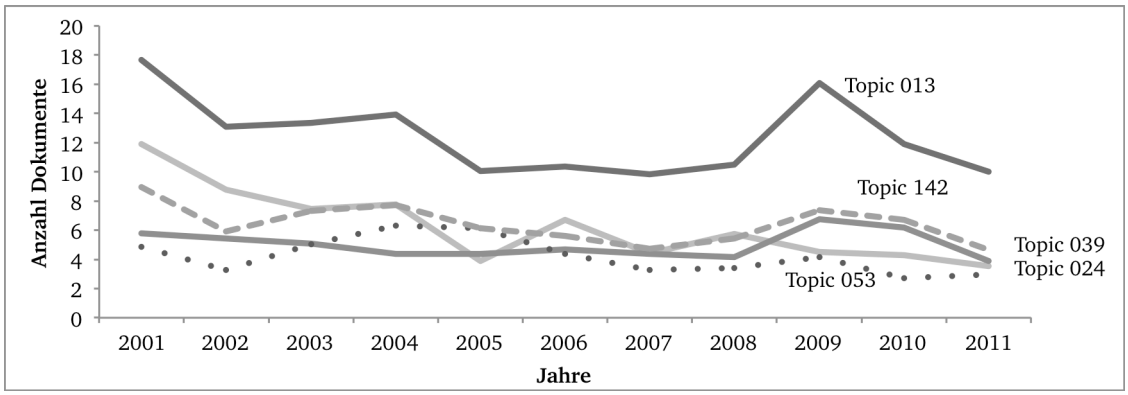
[193] Meinung des Autors.

[194] Pussep et al. (2012), S. 11 f.



119	78	76	71	135
social	health	green	creativ	agil
capit	patient	environment	idea	develop
network	healthcar	sustain	gener	improvis
blog	care	energi	product	practic
facebook	hospit	consumpt	contest	team
friend	medic	electr	crowd	custom
onlin	clinic	practic	crowdsourc	method
peopl	nurs	2008	solver	program
post	record	ecolog	innov	pair
twitter	provid	initi	collect	scrum

Abbildung 15 - 5 Topics mit den stärksten Zuwächsen 2001-2011. Darunter sind die 10 relevantesten Wörter je Topic abgebildet.



13	142	39	24	53
thi	web	internet	electron	agent
access	site	digit	e-commerc	auction
server	websit	access	sme	bid
databas	page	divid	commerc	negoti
data	content	home	e-busi	price
applic	web-bas	isp	marketplac	buyer
file	internet	technolog	b2b	bidder
provid	design	connect	market	seller
architectur	navig	2000	small	item
request	link	provid	busi	mechan

Abbildung 16 - 5 Topics mit den höchsten Verlusten 2001-2011. Darunter sind die 10 relevantesten Wörter je Topic abgebildet.

5 Fazit

Das quantitative Wachstum wissenschaftlicher Publikationen im Bereich von IS-Konferenzen steigert die Komplexität der Forschungsgebiete. Den Überblick zu behalten wird zu einer Herausforderung. Die Analyse von Entwicklungen über einen bestimmten Zeitraum im IS-Bereich ist nur mit einem kaum vertretbaren Aufwand zu erstellen. Eine Software könnte diesen Prozess automatisch übernehmen. Das Ziel ist eine Software, die den Benutzer bei der Identifikation von Trends in der Software-Industrie und häufig diskutierten Problemstellungen weitestgehend unterstützt.

Für eine inhaltliche Einordnung der Publikationen bieten sich Topic Modelle an. Sie unterstellen jedem Text eine verborgene Struktur, die sich aus unterschiedlichen behandelten Themen zusammensetzt. Dieses Modell bildet die Grundlage von Latent Dirichlet Allocation (LDA). Anhand verschiedener Approximationsverfahren kann der Algorithmus Themen identifizieren, die in unterschiedlichen Texten behandelt werden. So können thematische Entwicklungen des zu untersuchenden Datenbestandes dargestellt werden. Des Weiteren können diskutierte Themen identifiziert, Themen visualisiert und Ähnlichkeiten von Dokumenten bestimmt werden. Verschiedene Gütemaße erlauben eine qualitative Bewertung von Topic Modellen.

Eine Software, die dieses Verfahren auf wissenschaftliche Publikationen anwenden soll, benötigt klar definierte Prozesse. Diese stellen sicher, dass der Verlauf einer Auswertung nachvollziehbar ist. So können die einzelnen Datenverarbeitungsschritte protokolliert werden. Die im Rahmen dieser Thesis entwickelte Software „SIERRA“ implementiert definierte Prozesse und unterstützt den Benutzer vom automatischen Herunterladen der Publikationen über verschiedene Verarbeitungsschritte hin zur Generierung von Ergebnissen durch die Anwendung von LDA.

Insgesamt werden von SIERRA innerhalb von 150 Stunden ca. 13.500 Dokumente verarbeitet. Die Ergebnisse können mit einem Tabellen-Kalkulationsprogramm weiter verarbeitet werden. Mittels verschiedener manueller Analysen konnten wertvolle Erkentnisse gewonnen werden.

Anhand eines Distanzmaßes konnten Publikationen beispielhaft identifiziert werden, die sich thematisch sehr ähnlich sind. Häufig diskutierte Themen drücken sich durch die Anzahl der ihnen zugeordneten Dokumente aus und können so in eine Rangreihenfolge gebracht werden. Die analysierten Konferenzbeiträge befassen sich häufig mit Themen rund um Ökonomie,

Modelle, Datenhaltung, Bildung und Informationstechnik. Für eine Betrachtung der Trends wurde das quantitative Wachstum der Beiträge über die Zeiträume der letzten 5 und 10 Jahre betrachtet. Beiträge zu Themen, die sich mit E-Business, Internet-Auktionen und eher klassischen Internet-Themen wie Webseiten beschäftigen, nehmen seit 10 Jahren ab. Über die letzten 5 Jahre nehmen Themen wie Outsourcing, Open Source, Usability, E-Business und XML ab. Deutliche Trends und starkes Wachstum zeigen hingegen Themen, die nicht nur in der Wissenschaft populär sind. Publikationen der letzten 5 Jahre beschäftigen sich insbesondere mit sozialen Netzwerken, Green IT, social media bzw. web 2.0, Medizintechnik und Crowdsourcing. Bei einer Betrachtung der letzten 10 Jahre kommt zu diesen Themen die agile Softwareentwicklung hinzu. Das Thema der Content-Verbreitung hat sich über die letzten 19 Jahre stark verändert. Es deckt den Weg von den klassischen Printmedien über Web 2.0 bis zum aktuellen Begriff Social Media ab.

Das Verfahren kann auf unterschiedliche Datenbestände angewendet werden, so z.B. auch auf Bilder. Um gute Ergebnisse zu erhalten, sollten die Daten eine gewisse Größe aufweisen. Die Menge, die verarbeitet werden kann, wird durch die technische Ausstattung des Rechners begrenzt.

Für Auswertungen mit einem höheren Detailgrad ist die Anwendung von LDA Weiterentwicklungen denkbar. So können die Varianten LLDA und pLDA mit Labels umgehen und könnten im Rahmen von wissenschaftlichen Publikationen mit Keywords verwendet werden. Des Weiteren ist die Anwendung einer Variante von LDA denkbar, die auf bestimmte Themen angelernt wird.

Um ein genaueres Bild der Entwicklungen in der Wirtschaftsinformatik zu erstellen muss ein Datenbestand existieren, der diese Entwicklungen möglichst gut abbildet. Dies kann durch eine Steigerung der Quantität und Qualität erreicht werden. Bei Erhöhung der Breite des Datenbestandes besteht die Chance, die Interessen hinter den Daten genauer abzubilden. Dies kann z.B. durch die Hinzunahme weiterer Konferenzen geschehen. Für eine Steigerung der Qualität des Datenbestandes könnten z.B. Dokumente mit einbezogen werden, denen strengere Bewertungskriterien zu Grunde liegen, beispielsweise Beiträge aus Journalen im IS-Bereich.

Es ist bemerkenswert, welche Ergebnisse ein ungelernter Algorithmus aus Daten extrahieren kann. Jedoch sollte bedacht werden, dass ein Automatismus eine Hilfe darstellt und die Fähigkeiten eines Menschen nur unterstützen und nicht ersetzen kann.

Anhang

A Stoppwörter von TMT

a	beyond	former	ltd	over	therein	while
about	bill	formerly	made	own	thereupon	whither
above	both	forty	many	part	these	who
across	bottom	found	may	per	they	whoever
after	but	four	me	perhaps	thick	whole
afterwards	by	from	meanwhile	please	thin	whom
again	call	front	might	put	third	whose
against	can	full	mill	rather	this	why
all	cannot	further	mine	re	those	will
almost	cant	get	more	same	though	with
alone	co	give	moreover	see	three	within
along	computer	go	most	seem	through	without
already	con	had	mostly	seemed	throughout	would
also	could	has	move	seeming	thru	yet
although	couldnt	hasnt	much	seems	thus	you
always	cry	have	must	serious	to	your
am	de	he	my	several	together	yours
among	describe	hence	myself	she	too	yourself
amongst	detail	her	name	should	top	yourselves
amoungst	do	here	namely	show	toward	
amount	done	hereafter	neither	side	towards	
an	down	hereby	never	since	twelve	
and	due	herein	nevertheless	sincere	twenty	
another	during	hereupon	next	six	two	
any	each	hers	nine	sixty	un	
anyhow	eg	herself	no	so	under	
anyone	eight	him	nobody	some	until	
anything	either	himself	none	somehow	up	
anyway	eleven	his	noone	someone	upon	
anywhere	else	how	nor	something	us	
are	elsewhere	however	not	sometime	very	
around	empty	hundred	nothing	sometimes	via	
as	enough	i	now	somewhere	was	
at	etc	ie	nowhere	still	we	
back	even	if	of	such	well	
be	ever	in	off	system	were	
became	every	inc	often	take	what	
because	everyone	indeed	on	ten	whatever	
become	everything	interest	once	than	when	
becomes	everywhere	into	one	that	whence	
becoming	except	is	only	the	whenever	
been	few	it	onto	their	where	
before	fifteen	its	or	them	whereafter	
beforehand	fify	itself	other	themselves	whereas	
behind	fill	keep	others	then	whereby	
being	find	last	otherwise	thence	wherein	
below	fire	latter	our	there	whereupon	
beside	first	latterly	ours	thereafter	wherever	
besides	five	least	ourselves	thereby	whether	
between	for	less	out	therefore	which	

B Auflistung aller Topics im untersuchten Datenbestand

Topic 000	{ construct, measur, item, valid, model, indic, reliabl, thi, research, structur }
Topic 001	{ share, contribut, incent, reward, exchang, individu, reciproc, motiv, knowledg, help }
Topic 002	{ compani, market, industri, product, new, busi, technolog, competit, strategi, corpor }
Topic 003	{ data, databas, warehous, qualiti, sourc, attribut, manag, error, wareh, dimens }
Topic 004	{ eci, 2002, european, content, previou, june, poland, 2001, gda, ?sk, }
Topic 005	{ 0.00, 0.01, 0.05, tabl, 0.02, 1.00, 0.03, 0.04, 0.10, 0.06 }
Topic 006	{ power, polit, discours, societi, empower, critic, social, claim, ration, argument }
Topic 007	{ thei, thi, work, peopl, time, wai, becaus, make, differ, veri }
Topic 008	{ busi, oper, enterpris, unit, custom, transform, understand, concept, harvard, deliv }
Topic 009	{ comput, gender, women, femal, male, self-efficaci, career, men, technolog, famili }
Topic 010	{ integr, applic, infrastructur, technolog, support, data, enterpris, process, develop, eai }
Topic 011	{ thi, e.g, germani, onli, requir, approach, result, regard, process, possibl }
Topic 012	{ recommend, profil, prefer, person, filter, match, item, similar, rate, set }
Topic 013	{ thi, access, server, databas, data, applic, file, provid, architectur, request }
Topic 014	{ cultur, differ, nation, dimens, countri, hofsted, individu, global, power, valu }
Topic 015	{ ethic, moral, piraci, behavior, abus, pirat, sanction, comput, intent, issu }
Topic 016	{ network, structur, node, social, central, connect, link, analysi, individu, degre }
Topic 017	{ travel, hotel, tourism, car, vehicl, airlin, destin, locat, rout, tourist }
Topic 018	{ inform, univers, electron, amci, librari, america, proceed, aisel, confer, thi }
Topic 019	{ media, commun, messag, email, rich, e-mail, medium, social, cmc, channel }
Topic 020	{ intern, confer, :1),, :2),, 2002, :3),, :4),, ici, ?the, 2001 }
Topic 021	{ problem, solut, solv, intellig, constraint, support, situat, heurist, help, resolut }
Topic 022	{ advertis, attitud, credibl, persuas, messag, cue, brain, cognit, involv, banner }
Topic 023	{ user, featur, function, end-us, end, usag, comput, provid, applic, need }
Topic 024	{ electron, e-commerc, sme, commerc, e-busi, marketplac, b2b, market, small, busi }
Topic 025	{ relationship, transact, partner, exchang, allianc, relat, interorganiz, cooper, govern, cost }
Topic 026	{ secur, comput, attack, threat, polici, vulner, protect, breach, awar, password }
Topic 027	{ time, number, thi, result, figur, rate, increas, averag, total, tabl }
Topic 028	{ case, complianc, regul, regulatori, studi, replic, report, rule, compli, mandatori }
Topic 029	{ payment, card, authent, certif, kei, fraud, credit, secur, ident, transact }
Topic 030	{ success, measur, portal, mclean, delon, impact, satisfact, dimens, individu, instrument }
Topic 031	{ task, perform, fit, complex, technolog, work, interrupt, overload, characterist, time }
Topic 032	{ platform, legal, law, ownership, ecosystem, develop, right, properti, app, protect }
Topic 033	{ manag, implement, support, process, inform, plan, oper, depart, develop, need }
Topic 034	{ inform, technolog, provid, confer, need, access, sourc, avail, electron, seek }

Topic 035	{ factor, success, critic, import, identifi, influenc, csf, affect, literatur, 2001 }
Topic 036	{ 2006, 2005, 2004, research, 2007, 2003, inform, journal, manag, 2002 }
Topic 037	{ matur, level, process, measur, perform, manag, bpm, improv, practic, assess }
Topic 038	{ review, seller, rate, reput, onlin, buyer, product, market, marketplac, feedback }
Topic 039	{ internet, digit, access, divid, home, isp, technolog, connect, 2000, provid }
Topic 040	{ behavior, individu, motiv, social, theori, psycholog, person, influenc, goal, outcom }
Topic 041	{ page, european, 18th, fig, delet, format, eci, vol, [1], 1218th }
Topic 042	{ china, paci, chines, asia, pacif, hong, kong, taiwan, singapor, nation }
Topic 043	{ content, 2.0, wiki, web, media, new, wikipedia, technolog, newspap, video }
Topic 044	{ collabor, activ, work, tool, support, particip, awar, share, facilit, commun }
Topic 045	{ variabl, regress, estim, data, coeffici, correl, signific, statist, sampl, analysi }
Topic 046	{ ontolog, semant, languag, domain, diagram, conceptu, class, represent, construct, uml }
Topic 047	{ cognit, map, expert, memori, represent, mental, style, concept, understand, expertis }
Topic 048	{ project, manag, develop, success, team, failur, involv, member, complet, commit }
Topic 049	{ rule, algorithm, set, data, method, classif, predict, attribut, tree, fuzzi }
Topic 050	{ languag, text, word, extract, linguist, sentenc, analysi, phrase, featur, term }
Topic 051	{ usabl, visual, interfac, design, imag, displai, screen, color, present, navig }
Topic 052	{ sourc, open, develop, oss, softwar, code, project, linux, contribut, licens }
Topic 053	{ agent, auction, bid, negoti, price, buyer, bidder, seller, item, mechan }
Topic 054	{ sale, retail, channel, book, music, download, store, digit, sell, demand }
Topic 055	{ custom, bank, crm, relationship, market, servic, channel, sale, loyalti, transact }
Topic 056	{ cluster, document, similar, techniqu, categori, featur, event, new, vector, matrix }
Topic 057	{ cost, benefit, evalu, invest, improv, assess, measur, benchmark, save, method }
Topic 058	{ ident, emot, person, affect, psycholog, cope, feel, identif, social, cognit }
Topic 059	{ mind, mean, thi, human, perspect, world, understand, metaphor, concept, interpret }
Topic 060	{ commun, member, onlin, particip, social, discuss, forum, post, support, virtual }
Topic 061	{ 2010, proceed, america, august, confer, 2011, peru, sixteenth, lima, 42339 }
Topic 062	{ reus, taxonomi, repositori, asset, domain, reusabl, object, plm, metadata, refin }
Topic 063	{ decept, detect, spam, 0.0, phish, 100.0, tabl, email, 4.0, 3.0 }
Topic 064	{ comput, cloud, resourc, grid, peer, p2p, saa, alloc, provid, mechan }
Topic 065	{ mobil, devic, wireless, phone, technolog, applic, comput, locat, network, access }
Topic 066	{ global, local, region, intern, countri, subsidiari, locat, nation, foreign, european }
Topic 067	{ capabl, resourc, firm, competit, manag, advantag, strateg, perform, compet, technolog }
Topic 068	{ consum, product, onlin, purchas, shop, market, brand, internet, experi, bui }
Topic 069	{ america, 2003, confer, proceed, 2004, new, 2002, august, inform, york }
Topic 070	{ market, trade, stock, event, announc, price, investor, return, order, financi }
Topic 071	{ creativ, idea, gener, product, contest, crowd, crowdsourc, solver, innov, collect }
Topic 072	{ virtual, world, presenc, environ, life, avatar, second, social, real, interact }
Topic 073	{ emerg, disast, respons, incid, spatial, recoveri, crisi, plan, geograph, event }

Topic 074	{ actor, boundari, broker, translat, ant, span, object, network, role, actor-network }
Topic 075	{ tion, ing, ment, tio, volum, com, pro, tive, tional, iti }
Topic 076	{ green, environment, sustain, energi, consumpt, electr, practic, 2008, ecolog, initi }
Topic 077	{ group, member, support, process, discuss, individu, gss, facilit, meet, idea }
Topic 078	{ health, patient, healthcar, care, hospit, medic, clinic, nurs, record, provid }
Topic 079	{ architectur, enterpris, soa, layer, busi, implement, service-ori, compon, function, descript }
Topic 080	{ studi, research, data, interview, thi, analysi, case, qualit, collect, method }
Topic 081	{ design, artifact, pattern, scienc, principl, research, evalu, build, hevner, engin }
Topic 082	{ ict, develop, countri, nation, econom, technolog, economi, world, infrastructur, polici }
Topic 083	{ action, act, situat, intervent, state, plan, perform, world, phase, reflect }
Topic 084	{ intranet, sensemak, weick, extranet, corpor, stenmark, subcontractor, lai, scheeper, contractor }
Topic 085	{ 2009, san, francisco, california, america, august, proceed, fifteenth, 6th-9th, confer }
Topic 086	{ qualiti, satisfact, expect, dimens, perceiv, percept, measur, satisfi, continu, evalu }
Topic 087	{ coordin, distribut, work, time, tempor, distanc, mode, interdepend, mechan, remot }
Topic 088	{ model, figur, conceptu, refer, stage, propos, structur, construct, build, valid }
Topic 089	{ search, document, queri, retriev, engin, term, result, index, relev, keyword }
Topic 090	{ knowledg, manag, transfer, learn, process, share, tacit, creation, practic, individu }
Topic 091	{ organ, organiz, structur, manag, unit, level, environ, formal, central, extern }
Topic 092	{ perceiv, intent, technolog, accept, model, usag, behavior, attitud, davi, eas }
Topic 093	{ employe, work, commit, job, organiz, turnov, justic, workplac, climat, stress }
Topic 094	{ mexico, 2006, acapulco, twelfth, 04th-06th, august, america, proceed, que, del }
Topic 095	{ institut, cio, pressur, assimil, influenc, norm, ceo, role, execut, institution }
Topic 096	{ erp, implement, enterpris, sap, packag, process, busi, vendor, 2000, plan }
Topic 097	{ team, member, leadership, virtual, conflict, work, leader, perform, task, group }
Topic 098	{ game, player, hedon, plai, enjoy, onlin, utilitarian, experi, play, entertain }
Topic 099	{ ninth, 2008, pari, twenti, intern, confer, french, ici, franc, dan }
Topic 100	{ survei, respond, tabl, result, studi, thi, differ, respons, mean, signific }
Topic 101	{ thi, mai, inform, provid, need, thei, technolog, potenti, issu, requir }
Topic 102	{ 0.000, tabl, 0.001, 1000, signific, 0.002, 0.004, 0.003, 0.05, 0.005 }
Topic 103	{ privaci, concern, person, protect, consum, disclosur, onlin, polici, individu, collect }
Topic 104	{ process, activ, workflow, execut, improv, busi, autom, bpr, reengin, redesign }
Topic 105	{ stakehold, polici, broadband, vote, nonprofit, donat, donor, dwivedi, choudri, non-profit }
Topic 106	{ research, journal, articl, disciplin, field, paper, publish, scienc, author, academ }
Topic 107	{ australia, australian, 2003, south, pacif, asia, confer, juli, page, 2002 }
Topic 108	{ uncertainti, capac, acquisit, absorpt, acquir, merger, exploit, environment, environ, scan }

Topic 109	{ govern, public, e-govern, citizen, agenc, sector, servic, administr, state, electron }
Topic 110	{ canada, 2008, toronto, america, august, fourteenth, 14th-17th, proceed, confer, amci }
Topic 111	{ simul, scenario, input, effici, output, dynam, oper, perform, variabl, loop }
Topic 112	{ medic, physician, patient, emr, hospit, drug, clinic, error, record, care }
Topic 113	{ interact, convers, involv, mediat, human, pattern, human-comput, ritual, goffman, criterion }
Topic 114	{ research, theori, method, approach, theoret, practic, critic, scienc, ground, paradigm }
Topic 115	{ client, consult, advisor, advisori, advic, thai, thailand, relationship, engag, dutch }
Topic 116	{ servic, provid, custom, offer, deliveri, oper, provis, deliv, market, need }
Topic 117	{ manag, 1999, 1998, 1997, 1996, research, 1995, 2000, 1994, inform }
Topic 118	{ softwar, develop, product, test, engin, program, qualiti, code, requir, effort }
Topic 119	{ social, capit, network, blog, facebook, friend, onlin, peopl, post, twitter }
Topic 120	{ technolog, social, practic, structur, context, thi, work, inform, process, perspect }
Topic 121	{ adopt, technolog, diffus, innov, factor, influenc, roger, stage, studi, characterist }
Topic 122	{ rfid, tag, product, technolog, food, retail, sensor, store, track, reader }
Topic 123	{ risk, manag, loss, assess, mitig, probabl, failur, riski, percept, impact }
Topic 124	{ innov, new, technolog, patent, fashion, r&d, product, radic, entrepreneur, idea }
Topic 125	{ standard, xml, document, interoper, element, xbrl, format, exchang, data, schema }
Topic 126	{ outsourc, vendor, contract, offshor, relationship, manag, cost, client, servic, sourc }
Topic 127	{ chang, adapt, new, flexibl, transform, dynam, resist, transit, environ, time }
Topic 128	{ flow, question, answer, experi, ask, challeng, activ, concentr, hoffman, novak }
Topic 129	{ decis, make, decision-mak, support, choic, altern, dss, maker, process, select }
Topic 130	{ learn, student, educ, cours, technolog, e-learn, onlin, teach, instructor, environ }
Topic 131	{ thi, inform, differ, framework, approach, base, level, object, relat, type }
Topic 132	{ control, account, monitor, audit, report, intern, financi, mechan, auditor, object }
Topic 133	{ particip, experi, subject, test, experiment, result, evalu, studi, condit, feedback }
Topic 134	{ organis, thi, behaviour, centr, analys, develop, approach, programm, eci, structur }
Topic 135	{ agil, develop, improvis, practic, team, custom, method, program, pair, scrum }
Topic 136	{ firm, invest, product, industri, perform, technolog, capit, measur, econom, impact }
Topic 137	{ develop, requir, method, methodolog, phase, process, analysi, approach, techniqu, engin }
Topic 138	{ price, market, cost, thi, econom, profit, consum, product, optim, good }
Topic 139	{ chain, suppli, supplier, product, manufactur, manag, order, inventori, demand, logist }
Topic 140	{ effect, studi, research, thi, inform, posit, perform, impact, level, result }
Topic 141	{ invest, option, financi, portfolio, real, return, valu, valuat, asset, cash }
Topic 142	{ web, site, websit, page, content, web-bas, internet, design, navig, link }
Topic 143	{ trust, perceiv, relationship, vendor, belief, onlin, intent, transact, trustworthi, parti }
Topic 144	{ compon, complex, metric, mainten, modul, modular, measur, coupl, defect,

	inspect }
Topic 145	{ student, program, cours, univers, educ, faculti, curriculum, teach, school, academ }
Topic 146	{ strategi, strateg, align, busi, plan, manag, perform, orient, dimens, technolog }
Topic 147	{ skill, profession, train, job, worker, compet, work, employ, human, recruit }
Topic 148	{ 2009, 2010, 2008, 2011, thirti, intern, second, 2007, shanghai, loui }
Topic 149	{ valu, creat, object, creation, maxim, potenti, realiz, mean, assess, configur }

C Die 50 bzw. 20 % häufigsten Wörter des Korpus

Wort	Häufigkeit	Wort	Häufigkeit
inform	499719	social	104771
thi	380116	support	102726
research	263604	work	99056
manag	255264	design	99016
model	228972	valu	95216
process	222718	perform	94441
studi	195761	proceed	94302
technolog	191473	group	93801
develop	191425	analysi	93726
busi	173618	time	93559
knowledg	162615	level	92980
servic	153893	mai	92977
data	152759	case	91367
user	139698	base	88566
organ	135408	network	88124
commun	134700	custom	87975
new	128680	journal	87851
differ	128402	market	87026
provid	126756	chang	86089
thei	126046	intern	85742
project	124342	need	85351
confer	121302	softwar	85165
effect	112704	factor	83646
product	111683	electron	83455
result	109233	univers	83388

D Topic-Entwicklung von 1993 - 2011

Topic	Jahr	Wörter									
Topic 026	1993	comput	crime	polici	backup	practic	ernst	assur	investig	framework	human
Topic 026	1994	secur	threat	comput	polici	protect	patch	breach	viru	attack	vulner
Topic 026	1995	secur	ism	threat	comput	breach	woon	vulner	worm	polici	confidenti
Topic 026	1996	secur	intrus	password	vulner	polici	protect	threat	symantec	comput	hacker
Topic 026	1997	secur	threat	intrus	comput	detect	anderson	attack	polici	incid	malici
Topic 026	1998	secur	comput	cyber	threat	protect	polici	confidenti	attack	infosec	spoof
Topic 026	1999	secur	threat	intrus	detect	comput	protect	firewal	spoof	attack	polici
Topic 026	2000	secur	comput	polici	attack	threat	protect	firewal	vulner	skm	bs7799
Topic 026	2001	secur	threat	attack	comput	protect	password	polici	vulner	breach	hacker
Topic 026	2002	secur	intrus	detect	comput	breach	firewal	polici	hacker	attack	vulner
Topic 026	2003	secur	attack	comput	polici	protect	threat	detect	intrus	vulner	awar
Topic 026	2004	secur	attack	threat	vulner	comput	polici	intrus	protect	detect	password
Topic 026	2005	secur	vulner	attack	comput	patch	protect	threat	polici	password	worm
Topic 026	2006	secur	vulner	polici	comput	attack	protect	threat	breach	awar	patch
Topic 026	2007	secur	comput	threat	polici	attack	protect	vulner	awar	viru	complianc
Topic 026	2008	secur	threat	vulner	polici	comput	attack	protect	awar	breach	cyber
Topic 026	2009	secur	attack	comput	threat	vulner	breach	polici	protect	password	awar
Topic 026	2010	secur	polici	attack	password	threat	comput	awar	protect	vulner	breach
Topic 026	2011	secur	threat	polici	comput	attack	breach	protect	vulner	complianc	cyber

Topic	Jahr	Wörter									
Topic 043	1993	multimed	new	technolog	videotex	http	author	multi-medi	evolut	turban	applic
Topic 043	1994	content	2	edit	newslett	newspap	print	new	reader	media	publish
Topic 043	1995	content	2	newspap	new	technolo	articl	editor	page	collect	corpor
Topic 043	1996	content	publish	web	media	reader	new	read	readership	forum	comment
Topic 043	1997	content	newspap	2	journalist	media	web	new	technolog	reader	editori
Topic 043	1998	content	publish	web	2	video	new	reader	page	media	provid
Topic 043	1999	content	newspap	2	reader	new	print	media	web	technolog	editori
Topic 043	2000	content	newspap	new	post	eiu	publish	print	media	web	edit
Topic 043	2001	content	newspap	new	media	web	2	onlin	technolog	audienc	publish
Topic 043	2002	content	syndic	2	publish	new	media	web	technolog	quadrant	page
Topic 043	2003	content	ecm	media	web	new	technolog	2	video	newspap	onlin
Topic 043	2004	content	wiki	media	new	video	2	web	technolog	edit	onlin
Topic 043	2005	content	media	weblog	wiki	web	technolog	ecm	video	edit	new
Topic 043	2006	content	newspap	weblog	podcast	new	wiki	media	technolog	web	page
Topic 043	2007	content	wiki	2	web	podcast	media	new	wikipedia	technolog	edit
Topic 043	2008	content	2	wiki	wikipedia	web	ecm	new	video	technolog	media
Topic 043	2009	content	2	web	wiki	new	media	wikipedia	blog	technolog	onlin
Topic 043	2010	content	2	wiki	web	wikipedia	new	technolog	newspap	media	onlin
Topic 043	2011	content	2	wiki	web	media	wikipedia	new	technolog	video	edit

Literaturverzeichnis

Agarwal, Nitin; Liu; Huan (2009): Modeling and Data Mining in Blogosphere. Morgan & Claypool.

Association for Information Systems (o. J.): AIS Electronic Library - All Content. http://aisel.aisnet.org/communities.html, Abruf am 2012-03-28.

Associaton for Information Systems Research (o. J.): AIS Electronic Library (AISeL). http://aisel.aisnet.org/communities.html, Abruf am 2012-03-31.

Baeza-Yates, Ricardo; Ribeiro-Neto, Berthier; (1999): Modern Information Retrieval. 1. Auflage, Addison Wesley.

Blei, David M. (2006): Latent Dirichlet allocation in C. http://www.cs.princeton.edu/~blei/lda-c/, Abruf am 2012-03-31.

Blei, David M. (2009a): Topic Models. http://videolectures.net/site/normal_dl/tag=50740/mlss09uk_blei_tm.pdf, Abruf am 2012-03-31.

Blei, David M. (2009b): Topic Models. http://videolectures.net/mlss09uk_blei_tm/, Abruf am 2012-03-31.

Blei, David M. (2011a): Introduction to probabilistic topic models. In: Communications of the ACM, 55 (4), S. 77-84.

Blei, David M. (2011b): Probabilistic Topic Models. http://www.cs.princeton.edu/~blei/kdd-tutorial.pdf, Abruf am 2012-03-31.

Blei, David M. (2012): Topic modeling. http://www.cs.princeton.edu/~blei/topicmodeling.html, Abruf am 2012-03-31.

Blei, David M.; Lafferty, John D. (2009): Topic Models. In: Srivastava, Ashok; Sahami, Mehran (Hrsg.): Text Mining Classification, Clustering, and Applications. Chapman & Hall/CRC, Boca Raton, S. 71-89.

Blei, David M.; Ng, Andrew Y.; Jordan, Micheal I. (2003): Latent Dirichlet Allocation. In: Journal of Machine Learning Research, 2003 (3), S. 993–1022.

Buntine, Wray (2012): Discrete Component Analysis. http://www.nicta.com.au/people/buntinew/discrete_component_analysis, Abruf am 2012-03-31.

Carenini, Giuseppe; Murray, Gabriel; Ng, Raymond (2011): Methods for Mining and Summarizing Text Conversations. Morgan and Claypool.

Chang, Jonathan (2012): lda: Collapsed Gibbs sampling methods for topic models. http://cran.r-project.org/web/packages/lda/, Abruf am 2012-03-31.

Chang, Jonathan; Gerrish, Sean; Wang, Chong; Blei, David M. (2009): Reading Tea Leaves: How Humans Interpret Topic Models. In: Neural Information Processing Systems.

Deerwester, Scott; Dumais, Susan T.; Furnas, George W.; Landauer, Thomas K.; Harshman, Richard (1990): Indexing by Latent Semantic Analysis. In: Journal of the American Society for Information Science, 41 (6), S. 391-407.

École Polytechnique Fédérale de Lausanne (2012): Scala License | The Scala Programming Language. http://www.scala-lang.org/node/146, Abruf am 2012-03-31.

Erickson, Lee B. (2011): Social media, social capital, and seniors: The impact of Facebook on bonding and bridging social capital of individuals over 65. AMCIS 2011 Proceedings - All Submissions. Paper 85.

Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996): From Data Mining to Knowledge Discovery in Databases. In: AI Magazine, 17 (3), S. 37-54.

Free Software Foundation (1991): GNU GENERAL PUBLIC LICENSE, Version 2, June 1991. http://www.gnu.org/licenses/gpl-2.0.html, Abruf am 2012-03-31.

Google (2011): Unternehmen. http://www.google.de/intl/de/about/corporate/company/index.html, Abruf am 2012-03-31.

Google (2012): Deerwester: Indexing by latent semantic analysis. http://scholar.google.de/scholar?cites=1267915114494150539, Abruf am 2012-03-31.

Goswami, Suparna; Köbler, Felix; Leimeister, Jan Marco; Krcmar, Helmut (2010): Using Online Social Networking to Enhance Social Connectedness and Social Support for the Elderly. ICIS 2010 Proceedings. Paper 109.

Griffiths, Thomas L.; Steyvers, Mark (2004): Finding Scientific Topics. In: Proceedings of the National Academy of Sciences, 101 (1), S. 5228-5235.

Grossman, David A.; Frieder, Ophir (2004): Information Retrieval: Algorithms and Heuristics. 2. Auflage, Springer Netherlands, Dordrecht.

Han, Jiawei; Cai, Yandong; Cercone, Nick (2004): Knowledge Discovery in Databases: An Attribute-Oriented Approach. In: Proceedings of the 18th VLDB Conference. Vancouver, S. 547-559.

Hofmann, Thomas (1999): Probabilistic Semantic Indexing. Proceedings oft he Twenty-Second Anual International SIGIR Conference on Research and Development in Information Retrieval.

Intel (2005): Excerpts from A Conversation with Gordon Moore: Moore's Law. ftp://download.intel.com/museum/Moores_Law/Video-Transcripts/Excepts_A_Conversation_with_Gordon_Moore.pdf, Abruf am 2012-03-31.

Jochum, Clemens (2012): Forschungsgemeinschaft Frankfurt Cloud. http://www.frankfurt-cloud.com/, Abruf am 2012-03-31.

K**azienko, Przemyslaw; Musial, Katarzyna; Kajdanowicz, Tomasz (2008):** Profile of the Social Network in Photo Sharing Systems. AMCIS 2008 Proceedings. Paper 173.

Kremp, Matthias (2008): Green IT: Die grüne Cebit ist eine Mogelpackung. http://www.spiegel.de/netzwelt/tech/0,1518,540163,00.html, Abgerufen am 2012-03-31.

Lacoste-Julien, Simon; Sha, Fei; Jordan, Micheal I. (2008): DiscLDA: Discriminative Learning for Dimensionality Reduction and Classification. In: Neural Information Processing Systems.

Lieberman, Marvin B.; Montgomery, David B. (1988): First-Mover Advantages. In: Strategic Management Journal, 9 (1), S. 41-58.

Maier, Nils; Parodi, Federico; Verna, Stefano (2012): DownThemAll! :: Add-ons für Firefox. https://addons.mozilla.org/de/firefox/addon/downthemall/, Abruf am 2012-03-31.

McCallum, Andrew (2011): MAchine Learning for LanguagE Toolkit. http://mallet.cs.umass.edu/, Abruf am 2012-03-31.

Mimno, David (2012): Topic Modeling Bibliography. http://www.cs.princeton.edu/~mimno/topics.html, Abruf am 2012-03-31.

Minka, Thomas; Lafferty, John (2002): Expectation-Propagation for the Generative Aspect Model. In: Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence, S. 352-359.

Mozilla Corporation (2012): mozilla Firefox. http://www.mozilla.com/firefox/, Abruf am 2012-03-31.

Nallapati, Ramesh (2012): Software - Ramesh Nallapati's homepage. https://sites.google.com/site/rameshnallapati/software, Abruf am 2012-03-31.

Narayanamurthy, Shravan (2011): Yahoo_LDA wiki!. https://github.com/shravanmn/Yahoo_LDA/wiki, Abruf am 2012-03-31.

Niesler, T. R.; Woodland, P. C. (1996): A variable-length category-based n-gram language model. IEEE International Conference on ACOUSTICS, SPEECH AND SIGNAL PROCESSING, Atlanta, S. 164-167.

Phan, Xuan-Hieu; Nguyen, Cam-Tu (2008): GibbsLDA++: A C/C++ Implementation of Latent Dirichlet Allocation. http://gibbslda.sourceforge.net/, Abruf am 2012-03-31.

Pussep, Anton; Schief, Markus; Schmidt, Benedikt; Friedrichs, Florian; Buxmann, Peter (2012): Topics in Software Industry Transformation Research: A Topic Analysis of Major IS Conferences. International Conference on Software Business, Cambridge, Massachusetts.

Ramage, Daniel; Dumais, Susan; Liebling, Dan (2010): Characterizing Microblogs with Topic Models. In: Hearst, Marti; Cohen, William; Gosling Samuel (Hrsg.): Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media, Washington, S. 130-137.

Ramage, Daniel; Manning, Christopher D.; Dumais, Susan (2011): Partially Labeled Topic Models for Interpretable Text Mining. In: The 17th ACM SIGKDO Conference on Knowledge Discovery and Data Mining. San Diego.

Řehůřek, Radim (2012): Gensim – Topic Modelling for Humans. http://radimrehurek.com/gensim/, Abruf am 2012-03-31.

Runkler , Thomas A. (2010): Data Mining. Vieweg+Teubner, GWV Fachverlage GmbH, Wiesbaden.

Schürig, Henning (2010): Social Media statt Web 2.0. http://www.henningschuerig.de/blog/2010/social-media-statt-web-20/, Abgerufen am 2012-03-31.

Smith, Glen; Romanda, J.C.; Conway, Scott; Sullivan, Sean (2011): opencsv. http://opencsv.sourceforge.net/, Abruf am 2012-03-31.

Sommer, Stefan; Schieber, Andreas; Hilbert, Andreas; Heinrich, Kai (2011): Analyzing customer sentiments in microblogs – A topic- model-based approach for Twitter datasets. In: AMCIS 2011 Proceedings - All Submissions, Detroit, Paper 227.

Steyvers, Mark; Griffiths, Tom (2006): Probabilistic Topic Models. In: Landauer, T.; McNamara, D.; Dennis, S.; Kintsch, W. (Hrsg.): Latent Semantic Analysis: A Road to Meaning. Laurance Erlbaum.

Steyvers, Mark; Griffiths, Tom (2011): Matlab Topic Modeling Toolbox 1.4 . http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm, Abruf am 2012-03-31.

Taranto, Claudio; Di Mauro, Nicola; Esposito, Floriana (2011): rsLDA: a Bayesian Hierarchical Model for Relational Learning. International Conference on Data and Knowledge Engineering, Italy.

Teh, Yee Whye; Newman, David; Welling, Max (2006): A Collapsed Variational Bayesian Inference Algorithm for Latent Dirichlet Allocation. In: Neural Information Processing Systems.

The Apache Software Foundation (2004): Apache License, Version 2.0, January 2004. http://www.apache.org/licenses/LICENSE-2.0.html, Abruf am 2012-03-31.

The Apache Software Foundation (2011a): Apache Commons Logging. http://commons.apache.org/logging/, Abruf am 2012-03-31.

The Apache Software Foundation (2011b): Apache Commons Codec. http://commons.apache.org/codec/, Abruf am 2012-03-31.

The Apache Software Foundation (2011c): Apache Commons Lang. http://commons.apache.org/lang/, Abruf am 2012-03-31.

The Apache Software Foundation (2011d): Apache Logging Services Project – Apache log4j. http://logging.apache.org/log4j/, Abruf am 2012-03-31.

The Apache Software Foundation (2011e): HttpClient. http://hc.apache.org/httpclient-3.x/, Abruf am 2012-03-31.

The Apache Software Foundation (2012): Apache PDFBox - Java PDF Library. http://pdfbox.apache.org/, Abruf am 2012-03-31.

The Stanford Natural Language Processing Group (2011): Standard Topic Modeling Toolbox. http://nlp.stanford.edu/software/tmt/tmt-0.4/, Abruf am 2012-03-31.

Thierse, Wolfgang (2003): Vortrag "Traditionswahrung und Modernisierung - Sozialdemokratie in der Entscheidung“. http://library.fes.de/fulltext/historiker/01705-03.htm, Abruf am 2012-03-31.

vom Brocke, Jan; Simons, Alexander; Niehaves, Björn; Riemer, Kai; Plattfaut, Ralf; Cleven, Anne (2009): Reconstructing the giant: on the importance of rigour in documenting the literature search process. In: Tuunainen, Virpi Kristiina; Rossi, Matti; Nandhakumar, Joe (Hrsg.): 17th European Conference on Information Systems, Helsinki, S. 1-13.

Watson, Richard; Boudreau, Marie-Claude; Chen, Adela; Huber, Mark (2008): Green IS : Building Sustainable Business Practices. Global Text Project, Atlanta, Georgia.

Westphal, Christopher; Blaxton, Teresa (1998): Data Mining Solutions. Wiley.

Wikipedia (o. J.): Crowdsourcing. http://de.wikipedia.org/wiki/, Abgerufen am 2012-03-31.

Wikipedia (o. J.): Dirichlet-Verteilung. http://de.wikipedia.org/wiki/Dirichlet-Verteilung, Abruf am 2012-03-31.

Wikipedia (o. J.): Karotte. http://de.wikipedia.org/wiki/Karotte#Kulturgeschichte, Abruf am 2012-03-31.

Wikipedia (o. J.): Mooresches Gesetz. http://de.wikipedia.org/wiki/Mooresches_Gesetz, Abruf am 2012-03-28.

Wikipedia (o. J.): Perplexität (Statistik). http://de.wikipedia.org/wiki/Perplexit%C3%A4t_%28Statistik%29, Abruf am 2012-03-28.

Wikipedia (o. J.): Scala (Programmiersprache). http://de.wikipedia.org/wiki/Scala_%28Programmiersprache%29, Abruf am 2012-03-31.

Wikipedia (o. J.): Webcrawler. http://de.wikipedia.org/wiki/Webcrawler, Abruf am 2012-03-31.

Witten, Ian H.; Frank, Eibe; Hall, Mark A. (2011): Data Mining: Practical Machine Learning Tools and Techniques. 3. Auflage, Morgan Kaufmann, Burlington.

World Wide Web Consortium (1999): Hypertext Transfer Protocol -- HTTP/1.1. http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html, Abruf am 2012-03-31.

Yan, Feng; Xu, Ningyi; Qi, Yuan (2009): Parallel Inference for Latent Dirichlet Allocation on Graphics Processing Units. In: Neural Information Processing Systems.