Rheinische Friedrich-Wilhelms-Universität Bonn

Die Untersuchung von thematischen Zentralitätsverläufen in Textsequenzen

Centime

Diplomarbeit zur Erlangung des akademischen Grades eines Diplom-Informatiker
im Studiengang Informatik

Eingereicht von: Florian Schulz
1597819

Erstgutachter: Prof. Dr. Stefan Wrobel
Zweitgutachter: Prof. Dr. Rainer Manthey

Bonn den, 29. September 2009

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel verfasst habe.

....

Datum Unterschrift

Inhaltsverzeichnis

1 Einleitung

Zeitliche Veränderung in der Thematik von Texten zu erfassen ist eine wesentliche Aufgabe in der modernen, durch eine Flut von Informationen charakterisierten Gesellschaft. Dies kann von der Überwachung von Nachrichtentexten [7] über Trenderkennung in Blogs [9] bis zur Identifikation von Themen in Chats reichen. Die Anwendungen sind vielfältig. Im Kontext einer Lernplattform kann man sich einen Chat vorstellen, in dem Schüler sich über verschiedene zu bearbeitende Themen unterhalten. Ein Lehrer, der beobachtet über welche Themen gesprochen wird, könnte erkennen, ob erstens überhaupt etwas getan wird und zweitens, ob noch Lernbedarf besteht.

Die bisherigen Verfahren weisen oft Einschränkungen bei der Erkennung der Themen bzw. der Verifizierbarkeit der Themen auf. Oftmals müssen die Texte manuell annotiert werden, um die Themen erkennen zu können, oder die Themen sind nur aufwändig zu ermitteln [15]. Ferner ist die betrachtete Zeitspanne häufig nicht variabel genug bzw. wird bei der Erstellung des Modells zum Erkennen der Themen festgelegt und kann somit nicht unabhängig von den Themen betrachtet werden [16].

In dieser Diplomarbeit soll eine Methode entwickelt werden, um die zeitliche Änderung der Themen in Texten zu erfassen. Die einzelnen Textfragmente dieser sequentiellen Texte müssen eine zeitliche Ordnung aufweisen und somit sequentiell angeordnet werden können. Diese zu entwickelnde Methode soll eine möglichst große Variabilität bzgl. der betrachteten Zeitspanne bieten. Ergänzend soll der Aufwand in der Erkennung der thematischen Verläufe gering gehalten werden, so dass eine Online-Anwendung möglich ist. Um dies zu erreichen, wird folgende Methode vorgeschlagen.

Ausgehend von der Annahme, dass sequentielle Texte vorliegen, werden zuerst die Themen anhand Hintergrundwissen oder der vorliegenden Textkollektion gelernt. Anhand dieses Modells werden für die sequentiellen Texte die Themen bestimmt und eine relationale Struktur in Form eines Graphen aufgebaut. Auf diesen Graphen werden Zentralitätsmaße angewandt, um die Wichtigkeit der Themen numerisch zu erfassen. Die zeitlichen Verläufe der Zentralitätsmaße werden dann geeignet visualisiert. Da die Ergebnisse und die entwickelte Methode im EU-Projekt *Science Created by You* (SCY) genutzt werden sollen, ergeben sich spezielle Anforderungen, die in Abschnitt 1.1 noch genauer erläutert werden.

Das SCY-Projekt[1] ist ein EU-Projekt mit dem Ziel eine Plattform zu entwickeln, die es Schülern ermöglicht, selbstständig zu lernen. Im Gegensatz zu vielen anderen E-Learning Systemen bietet das SCY-Projekt eine Plattform, die kollaboratives Lernen verschiedener Inhalte

[1] http://www.scy-net.eu

unterstützt. Die Lerninhalte werden als Missionen modelliert, in denen den Schülern unterschiedliche Werkzeuge zur Verfügung stehen, um die gestellte Aufgabe zu lösen.

Eine der ersten Missionen beschreibt die Aufgabe, ein CO_2-neutrales Haus zu entwickeln. Dazu müssen die Schüler sich zuerst generell über CO_2 und dessen Auswirkung auf den Treibhauseffekt informieren. Anschließend informieren sie sich über Maßnahmen, die den CO_2-Ausstoß eines Hauses senken können. Dies können verschiedene Baustoffe, Dämmmaterialien oder andere Techniken sein. Hierzu werden Hintergrundtexte gelesen und Simulationen durchgeführt, die den Zusammenhang zwischen den verschiedenen CO_2-Quellen und den Gegenmaßnahmen darstellen. Ausgehend von diesem Hintergrundwissen werden dann die Häuser entworfen.

SCY unterstützt den Schüler dabei durch verschiedene Werkzeuge, die kollaborativ genutzt werden können. Insbesondere können sich die Schüler per Chat über etwaige Unklarheiten oder den Entwurf des Hauses austauschen. Die Chats werden aufgezeichnet und sollen als Quelle für die zu untersuchenden Textsequenzen dienen. Wie genau die Themenveränderungen in den Chat- und sonstigen Texten erfasst werden sollen, welche Schwierigkeiten dabei auftreten können und wie diese gelöst werden, wird in dieser Arbeit erläutert.

1.1 Ziele der Arbeit

Das Ziel dieser Diplomarbeit ist es, die zeitliche Veränderung von Themen in Textsequenzen zu erfassen. Im Speziellen soll die Prominenz der Themen erfasst und die Veränderungen über die Zeit dargestellt werden. Ein Thema ist ein prominentes Thema, wenn es oft in einer Textsequenz auftritt. Es könnten nun, wie in es in Linstead u. a. [15] angewendet wird, für jeden Zeitabschnitt die Themen gelernt werden und gezählt werden, wie oft ein Thema vorkommt. Dieser Ansatz ist jedoch für die vorliegende Arbeit ungeeignet, da erstens die Ergebnisse als Online-Algorithmus im SCY-Projekt benutzt werden sollen und das Lernen der Themen doch eine zeitaufwändige Aufgabe ist. Zweitens werden hier die Wahrscheinlichkeiten des Auftretens der Themen nicht berücksichtigt, sondern es wird jedes Auftreten gezählt.

Der Ansatz in dieser Arbeit ist ähnlich, in wichtigen Punkten unterscheidet er sich jedoch. Zum einen wird zwischen der Lernphase und der Anwendungsphase unterschieden. In der Lernphase wird anhand von Hintergrundtexten ein Themenmodell trainiert. Diese Hintergrundtexte sind nicht notwendigerweise zeitlich sortiert und enthalten Informationen über die Themen, die trainiert werden sollen. Dies sind zum Beispiel Artikel der Deutschen Presse Agentur (dpa) zu bestimmten Themen wie Politik, Kunst oder, im Kontext des SCY-Projektes, Texte mit Hintergrundwissen für die aktuelle zu bearbeitende Mission. So kann ein Themenmodell erstellt werden, das für den speziellen Kontext geeignet ist. Diese Eignung soll in der Diplomarbeit auch validiert werden. Die genaue Vorgehensweise wird in Kapitel 4 behandelt. Die Qualität der Themenmodelle hängt direkt von der Vorverarbeitung der Texte ab. Es muss also zusätzlich ermittelt werden, welche Vorverarbeitunsschritte die Qualität der Themenmodelle beeinflussen.

Abbildung 1.1.1: Visualisierung der Themenveränderung

In der Anwendungsphase werden die vorkommenden Themen identifiziert, deren Wichtigkeit bewertet und schlussendlich die zeitliche Veränderung dargestellt. Dazu werden, wie in Shaffer u. a. [22], die Textsequenzen in diskrete zeitliche Abschnitte unterteilt. Die Einteilung richtet sich dabei nach vorhandenen Dokumenten. In einen zeitlichen Abschnitt werden eine bestimmte Anzahl von zeitlich aufeinander folgenden Dokumenten zusammengefasst. Je nach Größe der Zeitabschnitte können so verschieden Zeiträume betrachtet werden. Im Falle von Chatnachrichten kann ein Zeitabschnitt nur ein paar Minuten umfassen, während im Falle von Nachrichten die Zeitabschnitte eine Woche, einen Monat oder mehr abdecken können. Da die Wahl der Größe der Zeitabschnitte unabhängig vom gelernten Themenmodell ist, können so auch verschiedene Zeiträume in annehmbarer Zeit betrachtet und verglichen werden.

Für jedes Dokument in einem Zeitabschnitt werden die vorkommenden Themen und die Wahrscheinlichkeit, mit der diese Themen vorkommen, ermittelt. Aus den ermittelten Themen wird anhand der Kookkurrenz und den Wahrscheinlichkeiten eine Graphenstruktur aufgebaut. Es werden verschiedene Algorithmen entwickelt, die dann auf ihre Eignung geprüft werden. Auf die Graphen werden dann verschiedene Zentralitätsindizes angewendet. Zentralitätsindizes bewerten die Wichtigkeit eines Knoten in einem Graphen. So wird die Prominenz der Themen ermittelt. Auch hier gilt es zu evaluieren, welcher Zentralitätsindex am besten geeignet ist, die Prominenz der Themen zu bewerten.

Wenn mehrere Zeitabschnitte nacheinander betrachtet werden, kann die thematische Veränderung einfach visualisiert werden. Hierzu werden die Veränderungen als Kurve dargestellt. In Abbildung 1.1.1 ist ein Beispiel für eine solche Visualisierung dargestellt. Da die Visualisierung unübersichtlich werden kann, wenn viele Themen auftreten, werden auch hier verschiedene Ansätze untersucht.

Es gilt also folgende Probleme zu lösen:

- Themenmodelle anhand von Hintergrundtexten zu trainieren und ihre Eignung für die Aufgabenstellung zu evaluieren.
- Textsequenzen in diskrete Zeitabschnitte zu unterteilen und festzustellen, welche Größe der Zeitabschnitte für die verschiedenen Textdatensätze geeignet ist.
- Aus den ermittelten Themen einen Graphen zu erstellen und zu prüfen, welcher Algorithmus zusammen mit den Zentralitätsindizes die zugrundeliegenden Themen in den Texten am besten repräsentiert.
- eine geeignet Art der Visualisierung entwickeln.

1.2 Zusammenfassung

Im diesem Kapitel wurde die Motivation der geplanten Anwendung dargelegt und zwei Anwendungsbeispiele betrachtet. Anschließend wurde dargestellt, welche Ziele erreicht werden sollen und angeschnitten wie diese realisiert werden sollen. Dies wird in der weiteren Arbeit genauer beschrieben.

In Kapitel 2 wird auf die Grundlagen wie Themenmodelle und Zentralitätsmaße eingegangen, die in der Literatur schon erforscht wurden und hier nicht weiter entwickelt bzw. nur unverändert benutzt werden. Kapitel 3 stellt Arbeiten vor, die mit ähnlichen Ansätzen arbeiten, wie sie in der Diplomarbeit verfolgt werden. Das 4. Kapitel stellt die verwendeten Daten und die Vorarbeit, die für diese Daten nötig ist, dar. Insbesondere wird darauf eingegangen, welche Daten zum Training des Themenmodells benutzt und welche Daten als Textsequenz benutzt werden.

Der eigentliche entwickelte Algorithmus, bzw. die Applikation zur Erkennung und Bewertung der Themen wird in Kapitel 5 erläutert. Dort werden die einzelnen Schritte dargestellt, wie man von einem sequentiellen Text zu einer Darstellung von thematischen Verläufen kommt. Zusätzlich werden Methoden zur Evaluation des Algorithmus entwickelt. Die Ergebnisse der durchgeführten Experimente und die Bewertung derselben wird in Kapitel 6 veranschaulicht. Im letzten Kapitel wird noch ein kurzer Ausblick auf verschiedene Anwendungen der entwickelten Methode gegeben und die Methode und ihre Ergebnisse kritisch hinterfragt.

2 Grundlagen

Im folgenden Kapitel werden die Techniken erläutert, die als Grundlage für diese Diplomarbeit dienen. Im Abschnitt über Themenmodelle wird hergeleitet, wie aus einem Korpus von Dokumenten ein Themenmodell erzeugt. Dabei wird insbesondere auf die statistische Inferenz eingegangen. Im Abschnitt über Zentralitäten werden die benutzten Zentralitätsindizes erläutert.

Es wird folgende Notation durchgängig zur Bezeichnung von Wörtern, Dokumenten und Korpora benutzt[1].

- Ein Korpus $\mathcal{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_M)$ besteht aus M Dokumenten.
- Jedes Dokument $\mathbf{w}_m = (w_{m,1}, \ldots, w_{m,N})$ besteht aus einer Sequenz von N Wörtern $w_{m,n}$. Mit $n = \{1, \ldots, N\}$ und $m = \{1, \ldots, M\}$.
- Ein Wort $w_{m,n}$ bezeichnet das n-te Wort in Dokument $\mathbf{w}_m$. Ein Wort ist ein Vorkommen eines Terms t aus dem Vokabular V. Mehrere Wörter können den gleichen Term instantiieren. So enthält der Satz *Ein Affe bleibt ein Affe, auch in Seide gekleidet* zweimal das Wort *Affe*, dieses instantiiert aber denselben Term *Affe*.
- Ein Term t ist ein eindeutiges Merkmal im zugrundeliegenden Korpus. Im vorliegenden Fall sind dies Wörter im herkömmlichen Sinne. So ist z. B. *Affe* ein Term während *Affen* ein anderer Term ist.
- Mit $z_{m,n}$ wird das zugehörige Thema des Wortes $w_{m,n}$ in Dokument $\mathbf{w}_m$ bezeichnet. Ein Thema ist hier keine semantische Einheit im klassischen Sinne. Wie ein Thema definiert ist wird im folgenden Abschnitt 2.1 erklärt.
- $\mathbf{z}_m$ bezeichnet die Themen des Dokumentes $\mathbf{w}_m$. Es ist $\mathbf{z}_m = (z_{m,1}, \ldots, z_{m,N})$

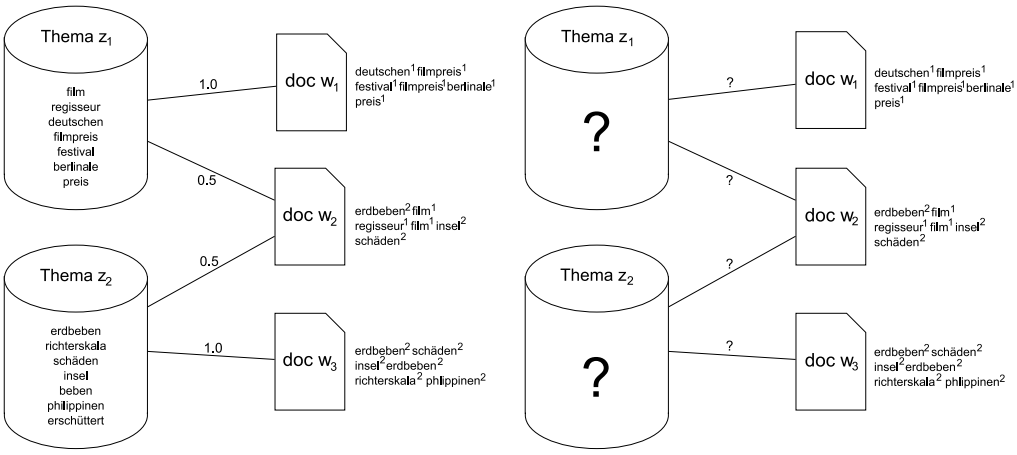
2.1 Themenmodelle

Ein Themenmodell [12, 4, 24] ist ein statistisches Modell um Themen aus Textkorpora zu extrahieren. Dazu werden Terme gruppiert, die oft zusammen auftreten. Diese Gruppierung wird als Thema bezeichnet. Ein Thema ist demnach durch die Verteilung der zugehörigen Terme charakterisiert. Ein Term kann dabei zu mehreren Themen gehören.

[1] Die Notation und die folgende Herleitung wurde zu Teilen aus [11] übernommen

Die Themen werden mit den zwei Termen, die mit der höchsten Wahrscheinlichkeit zu einem Thema gehören, benannt. Die beiden Themen in Abbildung 2.1(a) würden mit *film regisseur* und *erdbeben richterskale* bezeichnet. So kann in mehr oder weniger direkt auf den Inhalt des Themas geschlossen werden und die Themenbezeichnung ist besser lesbar.

Um die Themen aus dem Textkorpus zu extrahieren, wird ausgehend von einem generativen Modell eine Methode zur Inferenz der Themen aus den beobachteten Wörtern entwickelt.



(a) Generativer Prozess (b) Statistische Inferenz

Abbildung 2.1.1: Problemstellung der Themen Modelle nach [24].

Der generative Prozess beschreibt, wie aus Themen und den zugehörigen Termen Dokumente generiert werden. Dazu fassen Themenmodelle die Dokumente als Mischung verschiedener Themen auf. Für jedes Dokument wird eine Wahrscheinlichkeit festgelegt, mit der dieses Dokument zu einem Thema gehört. Anhand dieser Wahrscheinlichkeit wird für jedes Wort in einem Dokument ein Term aus dem Thema gezogen. In Abbildung 2.1(a) wird dieser Prozess schematisch dargestellt.

Hier werden zwei Themen dargestellt, einmal ein Thema über Filme und ein Thema über Erdbeben. Für Dokument $\mathbf{w}_1$ und $\mathbf{w}_3$ wurde jeweils festgelegt, dass sie nur aus Thema $\mathbf{z}_1$ bzw. $\mathbf{z}_2$ bestehen. Dementsprechend werden Wörter nur aus den Termen des zugehörigen Themas gezogen. Dokument $\mathbf{w}_2$ besteht jeweils zur Hälfte aus Thema $\mathbf{z}_1$ und $\mathbf{z}_2$. Es werden also Terme aus beiden Themen gezogen. Für jedes Wort wird vorher ermittelt, aus welchem Thema ein Term gezogen werden soll. Anschließend wird aus dem so bestimmten Thema ein Term gezogen. So kann ein Dokument generiert werden, das sowohl das Thema Film als auch Erdbeben beinhaltet. Die so generierten Dokumente weisen keine syntaktische Struktur auf, da vom generativen Prozess keine vollständigen Sätze erzeugt werden. Die Wörter instantiieren nur Terme aus den Themen. Dafür sind die Wahrscheinlichkeiten der Themen in diesem Dokument bekannt.

Das generative Modell erlaubt es zwar Dokumente zu bestimmten Themen zu erzeugen, die eigentliche Zielsetzung ist jedoch, für ein Textkorpus Themen zu identifizieren. Die Abbildung

2.1(b) zeigt analog zu Abbildung 2.1(a) das Problem auf: Aus Wörtern, die in Dokumenten beobachtet wurden, sollen Themen abgeleitet werden. Aus dem generativen Modell kann man mit statistischen Mitteln eine Methode entwickeln, die es erlaubt, aus den beobachteten Daten das dazu passende Modell zu finden. Im folgenden wird anhand der Latent Dirichlet Allocation (LDA) diese Methode hergeleitet.

2.1.1 Latent Dirichlet Allocation

Der Algorithmus Latent Dirichlet Allocation (LDA) gibt ein probabilistisches Modell zur Erzeugung von Dokumenten an. Anhand dieses Modells kann eine Methode zur statistischen Inferenz von Themen aus beobachteten Wörtern entwickelt werden. In Abbildung 2.1.2 wird das generative Modell als graphisch dargestellt. Im Unterschied zu anderen Themenmodellen gibt die LDA zusätzlich zur Termverteilung von Themen noch eine Themenverteilung für Dokumente an [12].

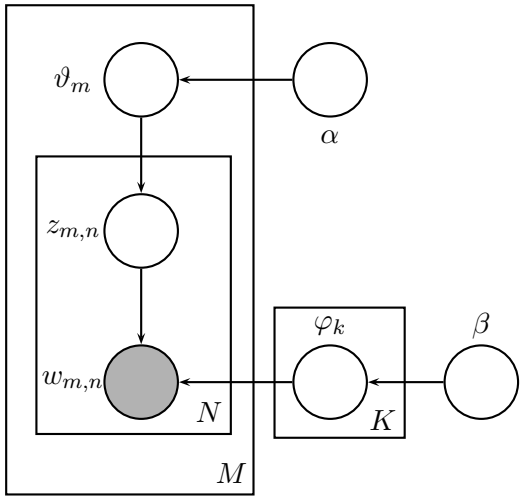


Abbildung 2.1.2: Generatives Modell für LDA nach [4].

In Abbildung 2.1.2 wird dargestellt, wie ein Dokument erzeugt wird. Die Pfeile geben Abhängigkeiten zwischen den Variablen an, während die Kacheln Wiederholungen von Variablen anzeigen. Die Buchstaben in den Kacheln geben an, wie oft die Variablen wiederholt werden. So können die Knoten für M Dokumente und $M \times N$ Wörter präziser dargestellt werden ohne die Variablen mehrfach zu notieren. Der schattierte Knoten gibt an, dass diese Variable beobachtbar ist und die leeren Knoten zeigen latente Variablen an.

Für jedes Dokument $\mathbf{w}_m$, wird eine Themenverteilung ϑ_m ermittelt. Anhand dieser wird für jedes Wort $w_{m,n}$ aus Dokument $\mathbf{w}_m$ ein Thema $z_{m,n} = k$ gezogen. Aus der Termverteilung φ_k für das Thema $z_{m,n} = k$ wird nun der Term bestimmt, der das Wort instantiiert.

Die Parameter α und β sind die Parameter der Dirichletverteilung, anhand derer die multinomialen Themenverteilungen ϑ_m und die Termverteilungen φ_k bestimmt werden. Im Weiteren wird α bzw. β synonym für $\alpha := (\alpha_1, \ldots, \alpha_k)$ und die symmetrische Variante $\alpha := \alpha_1 = \ldots = \alpha_k$

benutzt. Die Menge aller Themenverteilungen ϑ_m wird als $\Theta = \{\vartheta_m\}_{m=1}^{M}$ bezeichnet und die Menge aller Wortverteilungen φ_k als $\Phi = \{\varphi_k\}_{k=1}^{K}$.

Aus dem generativen Modell können nun verschiedene Wahrscheinlichkeiten abgeleitet werden. So ist die Wahrscheinlichkeit, dass ein Wort $w_{m,n}$ einen Term t instantiiert, gegeben eine Themenverteilung ϑ_m für das Dokument m

$$p(w_{m,n} = t|\vartheta_m, \Phi) = \sum_{k=1}^{K} p(w_{m,n} = t|\varphi_k)p(z_{m,n} = k|\vartheta_m) \tag{2.1.1}$$

Dies entspricht einer Iteration der Wortkachel, die die Knoten $z_{m,n}$ und $word_{m,n}$ enthält. Die gemeinsame Wahrscheinlichkeitsverteilung für alle beobachteten und unbeobachteten Variablen kann anhand des Modells in Abbildung 2.1.2 abgeleitet werden.

$$p(\mathbf{w}_m, \mathbf{z}_m, \vartheta_m, \Phi|\alpha, \beta) = \overbrace{\underbrace{\prod_{n=1}^{N_m} p(w_{m,n}|\varphi_{z_{m,n}})p(z_{m,n}|\vartheta_m)}_{\text{Wortkachel}} \cdot p(\vartheta_m|\alpha)}^{\text{Dokumentkachel (nur 1 Dokument)}} \cdot \underbrace{p(\Phi|\beta)}_{\text{Themenkachel}} \tag{2.1.2}$$

Integriert man nun über ϑ_m und Φ und summiert über alle Themen $z_{m,n}$ erhält man mit (2.1.1), die Wahrscheinlichkeit für ein Dokument $\mathbf{w}_m$.

$$p(\mathbf{w}_m|\alpha, \beta) = \int\int p(\vartheta_m|\alpha)p(\Phi|\beta) \cdot \prod_{n=1}^{N_m} \sum_{z_{m,n}} p(w_{m,n}|\varphi_{z_{m,n}})p(z_{m,n}|\vartheta_m) d\Phi d\vartheta_m \tag{2.1.3}$$

$$= \int\int p(\vartheta_m|\alpha)p(\Phi|\beta) \cdot \prod_{n=1}^{N_m} p(w_{m,n}|\vartheta_m, \Phi) d\Phi d\vartheta_m \tag{2.1.4}$$

Die Wahrscheinlichkeit für ein komplettes Korpus erhält man, indem man M mal ein Dokument erzeugt. Die Wahrscheinlichkeit für ein Korpus kann also mit folgender Formel ausgedrückt werden.

$$p(\mathcal{W}|\alpha, \beta) = \prod_{m=1}^{M} p(\mathbf{w}_m|\alpha, \beta) \tag{2.1.5}$$

2.1.2 Inferenz mit Gibbs-Sampling

Man kann nun ausgehend von diesen Wahrscheinlichkeiten eine Methode zur Inferenz der unbeobachteten Themen entwickeln. Hierzu nutzt man einen approximativen Algorithmus, da trotz der relativen Einfachheit des Modells eine exakte Inferenz nicht möglich ist. Hier benutzen wir dazu Gibbs-Sampling [26] .

Gibbs-Sampling ist eine Monte Carlo Markov Ketten (MCMC) Simulation. Mit MCMC Methoden kann man hochdimensionale Wahrscheinlichkeitsverteilungen anhand einer Markov Kette simulieren. Dies erlaubt es, die gesuchte Verteilung $p(\mathbf{z}|\mathbf{w})$, welche Themen in welchen Dokumenten auftreten, zu berechnen. Mit exakter Inferenz ist diese Verteilung sehr schwierig zu berechnen [11]. Der Gibbs-Sampling Algorithmus berechnet die Verteilung nicht exakt sondern nähert die Verteilung an. Der Algorithmus funktioniert wie folgt.

Sei $p(w, z)$ eine bivariate Zufallsvariable. Wir wollen $p(z)$ bzw. $p(w)$ berechnen. Anstatt nun $\int p(w, z)dw$ bzw. $\int p(w, z)dz$ direkt zu berechnen, berechnet der Gibbs-Sampler alternierende Sequenzen von $p(w|z)$ und $p(z|w)$. Das Sampling wird mit einem zufälligen Wert für z_0 gestartet und sampelt w_0 anhand $p(w|z = z_0)$. Dann wird w_0 dazu benutzt z_1 anhand der bedingten Verteilung $p(z|w = w_0)$ zu ermitteln. Das Sampling wird dann nach folgender Formel fortgesetzt:

$$w_i \approx p(w|z = z_{i-1})$$
$$z_i \approx p(z|w = w_i)$$

Nach k-maliger Wiederholung konvergiert die Sequenz gegen die tatsächliche Verteilung.

Für den multivariaten Fall wird die bedingte Wahrscheinlichkeit erweitert, indem aus dem Variablenvektor $\mathbf{w}$ die k-te Variable aus der Verteilung $p(w_i|\mathbf{w}_{\neg i})$ gezogen wird. Wobei $\mathbf{w}_{\neg i} = (w_1, \ldots, w_{i-1}, w_{i+1}, \ldots, w_k)$ ist.

Für die LDA wollen wir nun die Verteilung $p(\mathbf{z}|\mathbf{w})$ abschätzen. Um dies berechnen zu können, müssen wir die gemeinsame Verteilung $p(\mathbf{z}, \mathbf{w})$ bzw. $p(\mathbf{w})$ kennen (siehe Gleichung 2.1.6)

$$p(\mathbf{z}|\mathbf{w}) = \frac{p(\mathbf{w}, \mathbf{z})}{p(\mathbf{w})} = \frac{\prod_{i=1}^{W} p(w_i|z_i)}{\prod_{i=1}^{W} \sum_{k=1}^{K} p(z_i = k, w_i)} \tag{2.1.6}$$

Hier hilft uns die Gibbs-Sampling Prozedur, die es erlaubt, $p(\mathbf{z}|\mathbf{w})$ direkt zu berechnen, ohne über K^W Terme zu summieren. Der Gibbs-Sampler simuliert die Verteilung $p(\mathbf{z}|\mathbf{w})$ durch

$$p(z_i|\mathbf{z}_{\neg i}, \mathbf{w}) = \frac{p(\mathbf{w}, \mathbf{z})}{p(\mathbf{z}_{\neg i}, \mathbf{w})} \tag{2.1.7}$$

Hierzu müssen wir die gemeinsame Verteilung $p(\mathbf{w}, \mathbf{z})$ kennen. Im Falle der LDA kann diese Verteilung aufgespalten werden in

$$p(\mathbf{w}, \mathbf{z}|\alpha, \beta) = p(\mathbf{w}|\mathbf{z}, \beta) \cdot p(\mathbf{z}|\alpha), \tag{2.1.8}$$

da $\mathbf{w}$ unabhängig von α ist und $\mathbf{z}$ unabhängig von β.

Die Faktoren können einzeln hergeleitet werden. So ist

$$\begin{aligned} p(\mathbf{w}|\mathbf{z},\beta) &= \int p(\mathbf{w}|\mathbf{z},\Phi)p(\Phi|\beta)d\Phi \\ &= \int \prod_{z=1}^{K} \frac{1}{\Delta(\beta)} \prod_{t=1}^{V} \varphi_{z,t}^{n_z^{(t)}+\beta_t-1} d\varphi_z \qquad (2.1.9) \\ &= \prod_{z=1}^{K} \frac{\Delta(\mathbf{n}_z+\beta)}{\Delta(\beta)}, \quad \mathbf{n}_z = \left\{n_z^{(t)}\right\}_{t=1}^{V} \qquad (2.1.10) \end{aligned}$$

Hier wird benutzt, dass $p(\mathbf{w}|\mathbf{z},\Phi)$ multinomial und $p(\Phi|\beta)$ dirichlet verteilt ist. Unter der Annahme, dass die Reihenfolge der Wörter in einem Dokument unerheblich ist, gilt:

$$p(\mathbf{w}|\mathbf{z},\Phi) = \prod_{k=1}^{K} \prod_{i:z_i=k} p(w_i=t|z_i=k) = \prod_{k=1}^{K} \prod_{t=1}^{V} \varphi_{k,t}^{n_k^{(t)}}$$

und

$$Dir(\varphi_k|\beta) = \frac{1}{\Delta(\beta)} \prod_{t=1}^{V} \varphi_{k,t}^{\beta_t-1} \text{ mit } \Delta(\beta) = \frac{\prod_{k=1}^{dim\beta} \Gamma(\beta_k)}{\Gamma(\sum_{k=1}^{dim\beta} \beta_k)}$$

Aus den beiden vorhergehenden Gleichungen ergibt sich 2.1.9. Die Anwendung des Dirichletintegrals auf Gleichung 2.1.9 ergibt dann Gleichung 2.1.10.

Analog kann $p(\mathbf{z}|\beta)$ hergeleitet werden. Es ist:

$$\begin{aligned} p(\mathbf{z}|\alpha) &= \int p(\mathbf{z}|\Theta)p(\Theta|\alpha)d\Theta \\ &= \int \prod_{m=1}^{M} \frac{1}{\Delta(\alpha)} \prod_{k=1}^{K} \vartheta_{m,k}^{n_m^{(k)}+\alpha_k-1} d\vartheta_m \qquad (2.1.11) \\ &= \prod_{m=1}^{M} \frac{\Delta(\mathbf{n}_m+\alpha)}{\Delta(\alpha)}, \quad \mathbf{n}_m = \left\{n_m^{(k)}\right\}_{k=1}^{K} \qquad (2.1.12) \end{aligned}$$

Die gemeinsame Verteilung $p(\mathbf{w},\mathbf{z}|\alpha,\beta)$ ergibt sich aus 2.1.12 und 2.1.10.

$$p(\mathbf{w},\mathbf{z}|\alpha,\beta) = \prod_{z=1}^{K} \frac{\Delta(\mathbf{n}_z+\beta)}{\Delta(\beta)} \cdot \prod_{m=1}^{M} \frac{\Delta(\mathbf{n}_m+\alpha)}{\Delta(\alpha)}$$

Wenn man nun die gemeinsame Verteilung einsetzt, ergibt sich die Update-Gleichung für den Gibbs-Sampler.

$$p(z_i = k|\mathbf{z}_{\neg i}, \mathbf{w}) = \frac{p(\mathbf{w}, \mathbf{z})}{p(\mathbf{w}, \mathbf{z}_{\neg i})} = \frac{p(\mathbf{w}, \mathbf{z})}{p(\mathbf{w}_{\neg i}, \mathbf{z}_{\neg i})p(w_i)} \frac{p(\mathbf{z})}{p(\mathbf{z}_{\neg i})} \tag{2.1.13}$$

$$\propto \frac{\Delta(\mathbf{n}_z + \beta)}{\Delta(\mathbf{n}_{z,\neg i} + \beta)} \cdot \frac{\Delta(\mathbf{n}_m + \alpha)}{\Delta(\mathbf{n}_{m.\neg i} + \alpha)} \tag{2.1.14}$$

$$\propto \ldots$$

$$\propto \frac{n_{k,\neg i}^{(t)} + \beta}{\sum_{t=1}^{V} n_{k,\neg i}^{(t)} + \beta} \cdot \frac{n_{m,\neg i}^{(k)} + \alpha}{\sum_{k=1}^{K} n_{m,\neg i}^{(k)} + \alpha} \tag{2.1.15}$$

Die Update-Gleichung kann nun dazu benutzt werden, die Themen für Wörter in Dokumenten zu bestimmen. Es wird über alle Wörter in den Dokumenten iteriert und für jedes Wort anhand $p(z_i = k|\mathbf{z}_{\neg i}, \mathbf{w})$ ein neues Thema bestimmt. Dies wird solange wiederholt, bis die Themenverteilung stabil bleibt, bzw. der Gibbs-Sampler einen stabilen Zustand erreicht hat. Die Verteilungen ϑ_m und φ_k können direkt aus den Gleichungen 2.1.10 und 2.1.12 abgeleitet werden. So hat man nun ein Modell, welches Dokumente bzw. Wörter in Themen gruppieren kann. Genaueres zur Herleitung und Gibbs-Sampling im Kontext der LDA kann in [10] nachgelesen werden.

2.1.3 Inferenz für ungesehene Dokumente

Ausgehend vom gelernten Modell will man nun für neue Dokumente die Themen herausfinden. Sei $\tilde{\mathbf{w}}$ ein neues Dokument. Für dieses neue Dokument lassen wir wiederum den Gibbs-Sampler die Themen bestimmen. Anstatt $p(z_i = k|\mathbf{z}_{\neg i}, \mathbf{w})$ bestimmen wir aber $p(\tilde{z}_i = k|\tilde{\mathbf{z}}_{\neg i}, \tilde{\mathbf{w}}, \Phi, \Theta)$, da wir die Themen für das neue Dokument anhand der schon ermittelten Themen- und Termverteilungen inferieren wollen. Es ergibt sich folgende Update-Gleichung

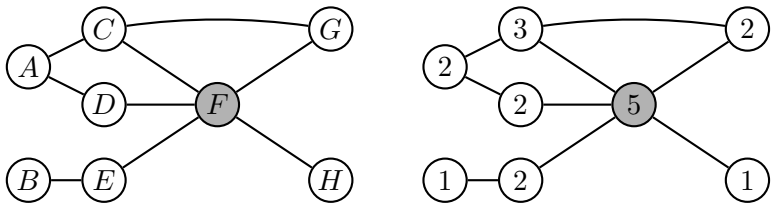
$$p(\tilde{z}_i = k|\tilde{\mathbf{z}}_{\neg i}, \tilde{\mathbf{w}}, \Phi, \Theta) = \frac{n_k^{(t)} + \tilde{n}_{k,\neg i}^{(t)} + \beta}{\sum_{t=1}^{V} n_k^{(t)} + \tilde{n}_{k,\neg i}^{(t)} + \beta} \cdot \frac{n_{\tilde{m},\neg i}^{(k)} + \alpha}{\sum_{k=1}^{K} n_{\tilde{m},\neg i}^{(k)} + \alpha} \tag{2.1.16}$$

wobei $\tilde{n}_k^{(t)}$ die Anzahl des Auftreten von Term t mit Thema k für das neue Dokument $\tilde{\mathbf{w}}$ bezeichnet. Analog zu normaler Inferenz können auch hier die Dokumentverteilung ϑ_m und die Termverteilung φ_k bestimmt werden. So können für ein neues Dokument die vorherrschenden Themen bestimmt werden.

Ein Problem sind Terme, die vorher nicht aufgetreten sind. Diese erscheinen in den Termverteilungen der Themen nicht und können somit nicht dazu beitragen, Themen zu inferieren. Sie fließen somit auch nicht in die Updategleichung 2.1.16 mit ein. Im Falle der Inferenz werden sie einfach übersprungen.

2.2 Zentralitätsindizes

Ein Zentralitätsindex ist ein Maß, das intuitiv die Wichtigkeit eines Knoten oder einer Kante in einem Graphen wiedergibt. Betrachtet man den Graphen in Abbildung 2.1(a) als Straßennetzwerk, welches verschiedene Knotenpunkte miteinander verbindet und möchte herausfinden, welcher Knotenpunkt der am meisten belastete ist, kann dies durch ein Zentralitätsmaß erfasst werden. Ein Zentralitätsmaß weist jedem Knoten einen Wert zu, der die Wichtigkeit dieses Knoten widerspiegelt. So ist in der Abbildung der Knoten F augenscheinlich der Knoten, der am meisten belastet wird. Zählt man nun für jeden Knoten die Anzahl der Kanten und benutzt den ermittelten Wert als Zentralitätsmaß, ergeben sich die Zentralitätswerte in Abbildung 2.1(b). Das Zentralitätsmaß gibt das Erwartete wieder, da dem Knoten F der größte Wert (5) zugewiesen wird.



(a) Straßennetzwerk (b) Straßennetzwerk mit Zentralitäten

Abbildung 2.2.1: Zentralitäten in einem Graphen

Es gibt noch viele anderen Zentralitätsmaße, die verschiedene Aspekte betrachten. Im Folgenden werden einige der für die Arbeit geeignet erscheinenden Zentralitätsmaße definiert und untersucht. Dazu benötigen wir zunächst einige wichtige Definitionen aus der Graphentheorie.

Definition: Ein Graph G ist ein Tripel (V, E, ω), wobei $V = \{v_1, \ldots, v_n\}$ eine Menge von Knoten ist, $E = \{e_1, \ldots, e_m\}$ eine Menge von Kanten und $\omega : E \to \mathbb{R}$ eine Funktion, die jeder Kante $e \in E$ eine reelle Zahl zuordnet.

Es kann zwischen gerichteten Graphen und ungerichteten Graphen unterschieden werden. Im ungerichteten Fall gilt

$$E \subseteq \{\{v, u\} | v, u \in V\},$$

d.h. die Kante $\{u, v\}$ ist dieselbe wie $\{v, u\}$, da die Kanten als Mengen betrachtet werden und diese nicht sortiert sind. Im gerichteten Fall gilt

$$E \subseteq \{(u, v) | u, v \in V\},$$

hier wird zwischen $(u, v) \in E$ und $(v, u) \in E$ unterschieden, da Paare sortiert sind.

Der Grad eines Knoten wird durch die Anzahl der eingehenden Kanten bzw. der ausgehenden Kanten bestimmt. Es ist $o(v)$ die Menge der ausgehenden Kanten eines Knoten v. Analog ist $i(v)$ die Menge der eingehenden Kanten. Im ungerichteten Fall gilt $o(v) = i(v)$.

Ein Pfad π ist definiert als eine Folge von Knoten $(v_1, \ldots, v_i, \ldots, v_n)$, wobei für alle $i, j \in \{1, \ldots, n\}$ gilt, dass $v_i \neq v_j$ falls $i \neq j$.

Die Distanz zwischen zwei Knoten u,v wird mit $d(u, v)$ bezeichnet. Die Distanz zwischen zwei Knoten wird als die Summe der Gewichte auf dem Pfad von u nach v definiert. Im Falle, dass kein Pfad von u nach v existiert, gilt $d(u, v) = \infty$ und es gilt $d(u, u) = 0$.

Eine formale Definition für Zentralitätsindizes wurde bis jetzt nicht aufgestellt. In Brandes und Erlebach [5] wird versucht, eine formalen Definition anzugeben. Dazu wird ein Zentralitätsindex als struktureller Index aufgefasst, der eine Halbordnung auf die Knoten bzw. Kanten des Graphen induziert.

Definition: Seien $G = (V_G, E_G)$ und $H = (V_H, E_H)$ gewichtete Graphen und Φ der Isomorphismus zwischen G und H. Es bezeichne X die Knotenmenge V_G bzw. die Kantenmenge E_G. Dann ist s ein struktureller Index genau dann, wenn $\forall x \in X : G \simeq H \Rightarrow s_G(x) = s_H(\Phi(x))$

Ein Zentralitätsindex muss ein struktureller Index sein, wobei nicht jeder strukturelle Index ein Zentralitätsindex ist. Durch die induzierte Halbordnung können wir Aussagen bzgl. eines Zentralitätsindex c_X treffen, wie $x \in V$ ist mindestens so zentral wie $y \in V$ wenn $c_X(x) \geq c_X(y)$.

Für die einzelnen Klassen der Zentralitätsindizes wird in Brandes und Erlebach [5] versucht, eine axiomatische Definition herzuleiten. Diese hier wiederzugeben, würde den Fokus der Diplomarbeit verlassen.

2.2.1 Degree-Zentralität

Degree-Zentralität ist die einfachste Art der Zentralität. Die Zentralität der einzelnen Knoten wird anhand der Anzahl der ein- und ausgehenden Kanten bestimmt. Für gerichtete Graphen kann zwischen eingehenden und ausgehenden Kanten unterschieden werden. So ist die eingehende Degree-Zentralität definiert als

$$c_{iD}(v) := |i(v)|$$

und die ausgehende Degree-Zentralität als

$$c_{oD}(v) := |o(v)|$$

definiert. Im ungerichteten Fall gilt

$$c_D(v) := |o(v)| \text{ oder } c_D(v) := |i(v)|.$$

Ein weiterer Zentralitätsindex bezieht zusätzlich zur Anzahl der Kanten noch die Gewichtung der Kanten mit ein. Die Shaffer-Zentralität c_{SH} eines Knoten v ist die Wurzel der Summe der quadrierten Gewichte der ein- bzw. ausgehenden Kanten.

$$c_{oSH}(v) = \sqrt{\sum_{u \in o(v)} \omega((v,u))^2} \text{ bzw. } c_{iSH}(v) = \sqrt{\sum_{u \in i(v)} \omega((u,v))^2}$$

Der Vorteil der Degree-Zentralität Indizes ist die Unabhängigkeit von der Struktur des Graphen. Auch wenn der Graph nicht vollständig verbunden ist, können alle Knoten bewertet werden. Dies ist möglich, da die Knoten nur lokal betrachtet werden und die globale Struktur des Graphen außer Acht gelassen wird. Das kann je nach Anwendung ausreichend sein. Betrachtet man jedoch zum Beispiel, das Problem eine industrielle Anlage möglichst nah an allen Zulieferern aufzustellen, muss man die globale Struktur des Graphen berücksichtigen. Dazu eignen sich die folgenden Indizes besser.

2.2.2 Distanz

Distanzbasierte Zentralitätsindizes berechnen die Zentralität anhand der Distanz zwischen den Knoten im Graphen. Bei distanzbasierten Zentralitäten geht es oft darum, eine öffentliche Einrichtungen in einem bestehenden Straßennetz zu platzieren, so dass bestimmte Kriterien erfüllt sind.

Eccentricity-Zentralität

Ein Anwendungsbeispiel für die Eccentricity-Zentralität ist es, ein Krankenhaus zu platzieren, so dass die maximale Distanz zu allen Haushalten minimiert wird. Hierzu kann man das Exzentrizität-Zentralitätsmaß nutzen. Dieser Index berechnet, inwieweit ein Knoten im "Inneren" des Graphen liegt und bestimmt die Zentralität anhand der Exzentrizität der Knoten. Die Exzentrizität $e(v)$ eines Knoten v ist definiert als der längste Pfad zu allen anderen Knoten u. D.h. $e(v) := max\{d(u,v)|u \in V\}$. Knoten, die in wenigen Schritten alle anderen Knoten erreichen können und im "Inneren" des Graphen liegen, sind somit zentraler als solche die weiter "außen" liegen. Die Eccentricity-Zentralität wird als die reziproke Exzentrizität definiert:

$$c_E(v) := \frac{1}{e(v)}$$

Die Eccentricity-Zentralität in der obigen Formulierung weist die Schwäche auf, dass sie nur auf ungerichteten und zusammenhängenden Graphen definiert ist. Ist der Graph gerichtet oder nicht zusammenhängend kann es passieren, dass zwei Knoten nicht verbunden sind. Da für zwei Knoten u, v, die nicht verbunden sind, gilt, dass $d(v,u) = \infty$, ergibt die Exzentrizität $e(v) = \infty$ für alle Knoten v. Daraus folgt, dass die Zentralität c_E nicht mehr definiert ist.

Closeness-Zentralität

Als Beispiel betrachte man das Problem einen Supermarkt zu bauen, so dass die totale Entfernung zu allen möglichen Kunden minimiert wird. Die Closeness-Zentralität kann bei der Lösung dieses Problem helfen. Sie wird als die Summe der Abstände zu allen anderen Knoten des Graphen definiert. Knoten mit einem niedrigen Closeness-Wert sind von allen anderen Knoten leichter erreichbar als Knoten mit einem hohen Wert. Entsprechend sind Knoten mit einem niedrigen Closenesswert zentraler als solche mit hohem Wert. Die Closeness-Zentralität wird dementsprechend analog zur Eccentricity-Zentralität definiert. Als reziproker Wert der Summe aller Pfade von v zu allen anderen Knoten u.

$$c_C(v) := \frac{1}{\sum_{u \in V} d(u, v)}.$$

Der Supermarkt muss also an dem Knoten aufgestellt werden, der die höchste Closeness-Zentralität besitzt.

Valente-Foreman-Closeness-Zentralität

Die Valente-Foreman-Closeness-Zentralität ist ein Abwandlung der Closeness-Zentralität. Sie ist definiert als

$$c_R(v) := \frac{\sum_{u \in V} \Delta_G + 1 - d(v, u)}{n - 1}$$

wobei Δ_G der Durchmesser des Graphen (der längste Pfad im Graphen) ist und $n := |V|$. Im Unterschied zur Closeness-Zentralität wird nicht der reziproke Wert der Closeness genommen, sondern die Distanz invertiert und so die entstehenden Werte gemittelt.

Auch hier gilt, dass die Closeness-Zentralität und die Valente-Foreman-Closeness-Zentralität aus den gleichen Gründen wie die Eccentricity-Zentralität für nicht zusammenhängende oder Spezialfälle von gerichteten Graphen nicht definiert sind.

2.2.3 Shortest-Path

Shortest-Path Zentralitätsindizes für Knoten berechnen die Zentralität anhand der Anzahl der kürzesten Pfade, die durch einen Knoten führen. Betrachtet man wieder ein Straßennetz, in dem die Knoten Mautstationen repräsentieren und möchte wissen, durch welche Mautstation die meisten Fahrzeuge fahren, kann man Shortest-Path-Zentralitäten benutzen. Diese geben wieder, wie viel Arbeit ein Knoten leisten muss, bzw. wie viele Fahrzeuge durch die Mautstation geschleust werden.

Stress-Zentralität

Die Stress-Zentralität misst, wie viel kürzeste Pfade durch einen Knoten im Graphen laufen. Zur Berechnung werden für einen Knoten v die Anzahl der kürzesten Wege, die von einem Knoten s zu einem Knoten t führen, aufsummiert. Die Formel der Zentralität lautet:

$$c_S(v) := \sum_{s \neq v \in V} \sum_{t \neq v \in V} \sigma_{st}(v)$$

$\sigma_{st}(v)$ gibt die Anzahl der kürzesten Pfade von s nach t über v an.

Dieser Index funktioniert für gerichtete und ungerichtete Graphen mit Kantengewichten. Definiert man $\sigma_{st}(v) = 0$, wenn es keinen Pfad von s nach t über v gibt, dann funktioniert die Stress-Zentralität auch für nicht zusammenhängende Graphen bzw. für spezielle Konfigurationen von gerichteten Graphen.

Betweenness -Zentralität

Betweenness-Zentralität kann als eine Art normalisierte Stress-Zentralität betrachtet werden. Hier wird die Anzahl der kürzesten Pfade von s nach t über v durch die Anzahl aller kürzesten Pfade von s nach t dividiert und somit relativiert. Die Formel ist analog zur Formel der Stress-Zentralität

$$c_B(v) := \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Hierbei bezeichnet σ_{st} die Anzahl der kürzesten Pfade von s nach t. Im Gegensatz zur Stress-Zentralität wird nicht die absolute Anzahl der kürzesten Pfade durch einen Knoten gezählt, sondern die relative Anzahl. So können Knoten, die eine hohe Last haben, besser bewertet werden (siehe [5] S. 30 Abb. 3.6). Im Gegensatz zur Stress-Zentralität funktioniert die Betweenness-Zentralität nicht auf unzusammenhängenden Graphen. Auch hier gilt wieder: Sobald kein Pfad zwischen zwei Knoten $s, t \in V$ existiert, ist die Betweenness-Zentralität nicht mehr definiert.

2.2.4 Feedback-Zentralitäten

Die Feedback Zentralitäten wurden zuerst bei der Bewertung von Webseiten benutzt [19, 14]. Die Idee ist, dass eine Webseite hoch bewertet wird, wenn viele andere gut bewertete Webseiten auf diese verlinken. Die Bewertung bzw. die Zentralität wird an die benachbarten Seiten weitergegeben. So können Seiten, auf die zwar nicht oft verlinkt wird, trotzdem gut bewertet werden, wenn die wenigen verlinkenden Seiten eine hohe Bewertung aufweisen. Zusätzlich werden Seiten mit vielen eingehenden Links besser bewertet.

PageRank

PageRank ist eine solche Feedback-Zentralität [19]. Die Zentralität eines Knoten wird anhand der Zentralitäten der benachbarten Knoten berechnet. Es gilt folgendes zu berechnen.

$$c_{PR}(v) = d \sum_{u \in i(v)} \frac{c_{PR}(u)}{|o(u)|} + (1 - d) \tag{2.2.1}$$

Für jeden Knoten v wird die Zentralität anhand der Summe der Zentralitäten der Knoten u, die auf v zeigen, gewichtet mit der Anzahl der ausgehenden Kanten von u, berechnet. Fasst man nun die Zentralitäten aller Knoten $v_i, \quad i \in \{1, \ldots, N\}$ in einem Vektor zusammen, so dass

$$\mathbf{c_{PR}} := (c_{PR}(v_1), \ldots, c_{PR}(v_N))$$

gilt, kann man die obige Formel auch in Matrixschreibweise darstellen.

$$\mathbf{c_{PR}} = dP\mathbf{c_{PR}} + (1 - d)\mathbf{1_N} \tag{2.2.2}$$

Wobei die Übergangsmatrix P definiert ist als

$$p_{ij} := \begin{cases} \frac{1}{|o(j)|} & \text{wenn } (j, i) \in E \\ 0 & \text{sonst} \end{cases}.$$

Das rekursive Gleichungssystem in 2.2.2 kann durch eine Jacobi-Iteration gelöst werden [5]. Mit den richtigen Werten initialisiert und mit $0 \leq d < 1$ konvergiert das Gleichungssystem gegen den gesuchten Wert $\mathbf{c_{PR}}^*$. Sei $\mathbf{c_{PR}}^0 := 1^N$ der Startwert für den Zentralitätsvektor $\mathbf{c_{PR}}$, dann ist

$$\mathbf{c_{PR}}^i = dP\mathbf{c_{PR}}^{i-1} + (1 - d)\mathbf{1_N}$$

und es gilt

$$\lim_{i \to \infty} \mathbf{c_{PR}}^i = \mathbf{c_{PR}}^*.$$

Durch die Konstruktion der Übergangsmatrix P wird die Gewichtung der Kanten nicht betrachtet und es wird nur die Struktur des Graphen und die Zentralität der benachbarten Knoten berücksichtigt. Verwendet man die Matrixschreibweise zur Berechnung der Zentralität, können auch nicht zusammenhängende Graphen betrachtet werden. So ist für den Graphen in Abbildung 2.2.2 die Übergangsmatrix definiert als

$$P = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Abbildung 2.2.2: Nicht verbundener Graph

und die Zentralitäten für u und v wären Null. Die rekursive Formel 2.2.1 wäre jedoch nicht mehr definiert, da durch Null geteilt würde.

Hits

Hits ist wie PageRank zuerst zur Bewertung von Internetseiten verwendet worden [14]. Der Algorithmus kann jedoch auch als Zentralitätsmaß verwendet werden. Traditionell wird aus einem Grundstock von Webseiten, passend zu einer Anfrage, ein Graph erstellt, auf den dann Hits angewendet wird. Im Gegensatz zu PageRank berechnet Hits für jeden Knoten zwei Werte. Einmal den Authority-Wert, der angibt, wie gut der Knoten die Anfrage widerspiegelt. Dies wird anhand der Hub-Gewichte der Knoten, auf die die Authority zeigt, berechnet. Zweitens das Hub-Gewicht, welches angibt, auf wie viele wichtige Knoten verlinkt wird. Das Hub-Gewicht wird anhand der Authority-Gewichte der Knoten berechnet, auf die der Hub zeigt.

Mathematisch kann das Authority-Gewicht folgendermaßen berechnet werden

$$c_{H_a}(v_i) = \sum_{k=1}^{n} A_{ik}^T c_{H_h}(v_k) \tag{2.2.3}$$

und das Hub-Gewicht folgendermaßen

$$c_{H_h}(v_i) = \sum_{j=1}^{n} A_{ik} c_{H_a}(v_j). \tag{2.2.4}$$

Wobei A die Adjazenzmatrix des Graphen ist, c_{H_a} die Authority-Gewichtung und c_{H_h} die Hub-Gewichtung. Substituiert man $c_{H_h}(v_k)$ mit $c_{H_a}(v_k)$ in Gleichung 2.2.3 und $c_{H_h}(v_k)$ mit $c_{H_a}(v_k)$ in Gleichung 2.2.4, erhält man die Formeln in Matrixschreibweise. So lässt sich die Authority-Gewichtung durch Gleichung 2.2.5 ausdrücken und die Hub-Gewichtung durch Gleichung 2.2.6.

$$\mathbf{c_{H_a}} = (A^T A)\mathbf{c_{H_a}} \tag{2.2.5}$$

$$\mathbf{c_{H_h}} = (AA^T)\mathbf{c_{H_h}} \tag{2.2.6}$$

Die $\mathbf{c_{H_a}}$ bzw. $\mathbf{c_{H_h}}$ bezeichnen dabei wieder die Vektoren der Zentralität. Als Zentralitätsindex kann, je nach Anwendung, sowohl die Authority-Gewichtung als auch die Hub-Gewichtung benutzt werden. Die Authority-Gewichtung eignet sich, wenn Knoten nach Anzahl der eingehenden Kanten und der Hubgewichtung der Knoten, die auf den Authority-Knoten zeigen, bewertet wer-

den sollen. Die Hub-Gewichtung eignet sich, wenn Knoten nach Anzahl der ausgehenden Kanten und der Authority-Gewichtung der Knoten, auf die die Kanten zeigen, bewertet werden sollen.

2.3 Zusammenfassung

In den letzten beiden Abschnitten wurden die in der Diplomarbeit verwendeten, grundlegenden Techniken vorgestellt. Es wurde gezeigt, wie aus Texten Themen extrahiert werden und es wurden die verwendeten Zentralitätsindizes vorgestellt. Im Abschnitt über Themenmodelle wurde die verwendete mathematische Notation für Korpora, Dokumente, Wörter und Themenverteilungen eingeführt. Es wurde mathematisch hergeleitet, wie der Algorithmus zur Erkennung von Themen funktioniert und auch wie er implementiert werden kann. Spezielles Augenmerk wurde auf die statistische Inferenz von Themen aus Texten und die Inferenz von Themen für neue Texte gelegt, da dies die Hauptanwendung der Diplomarbeit ist. Es wurde aber auch kurz auf das generative Modell eingegangen, welches dazu benutzt werden kann, aus einem gelernten Themenmodell wiederum Dokumente zu generieren. Dies wird später bei der Evaluation der Themenverläufe eine Rolle spielen.

Im Abschnitt über Zentralitätsindizes wurden die verwendeten Zentralitätsindizes vorgestellt. Es wurde die notwendigen Definitionen für Graphen erläutert und versucht eine mathematische Definition für Zentralitätsindizes wiederzugeben. Die einzelnen Zentralitätsindizes wurden durch Beispiele für ihre Anwendung beschrieben und anschließend wurde die mathematische Formulierung dargestellt. Insbesondere wurden die Schwachstellen der Zentralitätsindizes erläutert, die später berücksichtigt werden müssen.

3 Verwandte Arbeiten

Viele andere Autoren haben schon den Versuch unternommen, aus Chatdaten Informationen zu extrahieren bzw. Themenmodelle um die Zeit zu erweitern. Einige dieser Arbeiten werden hier vorgestellt. Einerseits werden Arbeiten vorgestellt, die sich mit der Extraktion von Informationen aus Chattexten befassen. Diese Informationen sind nicht notwendigerweise automatisch ermittelte Themen und ein Teil der Arbeiten versucht auch nicht, die Zeit mit einzubeziehen. Diese Arbeiten wurden jedoch wegen der Analyse der Chattexte ausgewählt. Andererseits werden Arbeiten vorgestellt, die die Themenmodelle um die Zeit erweitern und dabei das Modell anpassen.

3.1 Analyse von Chattexten

Eine Arbeit von Tuulos und Tirri versucht automatische Themenidentifikation und soziale Netzwerke zu verbinden, um die Themen einer Diskussion stabiler zu identifizieren [25]. Hier werden jedoch die Netzwerke anhand der Chatstruktur aufgebaut. Es wird betrachtet, welcher Sprecher einen anderen Sprecher adressiert. Anhand der so erstellten sozialen Netzwerke werden den Sprechern Gewichte zugeteilt. So können für komplette Chatkanäle die Themen bestimmt werden. Der Fokus dieser Arbeit liegt jedoch darauf, die Themen einer statischen Chatkollektion zu bestimmen und diese zur Indexierung der Chattexte zu nutzen. Die Chattexte werden hier nicht in zeitlichen Abschnitten betrachtet, sondern als Ganzes. Auch werden die Graphen anhand der sozialen Struktur der Sprecher bestimmt und nicht anhand der Kookkurrenz der Themen.

Ein vergleichbares Retrieval System für Nachrichten in einem Internetforum auf der Basis von Themenmodellen wurde von Kim et. al entwickelt [13]. Es werden für neue Nachrichten, die ein Student in das Forum postet, zusätzlich automatische Antworten aus dem Korpus alter Nachrichten herausgesucht. Dazu werden Nachrichten als Termvektoren von vorher definierten technischen und aufgabenspezifischen Termen modelliert. Ähnliche Nachrichten werden dann anhand des Kosinus-Maßes bestimmt, wobei die Nachrichten vorher nach Themen klassifiziert wurden und nur Nachrichten in die Ergebnismenge aufgenommen werden, die Übereinstimmungen in den Themen aufweisen [8]. Diese Arbeit zeigt, wie Themenmodelle und Forentexte dazu verwendet werden können, Schüler und Studenten in ihrem Lernen zu unterstützen.

Beide Arbeiten analysieren Chattexte nach vorkommenden Themen. Diese extrahierte Information wird aber nur zur Indizierung der Nachrichtenkanäle [25] bzw. zum Auffinden automatischer Antworten genutzt [13]. Es wird keine Information über die Zeit dargestellt, sondern

die Chattexte werden als statische Sammlung von Texten betrachtet, die für spätere Suchen aufbereitet wurden.

Die folgenden Arbeiten sind näher mit dem Thema der Diplomarbeit verwandt. Hier werden die Veränderungen über die Zeit betrachtet. Es werden jedoch keine Themen identifiziert, sondern die Nachrichten in kommunikative Klassen eingeteilt. So werden in Anjewierden u. a. [2] die Chatnachrichten nach Funktionen klassifiziert. Dabei werden regulative, domänenspezifische, soziale und technische Funktionen unterschieden. Die Nachrichten werden dann in diese vier Klassen eingeteilt, indem erstens nur die vorkommenden Wörter betrachtet werden und in einem zweiten Schritt syntaktische Muster. Dazu werden die Nachrichten mit Part of Speech-Tags (POS-Tags) versehen und nach Mustern in den POS-Tagsequenzen klassifiziert. POS-Tags geben die syntaktische Funktion eines Wortes wieder. Es wird annotiert, ob ein Wort ein Verb oder ein Nomen ist oder eine andere Funktion besitzt. Wie in der Diplomarbeit werden zuerst die Klassen gelernt und dann neue Nachrichten online klassifiziert. Die Klassifikationsergebnisse bzw. das Verhältnis der klassifizierten Nachrichten zu Gesamtnachrichten werden dem Lernenden dann in aufbereiteter Form präsentiert. Dies soll den Lernenden auf etwaige Fokuswechsel hinweisen.

Einen ähnlichen Ansatz verfolgen McLaren et. al in [21, 18, 17]. Sie versuchen durch Analyse des Chats dem Lehrer ein Instrument zur Steuerung der Diskussionen an die Hand zu geben. Dabei werden allerdings nicht die einfachen Chattexte betrachtet, sondern die Schüler nutzen ein graphisches Werkzeug um die Art ihrer Beiträge zu kennzeichnen und durch Verlinkung die Antwortstruktur darzustellen. So wird die Struktur der Diskussion als Graph dargestellt. Anhand dieser zusätzlichen Informationen werden Interaktionsmuster klassifiziert und diese dem Lehrer zugänglich gemacht. Die These ist, dass bestimmte Interaktionsmuster eine Intervention des Lehrers erforderlich machen, um die Diskussion zu steuern. Diese Arbeit hat einen ähnlichen Fokus wie die vorhergehende Arbeit [13]. Es wird dem Lehrer ein Werkzeug an die Hand gegeben, um Diskussionen zu steuern und zu überwachen.

Die Anwendung, die in dieser Diplomarbeit entwickelt wird, soll im SCY-Projekt ähnliche Hilfestellung geben, wie die beiden Anwendungen die in Anjewierden u. a. [2] und Miksatko und Mclaren [18] entwickelt wurden. Einerseits soll dem Lehrer eine Möglichkeit gegeben werden Kommunikationsmuster zu erkennen und nötigenfalls in die Diskussion eingreifen, andererseits kann dem Lernenden direktes Feedback über die diskutierten Themen gegeben werden, um so eine eventuelle Selbstregulation zu bewirken.

In der Arbeit von Shaffer et. al. [22] werden Schüler in die Rolle von Städteplanern versetzt und müssen eine Wohngegend umgestalten, bzw. deren weitere Nutzung planen. Dabei werden ihnen reale Daten und Szenarien präsentiert, auf deren Grundlage sie dann Entscheidungen treffen müssen. Da die Arbeitsumgebung als Webapplikation modelliert wird, können die Aktionen der Schüler aufgezeichnet werden. Die Idee dahinter ist, dass Schüler, die solch ein Szenario bearbeiten, in verschiedene Rollen, wie zum Beispiel Ingenieur, Architekt oder Stadtplaner schlüpfen.

Aus den anfallenden Daten soll dann die eingenommene Rolle ermittelt werden. Das wird anhand der transkribierten Unterhaltungen der Schüler analysiert. Insbesondere wird die Veränderung der Rollen über die Zeit betrachtet. Dazu werden, wie in der Diplomarbeit, mehrere Chatnachrichten in diskrete zeitliche Abschnitte zusammengefasst. Die einzelnen Nachrichten werden dann von Experten in die Rollenklassen eingeordnet. Diese Funktionsklassen werden dann als Knoten in einem Graphen dargestellt und je nach Auftreten in einem Zeitabschnitt durch Kanten verbunden. Je nach Anzahl der Auftreten der Rollenklassen, wird die Länge der Kanten zwischen den Knoten festgelegt. So werden Knoten, deren Rollen oft in einem Zeitabschnitt auftreten, näher beieinander liegen als solche die nicht oft zusammen auftreten. Die dabei entstehenden Graphen werden mit verschiedenen Metriken bewertet und die Werte über die Zeit aufgetragen. Insbesondere wird ein speziell entwickelter Zentralitätsindex (siehe Abschnitt 2.2.1) dazu benutzt, die Knoten bzw. die repräsentierte Rolle zu bewerten.

Meine Diplomarbeit baut direkt auf der Idee von Shaffer auf. Wie schon kurz beschrieben wurde, werden auch hier die Nachrichten in zeitliche Abschnitte aufgeteilt und quasi klassifiziert. Im Falle der Diplomarbeit werden, anstatt die Nachrichten manuell zu klassifizieren, die Themen automatisch ermittelt; die Themen ersetzen die Rollenklassen. Die ermittelten Themen werden ähnlich weiterverarbeitet wie in der Arbeit von Shaffer, es werden allerdings Erweiterungen im Aufbau des Graphen erprobt und unterschiedliche Zentralitätsindizes auf ihre Eignung für den speziellen Anwendungsfall der Diplomarbeit geprüft.

3.2 Erweiterungen von Themenmodellen auf Zeit

Die in diesem Abschnitt vorgestellten Arbeiten beschäftigen sich nicht mehr mit Chattexten, sondern versuchen - wie auch in der Diplomarbeit verwendete - LDA-Modell mit einer zeitlichen Komponente zu erweitern. Linstead et. .al [15] verfolgt dabei den einfachsten Ansatz, in welchem für aufeinander folgende Versionen eines Softwaresystems die Themen gelernt werden und gezählt wird, wie oft die Themen in einer Version auftreten. Für die einzelnen Themen wird dann aufgetragen, wie oft sie in den einzelnen Versionen vorkommen. Dabei wird aber nur die absolute Anzahl der auftretenden Themen betrachtet und die Wahrscheinlichkeit eines Themas unberücksichtigt gelassen. Außerdem eignet sich dieser Ansatz nicht für die Diplomarbeit, da jedes mal ein neues Modell gelernt wird und dies erstens einen erheblichen Rechenaufwand darstellt und zweitens die Themen verschiedener gelernter Modelle einander wieder zugeordnet werden müssen.

Eine weitere Anwendung, die ohne aufwändige Anpassung des LDA-Modell auskommt, wird in [1] beschrieben. In dieser Arbeit wird das Modell kontinuierlich erweitert. Die Themen werden durch unbekannte Dokumente verändert. Dies wird erreicht, indem für neue Dokumente während der Inferenz der Themen die Themen- und Termverteilungen des Modells angepasst werden. So wird erreicht, dass die Themen sich über die Zeit verändern. Als Folge, verschieben die Themen

möglicherweise ihren Fokus. Dies ist für diese Diplomarbeit ein nicht gewünschter Effekt, da durch die SCY-Missionen die Themen vorgegeben sind und sich diese durch neue Dokumente nicht ändern sollen. Es wird deshalb die in den Grundlagen dargestellte Methode zur Inferenz der Themen neuer Dokumente genutzt, die ohne Veränderung der Themen- und Termverteilungen auskommt.

McCallum et. al. [16] ist eine Arbeit, in der das LDA-Modell zum Topics Over Time (TOT) Modell erweitert wird. Zusätzlich zu den Wörtern wird für jedes Dokument ein Zeitpunkt beobachtet und die Themen werden ferner anhand dieser Zeitpunkte konditioniert. Es werden also Themenverteilungen für Zeitpunkte gelernt. Dies erlaubt es, die Verteilung der Themen über die Zeit zu betrachten. Allerdings muss das TOT-Modell jedes mal aufwändig gelernt werden, wenn neue Dokumente bzw. die Verteilung der Themen über die Zeit für diese neuen Dokumente ermittelt werden soll. Es gibt einen Ansatz, der es erlaubt, ein Themenmodell kontinuierlich weiter zu trainieren [6]. Würde man diesen Ansatz auf das TOT-Modell erweitern, könnte man die neuen Variablen online lernen. Dies könnte man als Erweiterung der Diplomarbeit in Erwägung ziehen. Der Fokus liegt jedoch darauf, aus vorher gelernten Themen, insbesondere im SCY-Kontext, die Verläufe zu bestimmen.

Eine andere Technik, die von Blei und Lafferty entwickelt wurde [3], versucht die Zeit in das LDA-Modell zu integrieren, indem für diskrete Zeitabschnitte jeweils ein Modell trainiert wird, das auf dem vorhergehenden Modell basiert. Dabei werden die Themenverteilungen für Dokumente und die Wortverteilungen für Themen eines Zeitabschnittes t aufgrund der Hyperparameter α und β der vorhergehenden Zeitabschnitte $(1, \ldots, t-1)$ ermittelt (siehe Abschnitt 2.1). Hierbei verändern sich die Wörter, aus denen ein Thema aufgebaut ist. Dies ist im Kontext einer Analyse von Texten, in der sich die Terminologie ändert, das gewünschte Verhalten. In der Diplomarbeit wird jedoch von einem auf Hintergrundwissen trainierten Modell ausgegangen, anhand dessen die im Chat behandelten Themen erkannt werden können.

3.3 Zusammenfassung

Die vorgestellten Arbeiten in diesem Kapitel zeigen Ansätze auf, um aus Chattexten Informationen zu extrahieren und wie das Modell der LDA um die Zeit erweitert werden kann. Die Arbeiten, die sich mit der Extraktion von Informationen aus Chattexten beschäftigen, verwenden jedoch entweder keine Themenmodelle, sondern ordnen die Nachrichten in bestimmte vordefinierte Klassen ein oder betrachten nicht die zeitliche Veränderung.

Die Arbeiten, die das LDA-Modell erweitern, weisen eine komplizierte Implementierung oder eine hohe Laufzeit auf. Die in der Diplomarbeit entwickelte Methode benötigt im Gegensatz zu vorgestellten Arbeiten keine schwierige Anpassung des LDA-Modells. Auch wird in diesen Arbeit oftmals die Zeit mit gelernt, während die Diplomarbeit versucht, die Veränderung der Themen online zu erfassen, ohne die Themenmodelle zu erweitern.

4 Lernphase

In diesem Kapitel wird näher auf die Daten eingegangen, welche benutzt werden. Es wird kurz erläutert, wie die Daten vorverarbeitet werden, welche Daten zur Analyse der zeitlichen Zentralitätsverläufe herangezogen werden und mit welchen Daten man die Themenmodelle trainiert.

4.1 Daten

Aufgrund der Aufgabenstellung wird zwischen Textdaten zum Training des Themenmodells und Textdaten zur Entdeckung von Themen unterschieden. Diese müssen nicht notwendigerweise aus unterschiedlichen Korpora stammen, können es jedoch. So werden für die Experimente mit dpa-Daten diese sowohl zum Training des Themenmodelles als auch zur Entdeckung von Themen über die Zeit benutzt.

Für die zeitliche Betrachtung muss es möglich sein, die einzelnen Dokumente des Korpus zeitlich anzuordnen. Das heißt, jedes Dokument erhält einen Zeitstempel, anhand dessen es zeitlich einsortiert werden kann. Die Funktion $\tau(\mathbf{w})$ liefert den Zeitstempel eines Dokumentes $\mathbf{w}$. Für ein Korpus $\mathcal{W} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_m)$, das für die Entdeckung von zeitlichen Themenveränderungen genutzt wird, muss $\tau(\mathbf{w}_1) \leq \tau(\mathbf{w}_2) \leq \ldots \leq \tau(\mathbf{w}_m)$ gelten. Die Dokumente können also zeitlich sortiert werden. Ein Korpus wird nicht mehr als ungeordnete Sammlung von Dokumenten betrachtet, sondern als Liste von zeitlich sortierten Dokumenten.

Es wurden zwei Textkollektionen zum Training des Themenmodells ausgewählt, die diese Eigenschaften aufweisen; erstens die dpa-Textkollektion und zweitens reale Chatdaten und Hintergrundtexte aus dem SCY-Projekt. Die Texte zum Training der Themenmodelle müssen dabei nicht die Eigenschaft der zeitlichen Anordnung erfüllen.

Die dpa-Texte wurden ausgewählt, um zwei möglichst gegensätzliche Korpora zur Verfügung zu haben. Die dpa-Texte enthalten wesentlich mehr Terme und Dokumente zum Training als auch zur Bestimmung von Verläufen, als die SCY-Texte. Die SCY-Texte enthalten sehr wenige Dokumente zum Training und zur Bestimmung der Verläufe. So wurde der Ansatz dieser Diplomarbeit mit zwei sehr unterschiedlichen Kollektionen getestet; Zum einen mit einem großen Korpus zum Training der Modelle und dem Bestimmen der Themenverläufe und einem sehr kleinen Korpus mit wenig Trainingsdaten und wenig Texten zur Bestimmung der Themenverläufe. Der Fokus der Arbeit liegt jedoch auf Chatdaten und insbesondere Daten mit SCY Hintergrund, so dass die weniger umfangreichen Daten dem angestrebten Einsatzziel näher liegen. In den nächsten Abschnitten werden die beiden Textkollektionen näher vorgestellt.

4.1.1 Vorverarbeitung

Unabhängig davon, ob eine Kollektion zum Training des Themenmodells oder zur Analyse von zeitlichen Themenveränderungen benutzt wird, müssen die Daten vorverarbeitet werden. So werden standardmäßig die Texte der Dokumente in einzelne Wörter unterteilt. Nummern und Satzzeichen werden dabei nicht beachtet und aus den Texten entfernt. Abhängig von der Sprache der Dokumente werden anschließend die bekannten Stopwörter entfernt. Es wurde bei einigen Experimenten auf die Entfernung der Stopwörter verzichtet, um die Validation der Themenmodelle zu testen. Anschließend wurden die Wörter auf ihre Stammform reduziert. Dazu wurde der Snowball-Stemmer [23] verwendet. Dieser benutzt für Englisch den Porter-Algorithmus und für Deutsch einen Lexikon basierten Ansatz. Dies wurde auch nur für einige Experimente durchgeführt, um die Validation zu testen und zu überprüfen, wie sich die Ergebnisse der Themenerkennung bei verschiedenen Varianten der Vorverarbeitung der Texte verhalten.

4.1.2 dpa Nachrichtenmeldungen

Die dpa-Kollektion enthält Nachrichtenmeldungen der Deutschen Presse Agentur (dpa) der Jahre 2000 bis 2006. Sie wurde ausgewählt, weil die Nachrichtenmeldungen sich einfach zeitlich anordnen lassen. Die Meldungen erwähnen bekannte und zeitlich einzuordnende Ereignisse, die zur Überprüfung der entwickelten Methode geeignet sind.

Für jedes in der Kollektion enthaltene Jahr sind die Meldungen pro Monat in einer XML-Datei hinterlegt, die alle Meldungen dieses Monats enthält. Eine einzelne Meldung weist die XML-Struktur in Abbildung 4.1.1 auf. Es sind nur die XML-Tags dargestellt, die zur Extraktion von Informationen genutzt werden. Der Nachrichtentext ist im Tag `<tx>` und `<ta>` zu finden. Der Zeitstempel kann entweder aus dem Tag `<da>` und `<uhr>` extrahiert werden oder aus dem Tag `<dzg>`. Jede Meldung kann somit zeitlich eingeordnet werden, wie es vorausgesetzt wird. Eine Meldung entspricht also einem Dokument $\mathbf{w}_i$ und der Zeitstempel $\tau(\mathbf{w}_i)$ dieses Dokuments entspricht dem Zeitstempel der Meldung.

Zusätzlich wird für jede Meldung festgehalten, in welche IPTC-Kategorie sie einsortiert wurde. Die IPTC-Kategorie ist ein Zahlencode, der die behandelten Themen einer Nachrichtenmeldung angibt. Diese Kategorien stehen im Tag `<iptc>` als durch Leerzeichen getrennte Liste. Die IPTC-Kategorien erlauben es bestimmte Nachrichtenmeldungen für das Training der Themenmodelle auszuwählen und somit die Anzahl der potentiellen Themen zu verringern.

Für das Training der Themenmodelle wurden die Meldungen aus den Monaten November und Dezember des Jahres 2004 ausgewählt. Von diesen wurden nur die Meldungen zum Training benutzt, die in die IPTC-Kategorie Kunst (Zahlencode beginnt mit 01), Unglücke (Zahlencode beginnt mit 03) und Politik (Zahlencode beginnt mit 11) fallen. Die so erstellte Kollektion enthält 13.949 Dokumente mit 114.261 verschiedenen Termen. Mit dieser Kollektion wurden die Themenmodelle trainiert.

Die Textkollektion wurde auf die Monate November und Dezember begrenzt, um erstens die Anzahl der Dokumente zum Training des Themenmodells gering zu halten und zweitens aufgrund des Tsunamis um Weihnachten 2004. So wird das Themenmodell mit großer Wahrscheinlichkeit ein Thema enthalten, welches dieses Ereignis erfasst. Dieses soll dann wiederum in den Testdaten entdeckt werden.

Die drei IPTC-Kategorien wurden gewählt, um die Anzahl der potentiellen Themen indirekt zu begrenzen. Zusätzlich soll die Wahl der IPTC-Kategorien eine möglichst gute Trennung der Themenbereiche gewährleisten.

```
<meldung>
<ob>BDT Basisdienst</ob>
  <da>2004-12-26</da>
  <uhr>13:02:00</uhr>
  ...
  <hl>Zwei starke Beben im Indischen Ozean gemessen</hl>
  <tx>
    Straßburg (dpa) - Zwei Beben der Stärke 5,7 auf der Richterskala
    hat die französische Erdbebenwarte in Straßburg am Sonntag nach dem
    Sumatra-Erdstoß im Indischen Ozean gemessen. Die Epizentren der
    beiden Beben lagen nach den Angaben südlich sowie östlich der
    indischen Andamanen-Inseln, teilte die Warte mit. Die Erdstöße
    ereigneten sich um 5.21 und 10.19 MEZ. Straßburg hatte bei dem
    folgenschweren Beben auf Sumatra eine Stärke von 8,1 registriert.
  </tx>
  <ta>
  </ta>
  <dzg>261302 Dez 04</dzg>
  ...
  <iptc>08000000</iptc>
</meldung>
```

Abbildung 4.1.1: DPA-Meldung im XML-Format

Anhand dieser Kollektion wurden Themenmodelle trainiert, die jeweils 10, 20 bis 200 Themen enthalten. Es wurde mehrere Themenmodelle mit verschiedener Größe und verschiedenen Parametern gelernt, um die Validationsmethoden der Themenmodelle zu überprüfen und aus diesen ein passendes Themenmodell für die Bestimmung der Themenverläufe auszuwählen. Die Kriterien sind dabei eine möglichst gute Trennung der Themen anhand der Validationswerte bzw. die durch manuelle Inspektion festgestellte Adäquatheit der Themenmodelle in Bezug auf das Trainingskorpus.

Dieselben Texte, die zum Training des Themenmodells verwendet wurden, können als zeitliche Verlaufsdaten dienen. Tatsächlich wird jedoch nur eine Teilmenge der Trainingsdaten benutzt. Es wurden nur Meldungen vom 25.12.2004 bis 30.12.2004 benutzt, die auch in die drei oben genannten IPTC-Kategorien fallen. So kamen insgesamt 1.659 Dokumente zustande die einen Zeitraum von vier Tagen umfassen. Dieser spezielle Zeitraum wurde ausgewählt, da am 26.12.2004 ein

Erdbeben in Südostasien einen Tsunami ausgelöst hat, der große Teile der Küste verwüstet hat. Es wird erwartet, dass dieses Ereignis als prominentes Thema in den Verläufen entdeckt wird.

Zusätzlich zu den realen Daten aus dem dpa-Korpus wurden synthetische Verläufe aus den Themenmodellen konstruiert. Das generative Modell der LDA wurde dazu benutzt, Dokumente zu generieren, die bestimmte Themen mit fester Wahrscheinlichkeit enthalten. Diese werden als Validationsverläufe benutzt. Näheres zur Konstruktion und Evaluation dieser Verläufe findet sich in Abschnitt 5.5.

4.1.3 SCY Chat-Daten

Im Gegensatz zu den dpa-Texten muss bei den SCY-Daten zwischen Trainings und Analysetexten unterschieden werden. Das Training der Themenmodelle wird mit denselben Texten durchgeführt, die den Schülern zur Verfügung gestellt werden, um sich über die CO_2-Thematik zu informieren. Diese Hintergrundtexte sind Webseiten im Internet, deren Text zum Training der Modelle manuell extrahiert und als unstrukturierter Text gespeichert wurde.

Insgesamt wurden 56 Dokumente extrahiert und zum Training der Themenmodelle genutzt. Diese 56 Dokumente enthalten 5.605 unterschiedliche Terme. Im Gegensatz zu den dpa-Daten sind die Hintergrundtexte und die Chattexte Englisch. So kann zugleich die Eignung der hier entwickelten Methode für verschiedene Sprachen überprüft werden.

Aus den Hintergrundtexten wurden Themenmodelle mit jeweils 5, 10, 15 bis 100 Themen erzeugt. Diese Themenmodelle werden anhand der in Abschnitt 4.2 dargestellten Metrik überprüft, welches die vorkommenden Themen am besten wiedergibt. Aufgrund der wenigen verfügbaren Dokumente ist die Laufzeit des Trainingsalgorithmus kurz. Dies erlaubt es, mehr Themenmodelle zu trainieren, um mehr Daten zur Validation der Themenmodelle zur Verfügung zu haben.

```
...
Hilde: Should we just write down the numbers, then we have them   What is 24 cm
Per: Rise in sea level
Hilde: It will be increase in both temperature, sea level, emisions   or
Per: All the three factors   curves   temperature increases by 3,44 degrees
Hilde: ...and the sea level increases by 24 centimetre by the year 2100.
Tom: I will say that this is a sustainable world  Maybe a little unstable in the economy, and the
death rate
Mette: Oh, look at this, how much it  increases
Tom: Yes, it completely crazy. Uh   ohhh, how did we do that!!
Mette: CO2 went alt the way down, that because we took these down
Tom: OK
Mette: But why, did it increase so much there? OK, we should take it more down, it looks better
Tom: A little maybe, Imagine how cheap it   will be. OK, now everything changed. Don't touch it,
more now, it became such a nice curve  Hey, come on, drag the last one down, a little bit down
...
```

Abbildung 4.1.2: SCY-Chatdaten

Für die Analyse von Chattexten wurden reale Chattexte benutzt. Diese wurden von einem SCY-Projektpartner zur Verfügung gestellt. Es handelt sich dabei um Dialogprotokolle von Schülern, die eine Aufgabe zum Thema CO_2und dem Bau eines CO_2-neutralen Hauses bearbeiteten. Die Chats liegen in einer Textdatei vor, wobei jede Zeile dieser Datei einer Nachricht eines Schülers entspricht (siehe Abbildung 4.1.2). Jede Nachricht wird als Dokument aufgefasst und dem System zur Verfügung gestellt. Insgesamt wurden 72 Nachrichten mitprotokolliert, die dann dazu benutzt wurden die Themenverläufe zu ermitteln. Es sind leider nicht mehr Texte verfügbar, da das SCY-Projekt noch nicht so weit fortgeschritten ist, dass es produktiv eingesetzt werden kann. So sind leider nur die wenigen Dokumente aus der Konzeptionsphase der ersten SCY-Mission verfügbar.

Der Zeitstempel $\tau(\mathbf{w}_i)$ einer Nachricht $\mathbf{w}_i$ entspricht der Position in der Datei. Die Zeitstempel sind hier nicht das Datum der Nachricht, sondern die Nachrichten werden fortlaufend durchnummeriert Die erste Nachricht, die ein Schüler schreibt erhält den Zeitstempel eins, die darauf folgende Antwort den Zeitstempel zwei bis alle Nachrichten nummeriert sind. Die Chatnachrichten werden somit nach dem Zeitpunkt des Auftretens sortiert.

Die Verwendung einer Nummerierung zur Ordnung der Chatnachrichten ist mit dem Fehlen einer Zeitinformation zu jeder Nachricht begründet. In der Chatsoftware die in der finalen SCY-Applikation verwendet wird, wird jedoch der Zeitpunk zu dem eine Nachricht geschrieben wurde, mitprotokolliert. So kann analog zur dpa-Kollektion das Datum der Nachricht als Zeitstempel benutzt werden. Des Weiteren werden neben den Chattexten, entsprechend der dpa-Kollektion, aus den gelernten Themenmodellen synthetische Themenverläufe erzeugt, die der Evaluation der Zentralitätsmaße dienen.

4.2 Validierung der Modelle

Um die Themenmodelle bewerte zu können, muss man den Aufbau der Themen examinieren. Ein Thema ist durch die Verteilung der Terme charakterisiert. Die Termverteilung zu einem Thema gibt an, welche Terme mit welcher Wahrscheinlichkeit diesem zugeordnet werden. Die Eignung der Themenmodelle wird direkt anhand der Termverteilung eines Themas bestimmt. Da die Themen möglichst gut separieren sollen ist es naheliegend, die Eignung der Themenmodelle anhand dieser Werte zu bestimmen. Betrachtet man die Termverteilung zweier Themen, so sind die Themen dann möglichst unterschiedlich, wenn die Termverteilungen sich unterscheiden. Termverteilungen, die dieselben Terme mit ähnlicher Wahrscheinlichkeit enthalten, werden als ähnlich betrachtet. Themen deren Form der Termverteilung unterschiedlich aussieht, sind auch unterschiedlich.

Um zu berechnen, wie gut sich die Themen unterscheiden, wird für jedes Thema $\mathbf{z}_i$ die Kullback-Leibler-Divergenz anhand der Termverteilung φ_i zu allen anderen Themen $\mathbf{z}_j$ be-

stimmt. Daraus entsteht eine Matrix A, die die Kullback-Leibler-Divergenz jedes Themas zu allen anderen enthält. Die Abstandsmatrix ist definiert als

$$A_{ij} = D_{KL}(\varphi_i \| \varphi_j) := \sum_{k=0}^{V} \varphi_{i,k} \log \frac{\varphi_{i,k}}{\varphi_{j,k}}.$$

Anhand der wechselseitigen Divergenzwerte wird der mittlere Divergenzwert d_m aller Themen eines Modells bestimmt. Es wird

$$d_m = \frac{\sum_{i=0}^{K} \sum_{j=0}^{K} A_{ij}}{K^2}$$

berechnet.

Der Divergenzwert alleine ist nicht aussagekräftig. Er muss in Beziehung zu den Divergenzwerten anderer Themenmodelle gesetzt werden. Vergleicht man die mittleren Divergenzwerte von Themenmodellen, die mit verschiedener Anzahl von Themen trainiert wurden, kann man anhand der Divergenzwerte feststellen mit welcher Anzahl von Themen die Modelle am besten separieren. Trägt man die Divergenzwerte gegen die Anzahl der Themen auf, kann man einen Bereich ablesen für den die Themen gut separiert sind. Aus diesem Bereich soll dann ein Themenmodell zur Bestimmung der Themenverläufe ausgesucht werden.

Zur Validation von Themenmodellen werden normalerweise Testdokumente zurückgehalten, für die dann bestimmt wird wie gut das Modell die Themen in diesem Dokument ermittelt. Dies ist jedoch für die Anwendung in der Diplomarbeit nicht ausreichend, da wenig Dokumente zum Training zur Verfügung stehen, so dass alle zur Verfügung stehenden Dokumente zum Training der Modelle genutzt werden. Deshalb wurde eine eigene Methode zur Validierung der Themenmodelle entwickelt, die ohne Zurückhaltung von Dokumenten auskommt.

4.3 Zusammenfassung

Dieses Kapitel stellte die verwendeten Text vor, wie diese in Trainingsdaten und Analysedaten aufgeteilt werden und erläuterte ihren Aufbau. Anschließend wurde die Methode vorgestellt, anhand derer geeignete Themenmodelle ausgewählt werden können. Welche verschiedenen Modelle gelernt wurden und wie sie sich eignen, wird in Kapitel 6 dargelegt.

5 Anwendungsphase

Im Folgenden wird der Prozess vorgestellt, mit dem aus Textsequenzen und einem vorher trainierten Themenmodell die Verläufe der Themen ermittelt werden können. Dazu wird zuerst dargelegt, wie für die neuen Dokumente die Themen ermittelt werden. Dann wird aus den ermittelten Themen für die Dokumente eines Zeitabschnittes ein Graph aufgebaut, dessen Knoten dann mit einem Zentralitätsindex bewertet werden. Anschließend wird veranschaulicht, wie die Verläufe visualisiert werden. Schlussendlich wird vorgestellt, wie die Methode auf ihre Aussage und Richtigkeit überprüft wird.

5.1 Frames

Zuerst wird die Textsequenz in gleich große diskrete Zeitabschnitte unterteilt, Frames genannt. Es ist dabei nicht nötig, dass die Frames disjunkt sind. Es muss jedoch gelten, dass sie eine Sequenz bilden bzw. eindeutig anzuordnen sind. Für ein Korpus $\mathcal{W} = (\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_F)$ gilt:

1. die einzelnen Frames weisen eine Reihenfolge auf. D.h $\mathcal{F}_1 <_{\mathcal{F}} \mathcal{F}_2 <_{\mathcal{F}} \cdots <_{\mathcal{F}} \mathcal{F}_F$ wobei $<_{\mathcal{F}}$ die Ordnung auf den Frames darstellt.

2. die Frames stellen eine komplette Zerlegung des Korpus dar. D.h. $\mathcal{W} = \bigcup_{i=0}^{F} \mathcal{F}_i$.

3. die Dokumente innerhalb eines Frames müssen wiederum, wie in der Gesamtsequenz, sortiert sein. Für einen Frame $\mathcal{F} = (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_f)$ gilt auch, dass $\tau(\mathbf{w}_1) \leq \tau(\mathbf{w}_2) \leq \cdots \leq \tau(\mathbf{w}_f)$.

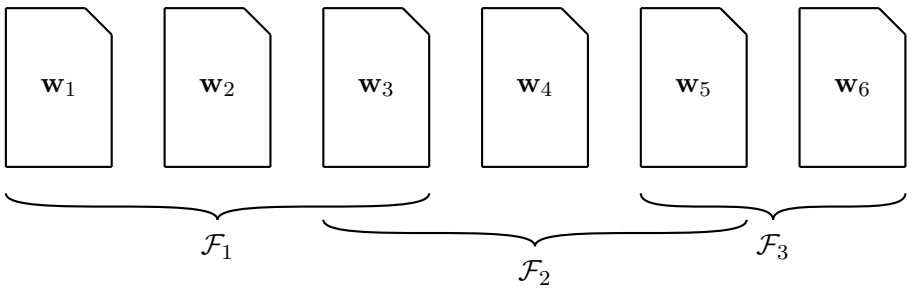


Abbildung 5.1.1: Zerlegung eines Korpus in Frames.

In Abbildung 5.1.1 ist eine Zerlegung von sechs Dokumenten in Frames der Größe drei dargestellt. Die Frames werden jeweils um zwei Dokumente versetzt erzeugt. Frame $\mathcal{F}_1$ und Frame $\mathcal{F}_2$ enthalten somit beide Dokument $\mathbf{w}_3$. Der letzte Frame $\mathcal{F}_3$ enthält nur zwei Dokumente, da durch die Unterteilung für den letzten Frame nicht genug Dokumente vorhanden sind, um ihn komplett aufzufüllen. Die Sortierung der Dokumente bleibt innerhalb der Frames erhalten, somit gilt Bedingung drei. Bedingung eins und zwei werden durch die Konstruktion der Frames erfüllt. Die Frames enthalten nicht weniger als alle Dokumente der Textsequenz und die Reihenfolge ist durch die Position der Frames gegeben.

Hier könnte die Ordnung $<_{\mathcal{F}}$ folgendermaßen definiert sein. Sei $\mathbf{w}_{\mathcal{F}_i,j}$ das j-te Dokument in Frame $\mathcal{F}_i = (\mathbf{w}_1, \ldots, \mathbf{w}_j, \ldots, \mathbf{w}_f)$ dann ist

$$\mathcal{F}_i <_{\mathcal{F}} \mathcal{F}_j \Leftrightarrow \tau(\mathbf{w}_{\mathcal{F}_i,1}) \leq \tau(\mathbf{w}_{\mathcal{F}_j,1}).$$

Abhängig von der Textsequenz steuert die Wahl der Framegröße den betrachteten Zeitraum. So ist es denkbar, dass für das dpa-Korpus die Framegröße so gewählt wird, das eine Woche oder ein Tag in einem Frame zusammengefasst wird. Bei Chattexten ist es besser, die letzten Minuten zu betrachten, da die Konversation schneller ist und sich die Themen dementsprechend schnell ändern können. Dementsprechend wählt man die Framegröße.

5.2 Themengraphen

Der nächste Schritt im Ablauf ist die Ermittelung der Themen für die Dokumente und die Erstellung des relationalen Graphen. Insgesamt wurden im Rahmen dieser Diplomarbeit drei Algorithmen entwickelt um aus Themenkookkurrenzen einen Graphen zu erstellen. Diese wurden entwickelt, da bisher keine Algorithmen bekannt sind, die aus Themenkookkurrenzen Graphen erzeugen. Shaffer et. al. [22] haben einen ähnlichen Algorithmus entwickelt, der Kookkurrenzen von Klassen eines Klassifikationsalgorithmus in einem Graphen kodiert. Die hier entwickelten Algorithmen sind von diesem Algorithmus inspiriert, kodieren jedoch zusätzliche Informationen in den Graphen. So fließen zum Beispiel zusätzlich die Wahrscheinlichkeiten der Themen in die Algorithmen ein.

Sei $\mathcal{F} = (\mathbf{w}_1, \ldots, \mathbf{w}_f)$ ein Frame des Korpus. Dann wird anhand des gelernten Themenmodells für jedes Dokument inferiert, welche Themen in diesem vorkommen. Dies geschieht anhand der Inferenz für ungesehene Dokumente in Abschnitt 2.1.3 des Grundlagenkapitels. So erhält man für jedes Dokument $\mathbf{w}_i$ in Frame $\mathcal{F}$ eine Themenverteilung ϑ_i. Anhand der Themenverteilung der Dokumente in den Frames kann ein Graph aufgebaut werden, der die Kookkurrenz und die Wahrscheinlichkeit der Themen in einem Frame kodiert. Die Knoten des Graphen stellen dabei die Themen dar und die Kanten kodieren die Kookkurrenz der Themen in einem Frame.

5.2.1 Vollständiger Graph (VVG)

Der erste Algorithmus erstellt einen vollständig verbundenen Graphen, dessen Kanten mit der inversen Häufigkeit des Themenauftretens gewichtet sind. Die Themen, deren Wahrscheinlichkeiten größer als ein Schwellwert ϵ sind, werden als Knotenmenge aufgefasst und es wird für jedes Themenpaar gezählt, wie oft es zusammen auftritt. In dem Pseudocode Algorithmus 5.2.1 wird der Algorithmus dargestellt.

Es wird über alle Dokumente $\mathbf{w}_m \in \mathcal{F}$ iteriert und die Themen $\mathbf{z}_k$, deren Wahrscheinlichkeit $\vartheta_{m,k}$ im Dokument größer als der gewünschte Schwellwert ist, werden als Knoten in den Graphen eingefügt. Dann werden für alle diese Themen $\mathbf{z}_k$ Kanten zu allen anderen Knoten v gezogen. Wenn die Kante im Graphen schon vorhanden ist, muss ihr Gewicht angepasst werden. Für jede bereits vorhandene Kante wird das Gewicht von $\frac{1}{n}$ auf $\frac{1}{n+1}$ angepasst. Dazu wird in Zeile 6 erst einmal das aktuelle Gewicht der Kante ausgelesen. Anhand des alten Gewichtes kann man feststellen, wie oft die Kante schon in den Graphen eingefügt wurde. Daraus berechnet sich dann das neue Gewicht. Der Vorteil besteht darin, dass man sich eine Datenstruktur erspart, die zählt, wie oft eine Kante bereits eingefügt wurde und, dass eine Iteration über alle Kanten wegfällt. Wenn die Kante im Graphen noch nicht vorhanden ist, wird sie hinzugefügt und bekommt ein initiales Gewicht von eins.

Algorithmus 5.2.1 Vollständig verbundener Graph (VVG)

```
1:  V = ∅, E = ∅
2:  for all w_m ∈ F do
3:     V = V ∪ {k|ϑ_{m,k} > ε}
4:     for k ∈ {ϑ_{m,k} > ε} do
5:        for v ∈ V do
6:           if {v, k} ∈ E ∧ v ≠ k then
7:              oldWeight = ω({v, k})
8:              count = 1/oldWeight
9:              ω({v, k}) = 1/(count+1)
10:          else
11:             E = E ∪ {{v, k}}
12:             ω({v, k}) = 1
13:          end if
14:       end for
15:    end for
16: end for
```

Alternativ kann der Graph in mathematischer Mengenschreibweise angegeben werden, wenn man festhält, wie oft ein Thema als Knoten eingefügt wurde. Dazu sei $G = (V, E)$ ein Graph mit V als Knotenmenge und E als Kantenmenge. Die Knotenmenge V ist hier ein Multiset, das festhält, wie oft ein Element in die Menge eingefügt wurde. Die Anzahl der einzelnen Mengenelemente x wird durch die Funktion $count(x)$ angegeben.

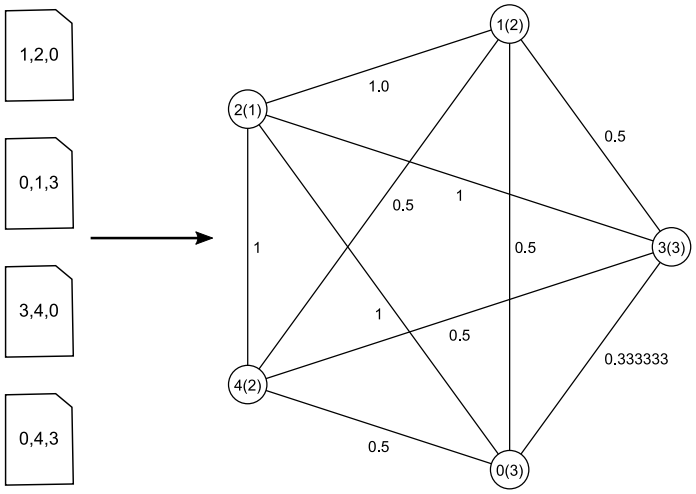


Abbildung 5.2.1: Erzeugung eines Graphen aus Dokumenten und den zugewiesenen Themen mittels des Algorithmus VVG

Die Knoten des Graphen ergeben sich aus den Themen, für die die Dokumentwahrscheinlichkeit größer als ein Schwellwert ϵ ist.

$$V = \bigcup_{\mathbf{w}_m \in \mathcal{F}} \{k | \vartheta_{m,k} > \epsilon\} \tag{5.2.1}$$

Zwischen allen Knoten im Graphen werden dann Kanten gezogen. Dabei sind Zyklen im Graphen nicht erlaubt. Es ist:

$$E = \{\{i, j\} | i, j \in V, i \neq j\} \tag{5.2.2}$$

Das Kantengewicht wird als die inverse Anzahl der Kookkurrenz von Themen definiert. Die Anzahl der Kookkurrenzen ist dabei das Minimum der Auftreten der Knoten einer Kante.

$$\omega(\{i, j\}) = \frac{1}{\min\{count(i), count(j)\}} \tag{5.2.3}$$

In Abbildung 5.2.1 ist der resultierende Graph eines Frames mit vier Dokumenten abgebildet. Die Zahlen in den Knoten geben den Index des Themas und die Zahlen in Klammern die Anzahl des Auftretens des Themas an.

5.2.2 Dokumentzentrierter Graph (DZG)

Der zweite Algorithmus erzeugt keinen vollständig verbundenen Graphen. Im Einzelnen werden, ähnlich wie im vorhergehenden Algorithmus VVG, die Themen der Dokumente mit einer Wahrscheinlichkeit größer als ein angegebener Schwellwert als Knoten in den Graphen eingefügt. Allerdings werden nur noch Kanten gezogen, wenn die Themen in den Dokumenten vorkommen. Da gleiche Themen oft in verschiedenen Dokumenten vorkommen, ergibt sich wieder ein zumin-

dest schwach zusammenhängender Graph. Ein Thema, das in allen Dokumenten vorkommt, wird mit allen anderen Themen verbunden sein. Dies kann man in Abbildung 5.2.2 sehen. Hier kommt Thema 1 in jedem Dokument vor und verbindet deshalb alle Dokumentsubgraphen miteinander. Thema 6 und 4 kommen in zwei Dokumenten vor und verbinden somit zwei Dokumente. Hier ist es jedoch möglich, dass ein Dokument nur aus Themen zusammengesetzt ist, die in keinem anderen Dokument vorkommen. So entsteht ein nicht verbundener Graph.

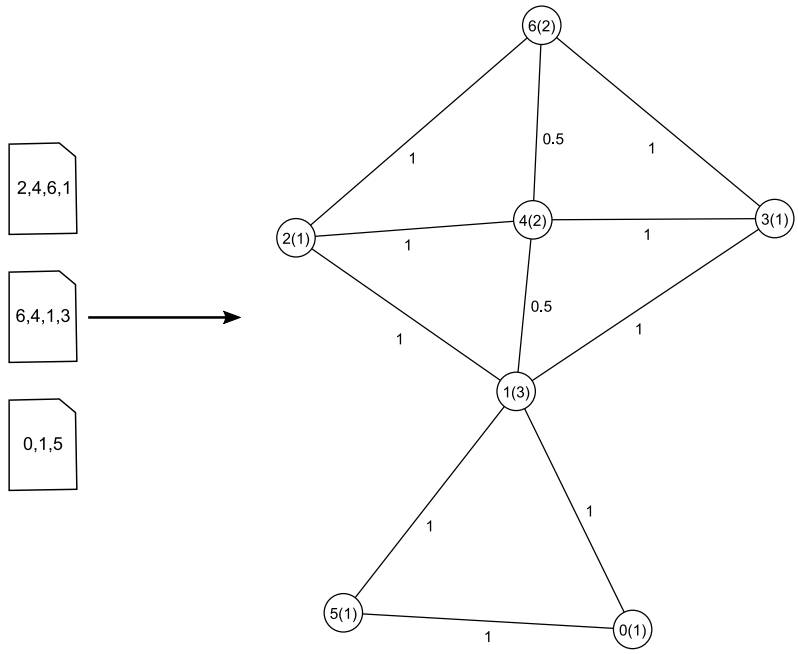


Abbildung 5.2.2: Erzeugung eines Graphen aus Dokumenten und den zugewiesenen Themen mittels des Algorithmus DZG

Der Algorithmus zur Erstellung der Graphen funktioniert ähnlich zum vorhergehenden. Es wird für jedes Dokument ein vollständig verbundener Subgraph erstellt. Dies wird in Algorithmus 5.2.2, in den Zeilen 3 bis 16 codiert. Dort werden auch die Themen $\mathbf{z}_k$ als Knoten eingefügt, deren Wahrscheinlichkeit $\vartheta_{m,k}$ größer als der Schwellwert ist. Dann wird für jeden Knoten im Subgraphen $G_m = (V_m, E_m)$ eine Kante zu jedem anderen Knoten gezogen. Wenn eine Kante im globalen Graphen $G = (V, E)$ schon vorhanden ist, muss das Gewicht dieser Kante angepasst werden. Dazu wird analog wie in Algorithmus 5.2.1 erst die Anzahl der Vorkommen dieser Kante bestimmt und dann das Gewicht angepasst (siehe Zeile 8-10). Wenn der Subgraph G_m erstellt wurde, werden die Knotenmengen und Kantenmengen des Subgraphen mit dem globalen Graphen vereinigt.

Hier ist es gleichfalls möglich, den Algorithmus direkt in Mengenschreibweise anzugeben. Es muss wieder festgehalten werden, wie oft ein Thema in die Knotenmenge eingefügt wurde. Die

Algorithmus 5.2.2 Dokument zentrischer Graphenalgorithmus (DZG)

```
1: V = ∅, E = ∅
2: for all w_m ∈ F do
3:    V_m = ∅, E_m = ∅
4:    V_m = {k|ϑ_{m,k} > ϵ}
5:    for k ∈ V_m do
6:       for v ∈ V_m do
7:          if {v, k} ∈ E ∧ v ≠ k then
8:             oldWeight = ω({v, k})
9:             count = 1/oldWeight
10:            ω({v, k}) = 1/(count+1)
11:         else
12:            E_m = E_m ∪ {{v, k}}
13:            ω({v, k}) = 1
14:         end if
15:      end for
16:   end for
17:   V = V ∪ V_m
18:   E = E ∪ E_m
19: end for
```

Knotenmenge ist die gleiche wie im vorhergehenden Algorithmus: Sie enthält alle Themen, deren Dokument-Wahrscheinlichkeit größer als ϵ ist.

$$V = \bigcup_{\mathbf{w}_m \in \mathcal{F}} \{k|\vartheta_{m,k} > \epsilon\}$$

Allerdings wird nicht mehr jeder Knoten mit allen anderen Knoten verbunden. Es werden nur noch die Knoten verbunden, für die gilt, dass sie im selben Dokument vorkommen und ihre Dokumentwahrscheinlichkeit größer als der geforderte Schwellwert ist.

$$E = \bigcup_{\mathbf{w}_m \in \mathcal{F}} \{i, j|\vartheta_{m,i} > \epsilon \wedge \vartheta_{m,j} > \epsilon, i \neq j\}$$

Die Kantengewichte wiederum werden analog zum vorhergehenden Algorithmus bestimmt.

$$\omega(\{i, j\}) = \frac{1}{min\{count(i), count(j)\}}$$

5.2.3 Graph mit Themenwahrscheinlichkeit (GMT)

Der nächste Algorithmus berücksichtigt die Wahrscheinlichkeit der Themen in den Dokumenten. Anhand der Wahrscheinlichkeiten werden die Gewichte der Kanten berechnet. Ansonsten wird der Graph wie in Abschnitt 5.2.2 aufgebaut. Zuerst werden alle Themen des aktuellen Dokuments, deren Wahrscheinlichkeit größer als ϵ sind, als Knoten in den Subgraphen eingefügt.

Danach werden zwischen allen Knoten des Subgraphen Kanten gezogen. Wenn die Kante im globalen Graphen noch nicht existiert, wird sie im Subgraphen hinzugefügt mit einem initialen Gewicht das dem inversen F-Wert der beiden Themenwahrscheinlichkeiten k und v im Dokument m entspricht. Ist die Kante im globalen Graphen schon vorhanden, wird das Gewicht folgendermaßen angepasst: Das neue Gewicht berechnet sich als Produkt des alten Gewichtes mit dem inversen F-Wert der Themenwahrscheinlichkeiten der beiden Themen k und v im Dokument m.

$$1 - \frac{2 \cdot \vartheta_{m,v} \cdot \vartheta_{m,k}}{\vartheta_{m,v} + \vartheta_{m,k}} \tag{5.2.4}$$

Der inverse F-Wert in Gleichung 5.2.4 nähert sich für zwei hohe Wahrscheinlichkeiten $\vartheta_{m,v}$ und $\vartheta_{m,k}$ Null an und für zwei niedrige Wahrscheinlichkeiten Eins an. Das heißt, zwei Knoten mit hoher Wahrscheinlichkeit liegen näher beieinander als zwei Knoten mit niedriger Wahrscheinlichkeit. Durch die Anpassung der Gewichte wird die Distanz zweier Themenknoten, die häufiger zusammen auftreten, kleiner. Dies ist für die später berechneten Zentralitätsindizes wichtig.

Algorithmus 5.2.3 Graph mit Themenwahrscheinlichkeit (GMT)

1: $V = \emptyset, E = \emptyset$
2: **for all** $\mathbf{w}_m \in \mathcal{F}$ **do**
3: $V_m = \emptyset, E_m = \emptyset$
4: $V_m = \{k | \vartheta_{m,k} > \epsilon\}$
5: **for** $k \in V_m$ **do**
6: **for** $v \in V_m$ **do**
7: **if** $\{v, k\} \in E \wedge v \neq k$ **then**
8: $\omega(\{v, k\}) = \omega(\{v, k\}) \cdot \left(1 - \frac{2 \cdot \vartheta_{m,v} \cdot \vartheta_{m,k}}{\vartheta_{m,v} + \vartheta_{m,k}}\right)$
9: **else**
10: $E_m = E_m \cup \{\{v, k\}\}$
11: $\omega(\{v, k\}) = 1 - \frac{2 \cdot \vartheta_{m,v} \cdot \vartheta_{m,k}}{\vartheta_{m,v} + \vartheta_{m,k}}$
12: **end if**
13: **end for**
14: **end for**
15: $V = V \cup V_m$
16: $E = E \cup E_m$
17: **end for**

Die Definition des Graphen in Mengenschreibweise ist wie folgt. Die Knoten des Graphen sind wieder diejenigen Themen deren Wahrscheinlichkeit in einem Dokument des Frames größer als ϵ ist.

$$V = \bigcup_{\mathbf{w} \in \mathcal{F}} \{k | \vartheta_{m,k} > \epsilon\}$$

Die Kanten werden zwischen Themenknoten gezogen, die in demselben Dokument vorkommen, analog zum vorhergehenden Algorithmus.

$$E = \bigcup_{\mathbf{w}_m \in \mathcal{F}} \{i, j | \vartheta_{m,i} > \epsilon \wedge \vartheta_{m,i} > \epsilon, i \neq j\}$$

Das Gewicht einer Kante von i nach j berechnet sich als das Produkt des inversen F-Wertes der Wahrscheinlichkeit des Themas i und j in allen Dokumenten, in denen die Themen i und j zusammen auftreten. Dies entspricht dem mehrmaligen Einfügen einer Kante, wie es im Algorithmus aufgeführt wird.

$$\omega(\{i, j\}) = \prod_{m:\vartheta_{m,i}>\epsilon \wedge \vartheta_{m,j}>\epsilon} 1 - \frac{2 \cdot \vartheta_{m,i} \cdot \vartheta_{m,j}}{\vartheta_{m,i} + \vartheta_{m,j}}$$

5.3 Zentralitäten

Auf die so erstellten Graphen werden nun die in Kapitel 2 vorgestellten Zentralitätsindizes angewendet. Für jeden Zeitabschnitt gibt es einen Graphen, dessen Knoten die vorkommenden Themen repräsentieren und der die Wahrscheinlichkeit der Themen in der Struktur des Graphen enthält. Im Falle der beiden Algorithmen VVG und DZG ist die Wahrscheinlichkeit implizit enthalten, im Falle des Algorithmus GMT explizit. Dennoch kann mit Hilfe der Zentralitätsindizes die Wichtigkeit der Themen bestimmt werden.

5.3.1 Normalisierung

Zur besseren Vergleichbarkeit der Ergebnisse der unterschiedlichen Zentralitäten werden die Verläufe normalisiert. Die Normalisierung der Zentralitätsindizes kann einerseits pro Graph vorgenommen werden oder über alle betrachteten Frames.

Bei der Normalisierung pro Graph werden die Zentralitätsindizes der Knoten auf den Wertebereich $[0, 1]$ abgebildet. Für jeden Graphen werden die Zentralitätsindizes anhand der L^∞-Norm normalisiert, indem die einzelnen Zentralitätswerte der Knoten durch den maximalen Zentralitätswert im Graphen geteilt werden. Die normalisierte Zentralität c_{nX} für einen Knoten v im Graphen $G = (V, E)$ mit Zentralität $c_X(v)$ ist

$$c_{nX}(v) = \frac{c_X(v)}{\max_{u \in V} \{c_X(u)\}}.$$

So erhält man für jeden Graphen Zentralitätswerte zwischen Null und Eins. Leider lassen sich damit verschiedene Graphen nicht ohne weiteres vergleichen [5], was direkte Auswirkungen auf die Vergleichbarkeit der prominenten Themen in den aufeinanderfolgenden Frames hat.

Es wird deshalb eine Normalisierung vorgeschlagen, die nicht die einzelnen Graphen normalisiert, sondern über alle Frames normalisiert. Hierzu wird auch die L^∞-Norm verwendet jedoch,

wird nicht mehr der Maximalwert eines einzelnen Graphen benutzt, sondern der Maximalwert aller Graphen bzw. Frames.

Sei $G_{\mathcal{F}_i} = (V_{\mathcal{F}_i}, E_{\mathcal{F}_i})$ der korrespondierende Graph zu Frame $\mathcal{F}_i$. Dann werden die Zentralitätswerte folgendermaßen normalisiert:

$$c_{nX}(v) = \frac{c_X(v)}{\max_{u \in V_{\mathcal{F}_i} \ \forall \mathcal{F}_i \in \mathcal{W}} \{c_X(u)\}}.$$

So wird die Vergleichbarkeit der unterschiedlichen Graphen und der verschiedenen Zentralitätsindizes gewährleistet, da jeder Knoten relativ zu allen Graphen bewertet wurde. In der späteren Online-Applikation ist eine Normalisierung nicht mehr möglich, da man im Voraus das Maximum nicht bestimmen kann. Da die Normalisierung aber nur zur Vergleichbarkeit der Zentralitätsindizes benutzt wird, sollten die absoluten Werte ausreichend sein.

5.3.2 Nicht verbundene Graphen

Bei den letzten beiden, in Abschnitt 5.2 beschriebenen Algorithmen kann es vorkommen, dass diese einen Graph erzeugen, der nicht verbunden ist. Wie schon in den Grundlagen dargelegt wurde, sind einige Zentralitätsindizes nicht auf unzusammenhängenden Graphen definiert.

Ein Ansatz dieses Problem zu handhaben, besteht darin die Zentralitätsindizes für jede Zusammenhangskomponente einzeln zu berechnen und die Werte dann mit der Größe der Komponente zu gewichten. Dies funktioniert, solange der Zentralitätsindex sich proportional zur Größe des Graphen verhält [5]. Die Closeness- und Eccentricity-Zentralität verhalten sich jedoch nicht proportional zur Größe des Graphen [20]. Somit werden die Ergebnisse dieser Zentralitäten für unzusammenhängende Graphen unbrauchbar. Gleichzeitig haben Poulin, Boily und Mâsse [20] einen neuen Zentralitätsindex entwickelt, der auch für unzusammenhängende Graphen funktioniert. Da der Fall, dass ein nicht zusammenhängender Graph erzeugt wird, jedoch selten auftritt und durch geeignete Wahl der Framegröße und der Frameüberlappung umgangen werden kann, wurde dieser Zentralitätsindex hier nicht implementiert.

Es wird jedoch folgende Heuristik verwendet, um unzusammenhängende Graphen bewerten zu können. Es wird generell die Zusammenhangskomponente mit der größten Anzahl an Knoten bewertet. Zusammenhangskomponenten, die um bis zu 20% von der Maximalgröße abweichen, werden ebenfalls bewertet. In allen anderen Zusammenhangskomponenten werden die Zentralitätswerte der Knoten auf Null gesetzt. So entdeckt man einen Themenbruch innerhalb eines Frames, wenn die Themen in ungefähr gleich vielen Dokumenten erwähnt sind.

5.4 Visualisierung

Für die Visualisierung der Themenverläufe wurde eine eigene Komponente entwickelt, die die Verläufe darstellt. Es wurden zwei Arten der Visualisierung implementiert. In der ersten Variante

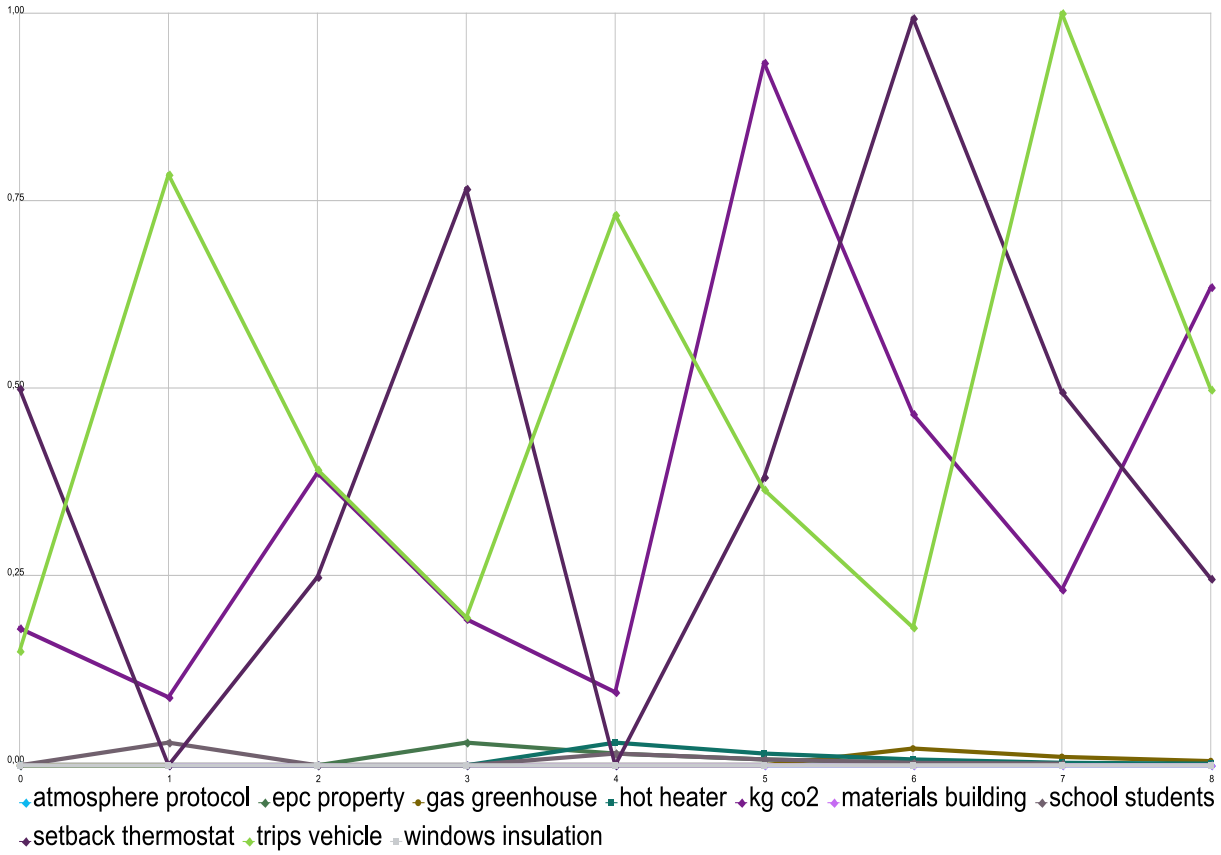


Abbildung 5.4.1: Alle Themenverläufe in einem Diagramm. Auf der X-Achse ist die Framezahl aufgetragen, auf der Y-Achse die normalisierte Zentralität

werden die Themenverläufe als Liniendiagramm in dasselbe Koordinatensystem eingezeichnet. Auf der X-Achse wird die laufende Framenummer abgetragen und auf der Y-Achse die normalisierte Zentralität. Jedes Thema wird durch eine eigene Linie repräsentiert. Welche Linie welches Thema repräsentiert wird als Beschriftung unterhalb des Diagramms eingeblendet (siehe Abbildung 5.4.1).

Bei dieser Art der Visualisierung kann das Problem auftreten, dass die Prominenz der Themen nicht mehr differenziert werden kann, wenn viele Themen einen ähnlichen Verlauf haben, bzw. einen Verlauf auftritt, der viele andere Verläufe schneidet. Andererseits können bei Verläufen, in denen nicht viele Themen eine hohe Prominenz aufweisen, mit einem Blick die wichtigen Themen abgelesen werden und das Verhältnis zu den anderen Themen bestimmt werden.

In der zweiten Variante zeichnet man die Themenverläufe jeweils in ein eigenes Diagramm ein (siehe Abbildung 5.4.2). Diese Diagramme werden dann in einer Matrix angeordnet. Diese Art der Visualisierung hat den Vorteil, dass man besser differenzieren kann, welche Themen wichtig sind. Jedoch ist die relative Zentralität eines Themas im Verhältnis zu anderen Themen schwieriger zu erkennen.

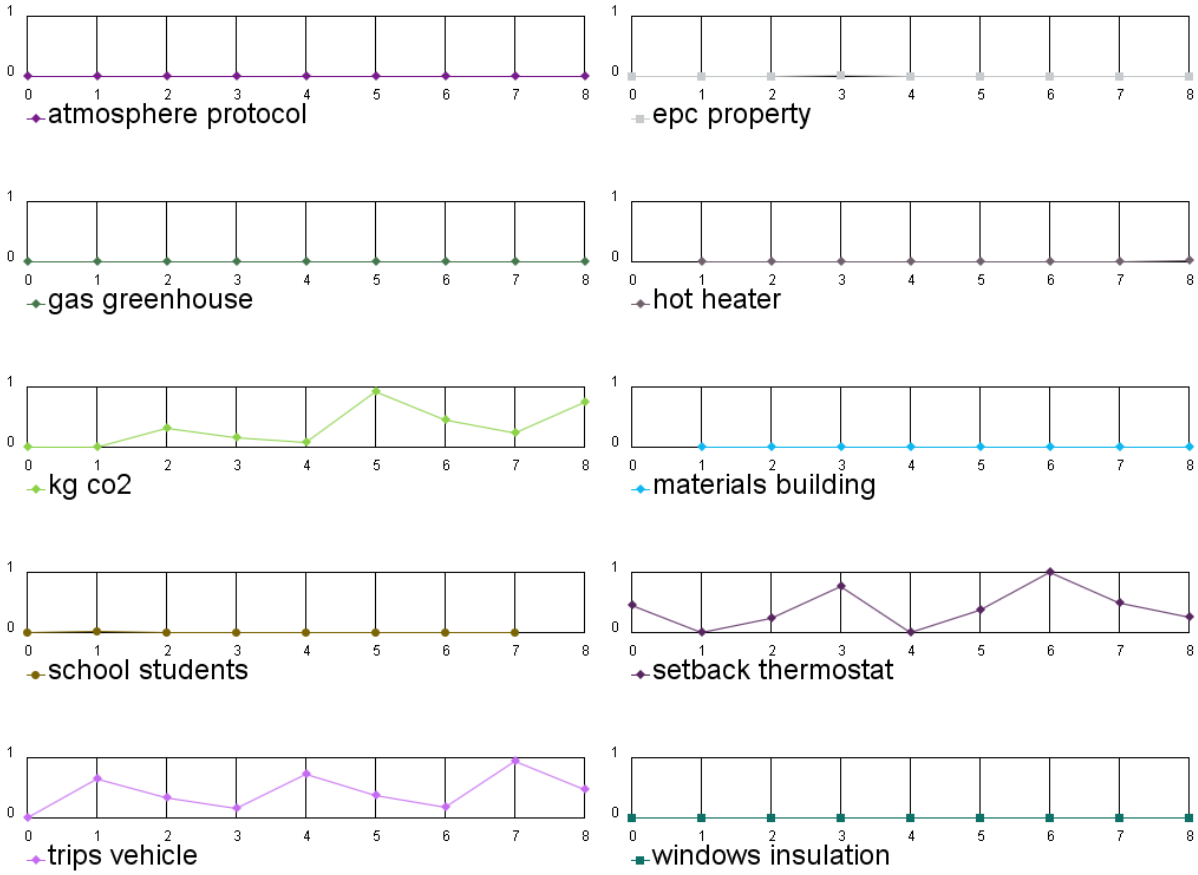


Abbildung 5.4.2: Jeder Themenverlauf in einem eigenen Diagramm. Auf der X-Achse ist die Framezahl aufgetragen, auf der Y-Achse die normalisierte Zentralität

5.5 Methoden der Evaluation

Die Validierung der Themenverläufe stellt eine Schwierigkeit dieser Diplomarbeit dar. Bisher haben sich wenige Arbeiten mit der Evaluation von Themenverläufen beschäftigt. Es gibt dementsprechend wenige publizierte Ansätze zur Validation. In [1] sollen Thementrends ähnlich zu dieser Diplomarbeit erkannt werden. Um den Algorithmus zu testen, werden Textsequenzen untersucht. Die Evaluation erfolgt nun durch zurückhalten von Dokumenten zu speziellen Themen, die ab einem vorher definierten Zeitpunkt in die Sequenz eingefügt werden. Entdeckt der Algorithmus diese Themen an dem definierten Zeitpunkt, wird dies als Erfolg gewertet.

Diese Art der Evaluation wird in der Diplomarbeit auch verwendet. Es werden jedoch nicht reale Dokumente aus einem Testkorpus benutzt und Dokumente zurückgehalten, sondern die Dokumente werden künstlich erzeugt. Es wird der generative Prozess des LDA-Modells genutzt, um Dokumente zu erzeugen, die definierte Themen mit einer festen Wahrscheinlichkeit enthalten. Erzeugt man mehrere Dokumente auf diese Weise, können Testdaten erzeugt werden,

die vorher definierte Themenverläufe aufweisen. Diese Testdaten werden dann dazu benutzt zu evaluieren, ob erstens die Verläufe tatsächlich entdeckt werden und wie gut sie durch den verwendeten Zentralitätsindex wiedergegeben werden. Dazu wird numerisch bestimmt, wie gut der Themenverlauf erfasst wird; einerseits von einer analytischen und andererseits von einer kognitiven Perspektive.

5.5.1 Synthetische Verläufe

Für jedes Themenmodell werden jeweils synthetische Textsequenzen erzeugt, anhand derer fest definierte Themenverläufe erzeugt werden. Es werden verschiedene Themenverläufe mit jeweils anderen Wahrscheinlichkeiten der Themen in den Dokumenten erzeugt. Insgesamt werden pro Themenmodell fünf Verläufe generiert, die sich in der Anzahl der Dokumente, der festgelegten Themen und deren Wahrscheinlichkeit unterscheiden.

3TSw 0.2: Eine Textsequenzen bestehend aus 120 Dokumenten. Jedes Dokument enthält drei Themen. Jeweils ein Thema und dessen Wahrscheinlichkeit ist vorher festgelegt. Die restlichen beiden Themen und deren Wahrscheinlichkeit werden zufällig ausgewählt. Es werden jeweils zehn Dokumente mit dem gleichen festgehaltenen Thema erzeugt. Die Wahrscheinlichkeit des Themas in den Dokumenten wird auf $0,2$ gesetzt. Nach 30 Dokumenten wird wieder das Thema der ersten zehn Dokumente gewählt. Es wird erwartet, dass bei einer Framegröße von zehn eine Sägezahnkurve auftritt, in der sich jeweils drei Themen abwechseln.

3TSw 0.8: Eine Textsequenzen wie in erstens, nur dass die Wahrscheinlichkeit des festgehaltenen Themas auf $0,8$ gesetzt wird. Auch hier wird wieder eine Sägezahnkurve erwartet jedoch mit anderen Werten der Zentralitätsindizes.

CT 0.2 : Eine Textsequenzen aus 100 Dokumenten die jeweils ein Thema festgelegt haben und zwei Themen zufällig gewählt. Die Wahrscheinlichkeit des festen Themas wurde auf $0,2$ gesetzt. Erwartet wird, dass das festgelegte Thema als prominentestes Thema auftritt und alle anderen Themen dominiert, da das Thema häufig zusammen mit anderen Themen auftritt.

CT 0.8 : Die Gleiche Textsequenzen wie in drittens nur das die Wahrscheinlichkeit auf $0,8$ gesetzt wurde.

3TS : Eine Textsequenzen von 100 Dokumenten mit drei Themen die gleichzeitig prominent sind. Die Wahrscheinlichkeit wurde für alle Themen auf $0,2$ gesetzt. Zusätzlich wurden zufällig drei Themen ausgewählt, auf die Restwahrscheinlichkeit von $0,4$ zufällig aufgeteilt wurde.

Da die wichtigen Themen in den Verläufen bekannt sind, kann numerisch ermittelt werden, wie gut diese vom System erkannt werden. Um dies zu evaluieren, wird ein dem Signal-Rausch-Verhältnis (SRV) verwandtes Maß benutzt, welches die Verläufe der relevanten Themen als Signal auffasst und die Zentralitätswerte als Signalstärke. Das Rauschen wird als der Mittelwert aller Themenverläufe aufgefasst. Betrachtet man nur die nicht relevanten Verläufe als Rauschen, unterscheiden sich diese oft nicht signifikant vom Mittelwert. Sie werden somit als relevant klassifiziert, obwohl sie im voraus nicht als solche definiert wurden.

Ein Verlauf stellt für eine Thema die ermittelten Zentralitätswerte über die Zeit dar. Dementsprechend kann ein Verlauf eines Themas als Vektor $\vec{v}$ aufgefasst werden. Die Komponenten stellen den Zentralitätswert zu einem bestimmten Zeitpunkt dar. So ist v_i der Zentralitätswert des Verlaufs zum Zeitpunkt i. Die Mittelwerte aller aller Verläufe werden mit $\vec{v_m}$ bezeichnet. Das Signal-Rausch-Verhältnis der einzelnen Werte ist definiert als

$$SRV = 10 \cdot \log_{10}\left(\frac{v_i}{v_{m_i}}\right)$$

und bewegt sich in diesem speziellen Anwendungsfall im Wertebereich $[-10, \ldots, 10]$.

Wenn für einen Wert eines relevanten Verlaufs das SRV größer als ein fester Wert t ist, wird dieser Wert als richtig positiver Fall angenommen. Ist der Wert kleiner als t wird er als falsch positiv angenommen. Für nicht relevante Verläufe wird auch das SRV bestimmt. Ein SRV-Wert der größer als t ist, wird als falsch positiv klassifiziert und ein Wert der kleiner als t ist, als richtig negativ. Zählt man die einzelnen Auftreten von richtig positiv, falsch positiv usw. kann man die Trefferquote

$$R = \frac{\text{richtig positiv}}{\text{richtig positiv} + \text{falsch negativ}}$$

und die Genauigkeit

$$P = \frac{\text{richtig positiv}}{\text{richtig positiv} + \text{falsch positiv}}$$

bestimmen. Aus Trefferquote und Genauigkeit kann dann ein Wert

$$F = \frac{2 \cdot P \cdot R}{P + R}$$

berechnet werden, der angibt wie gut ein Verlauf erkannt wurde.

In den ersten Versuchen die Themenverläufe numerisch zu bewerten, wurde ein konstanter Schwellwert benutzt, um die richtig positiven und falsch positiven Fälle zu ermitteln. Wenn der Zentralitätswert eines relevanten Verlaufes über dem Schwellwert lag wurde er als richtig positiv gewertet und wenn der Zentralitätswert eines nicht relevanten Verlauf über dem Schwellwert lag, wurde er als falsch positiv klassifiziert. Es gab allerdings Konfigurationen der Themenverläufe, in denen die nicht relevanten Verläufe einen hohen Zentralitätswert aufwiesen, die relevanten Verläufe aber klar unterschieden werden konnten. Eine solche Konfiguration ist in Abbildung

5.4.1 dargestellt. Hier ist im sechsten Frame das Thema *kg co2* das relevante und die beiden anderen Themen irrelevant. Sie würden aber als falsch positiv bewertet, obwohl sich das Thema *kg co2* klar von den beiden anderen Themen abhebt. Mit dem konstanten Schwellwert wurden somit die Bewertung schlechter, obwohl die die relevanten Verläufe klar als solche erkannt werden konnten.

Es wurde deshalb das SRV verwendet, um eine adaptive Methode zur Bewertung zu haben. Mit dem SRV werden solche Konfigurationen der Verläufe richtig bewertet, da es hier auf das Verhältnis von relevantem zu nicht relevantem Verlauf ankommt. Trotz der hohen Zentralitätswerte der nicht relevanten Verläufe ist die Bewertung hoch, da das Verhältnis zwischen relevantem Themenverlauf und nicht relevantem Themenverlauf hoch genug ist, um diese voneinander zu unterscheiden.

5.6 Zusammenfassung

In diesem Kapitel wurde dargelegt, wie die Anwendungsphase der entwickelten Methode funktioniert. Es wurden die einzelnen Teilschritte erläutert; insbesondere wie aus Texten ein Verlauf von Themen generiert wird, wie die Textsequenzen in Zeitabschnitte aufgeteilt werden, wie die Themen für die Dokumente in den Zeitabschnitten ermittelt werden, wie daraus wiederum Graphen aufgebaut werden und wie die Zentralitätsindizes auf diese Graphen angewendet werden und somit die Themenverläufe ermittelt werden. Zusätzlich wurde erläutert, wie die Themenverläufe visualisiert werden können und wie sie validiert werden können. Die Ergebnisse des in diesem Kapitel vorgestellten Algorithmus werden im nächsten Kapitel präsentiert. Anhand der Validierungskriterien wird ermittelt, welche Zentralitätsindizes geeignet sind, die Themenverläufe zu erfassen.

6 Ergebnisse

In diesem Kapitel werden die Ergebnisse der durchgeführten Experimente präsentiert. Zuerst wird die Eignung der Themenmodelle präsentiert. Anhand der Resultate werden die Themenmodelle ausgewählt, die dazu genutzt werden, die Themenverläufe zu erstellen. Anschließend werden die Ergebnisse der Evaluation der Zentralitätsindizes vorgestellt und die Themenverläufe der realen Textsequenzen abgebildet und diskutiert.

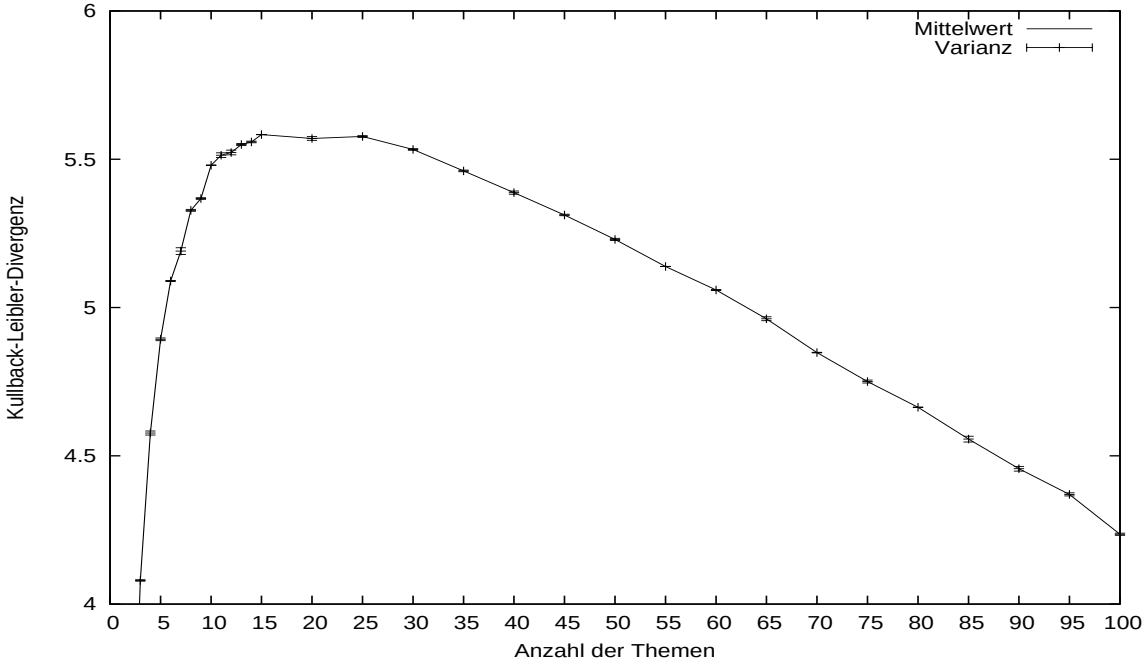
6.1 Evaluation der Themenmodelle

6.1.1 SCY-Chatdaten

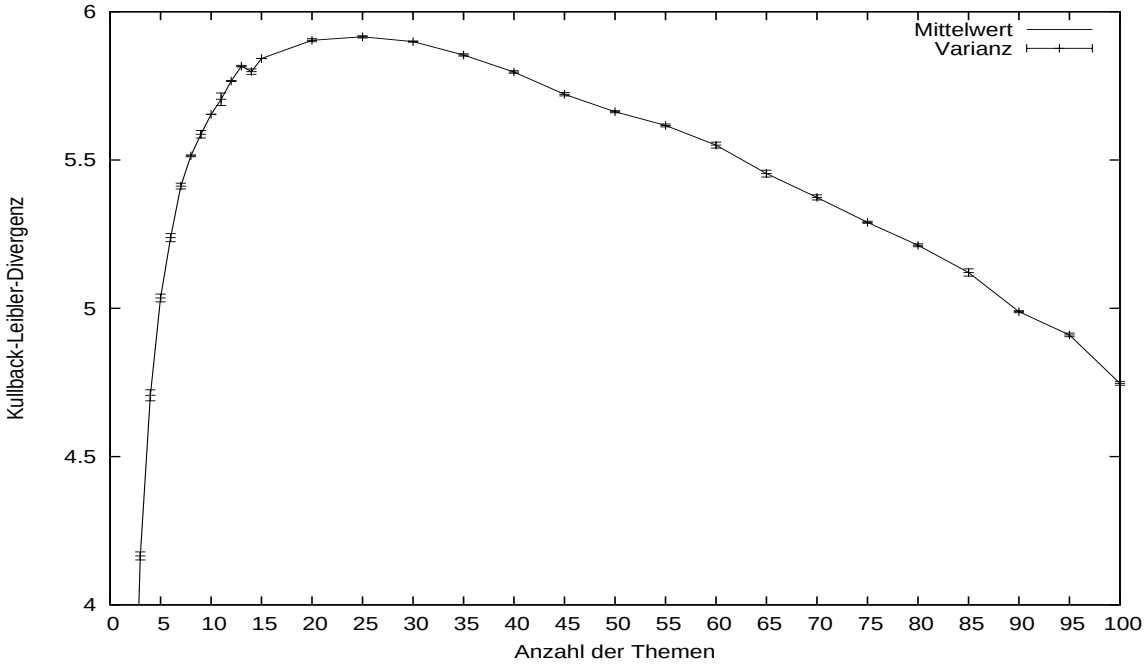
Für die SCY-Daten wurden zuerst Themenmodelle gelernt, die jeweils 5 bis 100 Themen enthalten. Ab einer Themenanzahl von 15 wurde nur jedes fünfte Modell gelernt, so wurden Modelle mit von 2-15 Themen, 20 Themen, 25 Themen, usw. gelernt. Dabei wurden verschiedene Parameter variiert. Zum einen wurden Stopwörter aus dem Trainingskorpus entfernt, zum anderen wurden die Terme auf ihre Stammform reduziert. Die Parameter α und β wurden auf Werte gesetzt, die im Allgemeinen gute Modelle ergeben [10]. Dies ist $\alpha = 50/K$ mit $K =$ Anzahl der Themen und $\beta = 0.01$. So wurden insgesamt vier Sätze von Themenmodellen gelernt.

Ein Satz von Modellen, in denen nur die Stopwörter entfernt wurden, ein Satz von Modellen, in denen sowohl Stopwörter entfernt wurden als auch die Terme auf ihre Grundform reduziert wurden, ein Satz von Modellen, bei denen die Terme nur auf ihre Grundform reduziert wurden und ein Satz von Modellen, bei denen weder die Stopwörter entfernt wurden als auch keine Grundformreduktion vorgenommen wurde. Auf jedes Modell in einem Satz wurde die in Abschnitt 4.2 erläuterte Validationsmethode angewendet. Für jedes Modell in einem Satz erhält man so eine Bewertungszahl.

Da sich die Ergebnisse des Approximationsalgorithmus, der verwendet wurde um die Themenmodelle zu lernen, in Abhängigkeit von den für den Gibbs-Sampler gewählten Startwerten geringfügig unterscheiden, wurden die verschiedenen Sätze insgesamt fünfmal neu gelernt. Dann wurde der Mittelwert und die Varianz aller fünf Evaluationsdurchgänge bestimmt. So sollen eventuelle Schwankungen bei der Approximation der Themen- und Termverteilungen ausgeglichen werden. Die berechneten Mittelwerte und Varianz wurden gegen die Anzahl der Themen in ein Diagramm eingetragen. Daraus kann man ablesen, welche Themenmodelle die geeignetsten sind, um möglichst klar voneinander abgegrenzte Themen zu erhalten. Für die verschiedene Sätze von



(a) Themenmodelle ohne Stammformreduktion



(b) Themenmodelle mit Stammformreduktion

Abbildung 6.1.1: Evaluationsergebnisse der Themenverläufe mit Stopwortentfernung für das SCY-Korpus. Es sind Mittelwert und Varianz aller Evaluationsdurchgänge abgetragen

Themenmodellen sind die Ergebnisse der Evaluation in Abbildung 6.1.1 und Abbildung 6.1.2 dargestellt.

Für alle Konfigurationen der Modelle steigt die Kullback-Leibler-Divergenzen stark an, hält sich kurzzeitig auf einem Plateau und fällt dann wieder ab. Das Plateau zeigt die optimale Anzahl der Themen an. Für die Modelle, die ohne Stammformreduktion trainiert wurden, ist erkennbar, dass die beste Themenanzahl im Bereich von 10 bis 25 Themen liegt. Für den Satz von Modellen, die mit Stammformreduktion der Terme gelernt wurden, wird eine höhere Kullback-Leibler-Divergenz erreicht. Für diese Modelle wird die optimale Themenanzahl um ca. fünf Themen nach oben verschoben. Die optimale Themenanzahl bewegt sich somit zwischen 15 und 30.

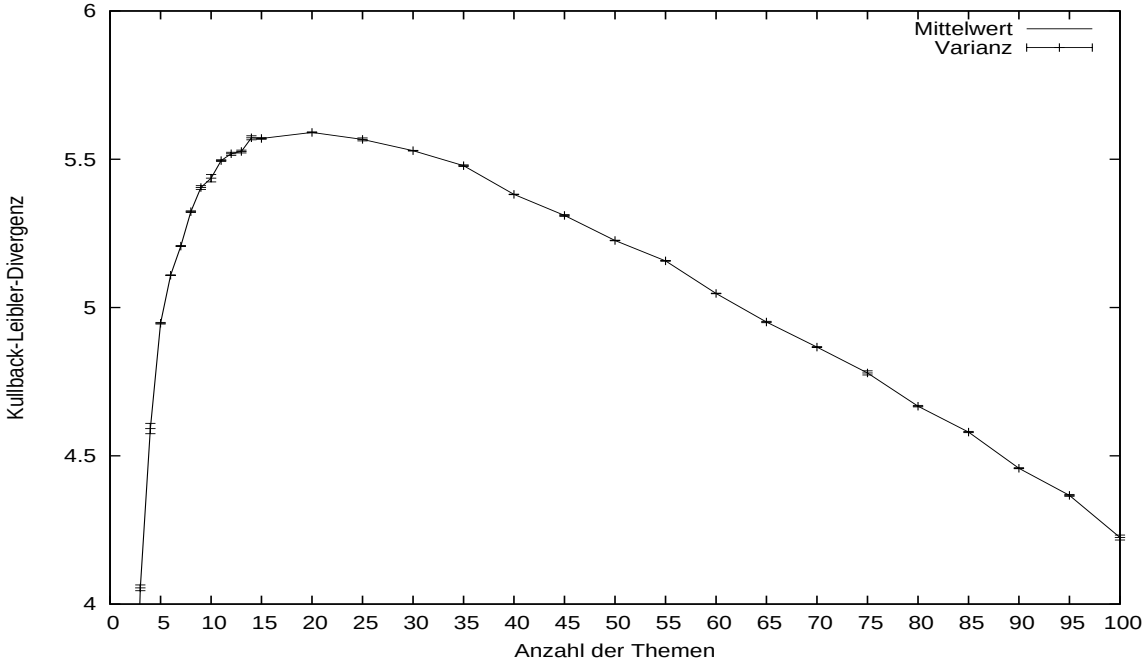
Alle Themenmodelle, deren Themenanzahl kleiner als zehn ist, repräsentieren die Hintergrundtexte nicht gut, da zu wenig Themen ausgewählt wurden, um die Komplexität der Texte zu erfassen. Die extrahierten Themen sind zu vage, um klar von einander abgegrenzt zu werden. Entsprechend ist es für Themenmodelle mit Anzahl der Themen größer als 25. Für diese werden die Terme auf zu viele Themen aufgeteilt. Die Themen werden zu speziell und somit auch nicht mehr interpretierbar.

6.1.2 dpa Nachrichtenmeldungen

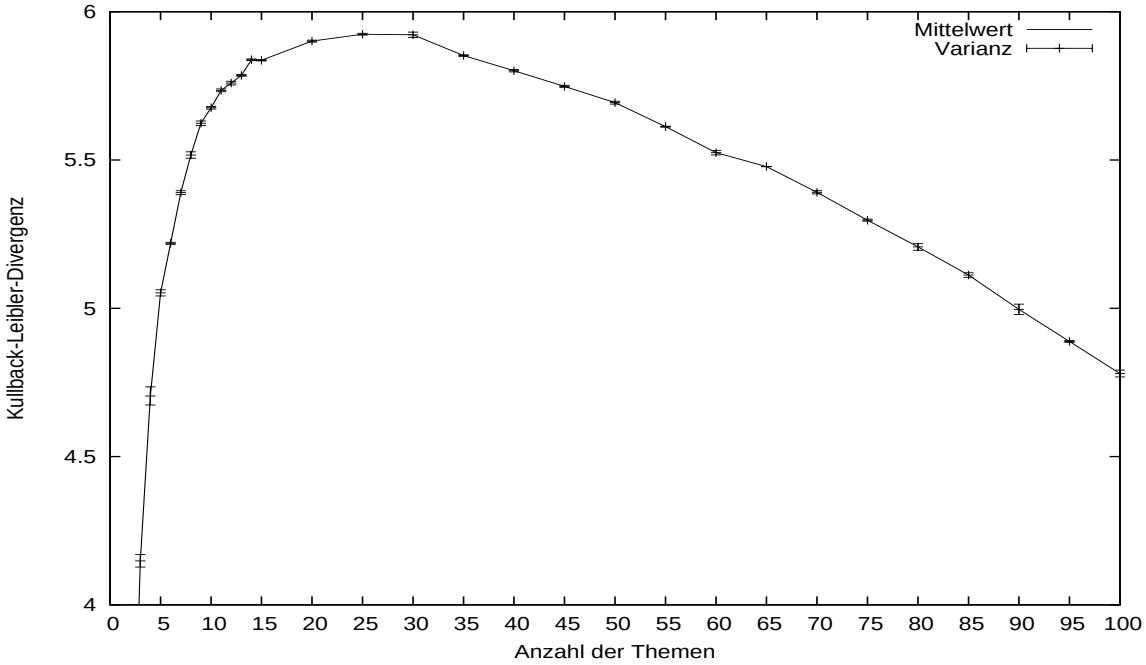
Für das dpa-Korpus wurden analog Themenmodelle mit einer Anzahl von 10 bis 200 Themen gelernt, wobei nur für jede zehnte Themenanzahl ein Modell gelernt wurde. Eine Ausnahme liegt im Bereich von 30 bis 40 Themen. Hier wurde jedes Themenmodelle gelernt, um das genaue Verhalten der Evaluationsmaße in diesem lokalen Bereich zu bestimmen. Es wurden auch wieder mehrere Sätze von Modellen gelernt, die sich durch die Parameter unterscheiden. Wie für das SCY-Korpus wurden Stopwörter entfernt und die Terme auf ihre Stammform reduziert. Die Parameter α und β wurden auf den Standardwerten belassen. Jeder Satz wurde insgesamt fünfmal anhand desselben Korpus und derselben Parameter neu gelernt. Aus den ermittelten Kullback-Leibler Divergenzwerten für die einzelnen Modelle wurde der Mittelwert und die Varianz berechnet. Diese wurden dann auch gegen die Themenanzahl aufgetragen. Die resultierenden Diagramme sind in Abbildung 6.1.3 und Abbildung 6.1.4 zu sehen.

Die Kurven zeigen das schon bei den SCY-Modellen gesehene Verhalten. Sie steigen recht schnell an, bewegen sich dann kurzzeitig auf einem Plateau und fallen dann wieder ab. Auch hier zeigt das Plateau die optimalen Anzahl von Themen an. Der optimale Bereich liegt für die Modelle, die ohne Stammformreduktion gelernt wurden, zwischen 40 und 70 Themen. Für die Modelle, die eine Stammformreduktion der Terme beinhalten, liegt der optimale Bereich zwischen 50 und 90.

Interessant ist der Bereich zwischen 30 und 40. Die Kurve weist hier starke Schwankungen bei hoher Varianz auf. Zum einen lassen sich die Schwankungen damit erklären, dass die Sätze nur fünfmal neu gelernt werden. Würden die Sätze öfter neu gelernt, würde der Mittelwert sich

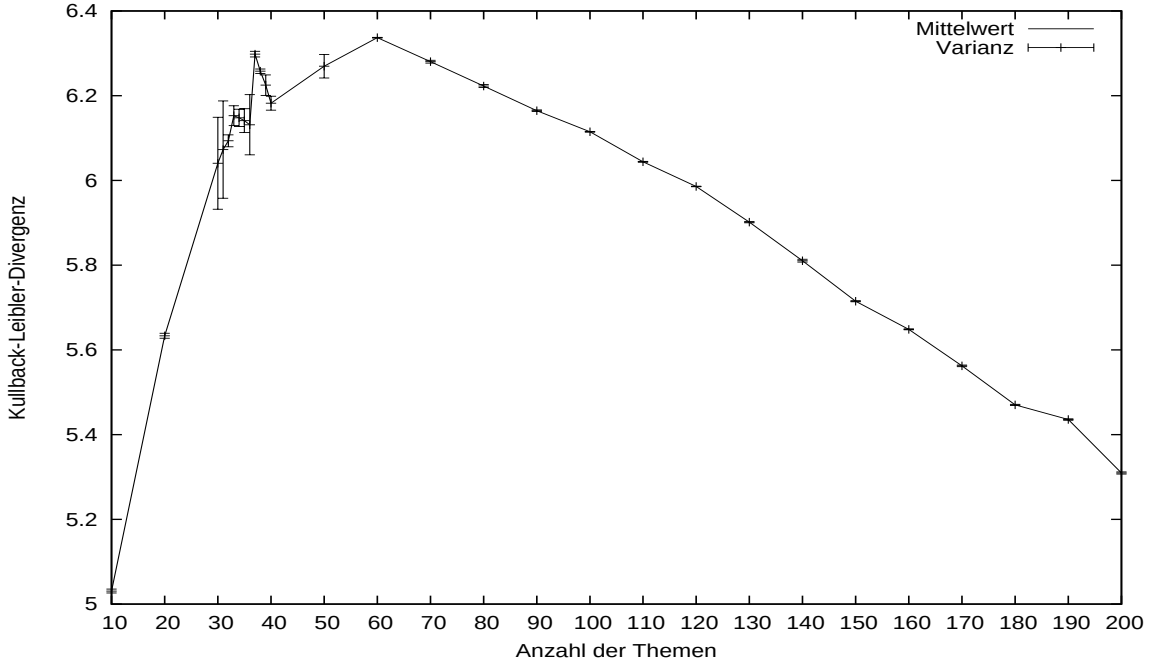


(a) Themenmodelle ohne Stammformreduktion

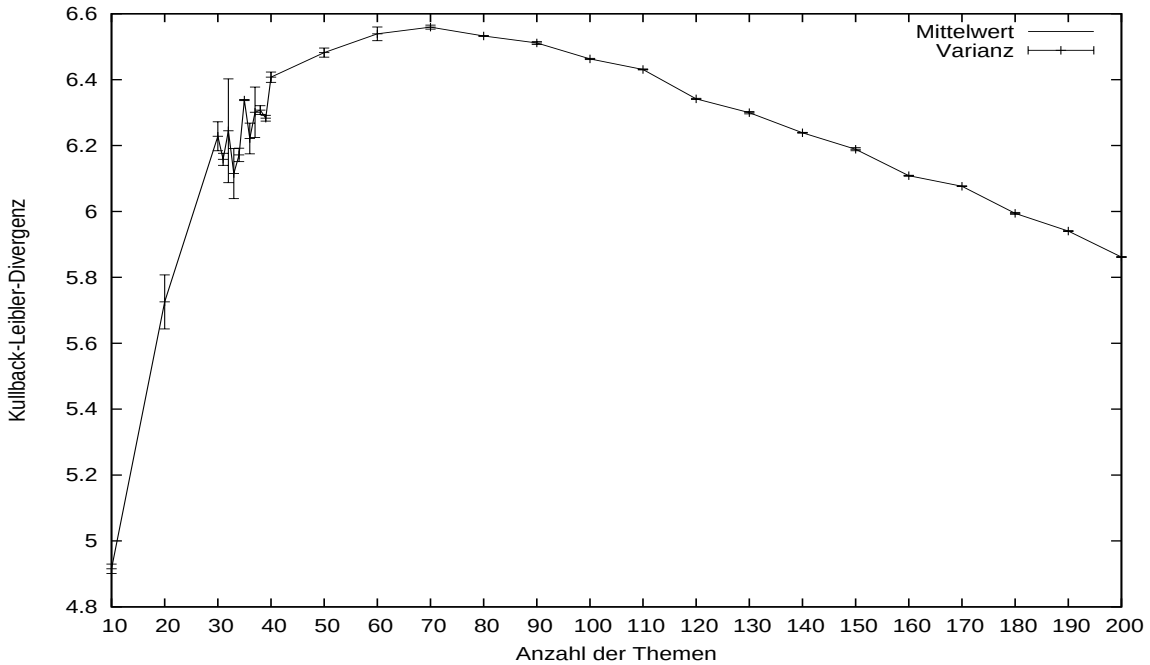


(b) Themenmodelle mit Stammformreduktion

Abbildung 6.1.2: Evaluationsergebnisse der Themenverläufe ohne Stopwortentfernung für das SCY-Korpus. Es sind Mittelwert und Varianz aller Evaluationsdurchgänge abgetragen



(a) Themenmodelle ohne Stammformreduktion



(b) Themenmodelle mit Stammformreduktion

Abbildung 6.1.3: Evaluationsergebnisse der Themenverläufe mit Stopwortentfernung für das dpa-Korpus. Es sind Mittelwert und Varianz aller Evaluationsdurchgänge abgetragen

wahrscheinlich dem idealen Verlauf der Kurve annähern. Andererseits schwanken die Kullback-Leibler-Werte aufgrund des verwendeten Approximationsalgorithmus für die Themen- und Termverteilungen. Der Gibbs-Sampling Algorithmus weist die Terme den Themen zufällig zu. Zwar konvergiert der Algorithmus gegen die gesuchte Verteilung, es kann dabei aber zu Schwankungen bei der Zuweisung von Termen zu Themen, kommen. So ist es möglich, dass die Themenanzahl einmal eine gute Trennung der Themen ergibt und in einem andern Fall keine gute Trennung erreicht wird. Dieser Effekt tritt bei den SCY-Texten nicht auf. Dies kann an der Anzahl von Termen im Korpus liegen. Da im SCY-Korpus nur ca. 8.000 Terme unterschieden werden und im dpa-Korpus mehr als 100.000 Terme vorhanden sind, können größere Schwankungen auftreten. Dies genauer zu untersuchen, würde jedoch den Fokus der Diplomarbeit verlassen.

Generell kann man sagen, dass die optimale Anzahl der Themen steigt, wenn eine Stammformreduktion durchgeführt wird. Die Entfernung der Stopwörter hat hingegen keinen Einfluss auf die Eignung der Themenmodelle.

6.2 Evaluation der Zentralitätsverläufe

Anhand der in Abschnitt 6.1 als geeignet identifizierten Themenmodelle und der synthetisch erstellten Dokumentverläufe werden nun die Zentralitätsmaße zusammen mit dem Graphenalgorithmus ermittelt, die geeignet sind die Themenveränderung wiederzugeben. Dazu wurden, wie schon in Abschnitt 5.5 beschrieben, die Themenverläufe für die synthetisch erstellten Dokumente ermittelt und anhand des F-Maßes bewertet wie gut diese erfasst und repräsentiert werden. Der Schwellwert t wurde dabei auf 1.0 gesetzt. Von den für jedes Modell berechneten F-Werten wird dann wiederum der Mittelwert bestimmt.

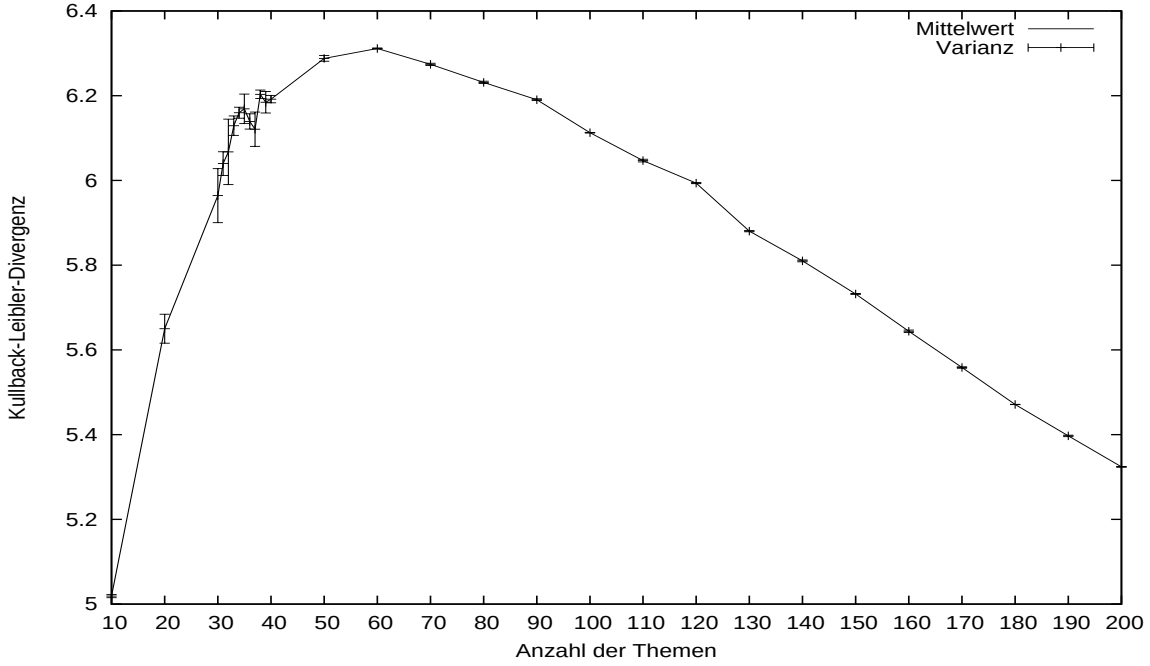
Für jeden in Abschnitt 5.2 erläuterten Algorithmus wird so eine Tabelle erstellt, die anzeigt welches Zentralitätsmaß, welchen Verlauf wie gut erfasst. Die Tabellen werden noch nach den Parametern des zugrundeliegenden Themenmodells unterschieden. Es werden also pro Korpus zwölf Tabellen erstellt. Wenn sich die Werte der mit verschiedenen Parametern trainierten Modelle nicht signifikant unterscheiden, wird aus Platzgründen nur eine Tabelle pro Algorithmus dargestellt.

6.2.1 SCY-Chatdaten

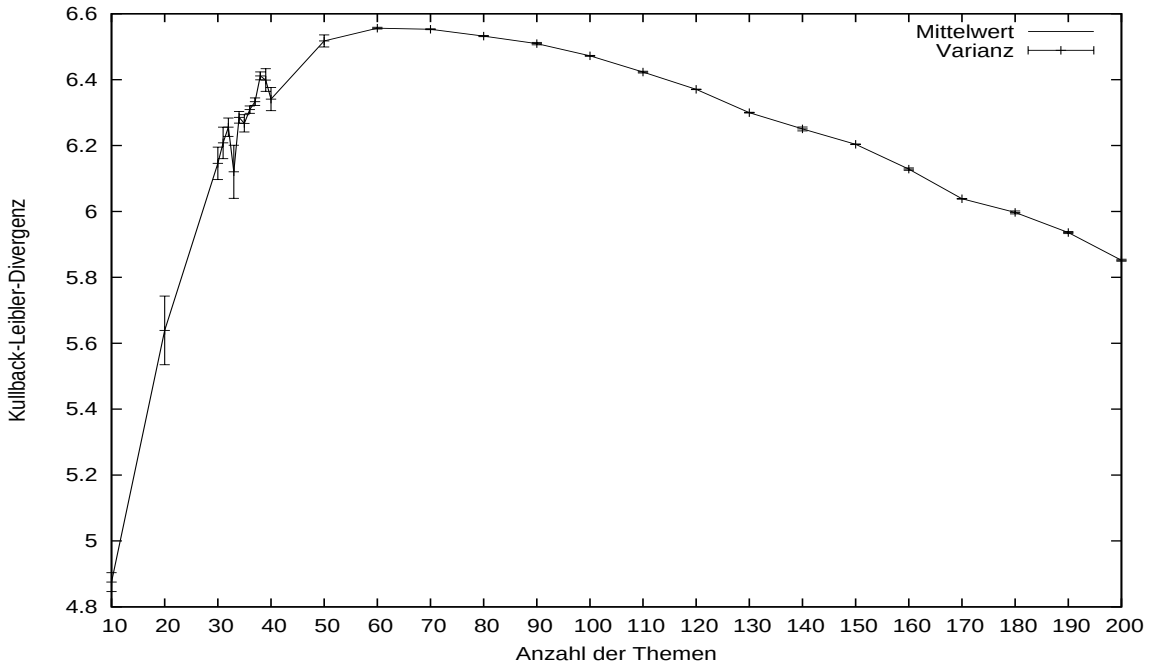
Für die synthetischen SCY-Chatdaten unterscheiden sich die Bewertungen der einzelnen Sätze nicht signifikant. Es werden deshalb nur die Ergebnisse jeweils eines Modellsatzes präsentiert.

Für den Graphenalgorithmus VVG sind die Ergebnisse der Bewertung in Tabelle 6.2.1 dargestellt. Die Ergebnisse des Algorithmus DZG werden in Tabelle 6.2.2 dargestellt und die Ergebnisse, die mit dem Algorithmus GMT erreicht wurden, in Tabelle 6.2.3

Die Spalten geben die verschiedenen verwendeten Zentralitäten (siehe Abschnitt 2.2) an und die Zeilen geben die synthetisch erstellten Dokumentkollektionen (siehe Abschnitt 5.5) an. Für



(a) Themenmodelle ohne Stammformreduktion



(b) Themenmodelle mit Stammformreduktion

Abbildung 6.1.4: Evaluationsergebnisse der Themenverläufe ohne Stopwortentfernung für das dpa-Korpus. Es sind Mittelwert und Varianz aller Evaluationsdurchläufe abgetragen

	c_B	c_C	c_D	c_E	c_H	c_{PR}	c_{SH}	c_R	c_S
CT 0.2	**1,00**	0,37	0,15	0,15	0,17	0,16	0,00	0,23	0,00
CT 0.8	**0,99**	0,33	0,19	0,20	0,20	0,20	0,01	0,24	0,01
3TS	**1,00**	0,74	0,34	0,35	0,39	0,36	0,02	0,43	0,00
3TSw 0.2	**0,57**	0,37	0,17	0,17	0,17	0,16	0,01	0,25	0,00
3TSw 0.8	**0,52**	0,29	0,17	0,15	0,18	0,16	0,02	0,20	0,02

Tabelle 6.2.1: F-Werte für Themenverläufe, die mit dem Graphenalgorithmus VVG (Abschnitt 5.2.1) erstellt wurden.

	c_B	c_C	c_D	c_E	c_H	c_{PR}	c_{SH}	c_R	c_S
CT 0.2	**1,00**	0,36	0,48	0,34	0,40	0,48	0,40	0,31	0,06
CT 0.8	**1,00**	0,34	0,47	0,32	0,33	0,48	0,32	0,29	0,15
3TS	**1,00**	0,77	0,88	0,66	0,78	0,88	0,55	0,58	0,00
3TSw 0.2	**0,64**	0,39	0,49	0,38	0,39	0,49	0,36	0,32	0,07
3TSw 0.8	**0,57**	0,31	0,44	0,32	0,31	0,46	0,32	0,26	0,13

Tabelle 6.2.2: F-Werte für Themenverläufe, die mit dem Graphenalgorithmus DZG (Abschnitt 5.2.2) erstellt wurden.

	c_B	c_C	c_D	c_E	c_H	c_{PR}	c_{SH}	c_R	c_S
CT 0.2	**1,00**	0,39	0,48	0,31	0,40	0,49	0,38	0,33	0,10
CT 0.8	**1,00**	0,36	0,48	0,32	0,33	0,48	0,32	0,31	0,19
3TS	**0,99**	0,80	0,88	0,64	0,78	0,88	0,57	0,61	0,13
3TSw 0.2	**0,58**	0,39	0,49	0,37	0,39	0,50	0,39	0,30	0,10
3TSw 0.8	**0,55**	0,32	0,44	0,30	0,31	0,47	0,31	0,28	0,15

Tabelle 6.2.3: F-Werte für Themenverläufe, die mit dem Graphenalgorithmus GMT (Abschnitt 5.2.3) erstellt wurden.

alle Algorithmen gilt, dass die Betweenness-Zentralität c_B die künstlichen Verläufe am besten erfasst. Der Verlauf, in dem drei Themen abwechselnd im Fokus sind, wird schlechter bewertet. Dies liegt daran, dass viele Zentralitätswerte als falsch positiv gewertet werden. Diese Werte werden jedoch meistens von einem der drei abwechselnd relevanten Themen erreicht. Die nicht relevanten Themen werden immer als richtig negativ klassifiziert. In Abbildung 6.2.1 ist der Verlauf visuell dargestellt. Man kann sehen, dass die relevanten Themen trotzdem noch gut unterschieden werden können.

Die Zentralitätsindizes, die nur die Struktur des Graphen bewerten und nicht die Kantengewichte mit einbeziehen, zeigen klare Verbesserungen beim Algorithmus DZG und GMT, die keinen vollständig verbundenen Graphen mehr erzeugen. Jedoch erfasst die Betweenness-Zentralität die Verläufe immer noch am besten. Dies liegt unter anderem daran, dass die nicht relevanten Themen von der Betweenness-Zentralität unterdrückt werden, während die anderen

Zentralitätsindizes die nicht relevanten Themen noch moderat bewerten und die Themenverläufe somit nicht mehr so klar voneinander zu unterscheiden sind.

6.2.2 dpa Nachrichtenmeldungen

Für die dpa-Nachrichtenmeldungen erweist sich wieder die Betweenness-Zentralität als am geeignetsten. Im folgenden wird nur das Ergebnisse für einen Satz von Themenmodellen für die Algorithmen VVG und DZG gezeigt und die Bewertung für die Betweenness-Zentralität. Alle anderen Bewertungen der Zentralitäten unterscheiden sich nicht signifikant und werden deshalb hier nicht gezeigt.

	c_B	c_C	c_D	c_E	c_H	c_{PR}	c_{SH}	c_R	c_S
CT 0.2	**1,00**	0,14	0,14	0,13	0,14	0,15	0,08	0,15	0,07
CT 0.8	**0,95**	0,17	0,18	0,17	0,18	0,18	0,16	0,19	0,13
3TS	**1,00**	0,37	0,36	0,33	0,37	0,37	0,26	0,39	0,05
3TSw 0.2	**0,36**	0,09	0,08	0,08	0,09	0,09	0,04	0,10	0,05
3TSw 0.8	**0,32**	0,10	0,08	0,10	0,10	0,10	0,08	0,11	0,07

Tabelle 6.2.4: F-Werte für Verläufe, die mit dem Algorithmus VVG ermittelt wurden.

	c_B	c_C	c_D	c_E	c_H	c_{PR}	c_{SH}	c_R	c_S
CT 0.2	**1,00**	0,15	0,16	0,15	0,16	0,16	0,16	0,15	0,14
CT 0.8	**1,00**	0,19	0,20	0,19	0,19	0,19	0,19	0,19	0,17
3TS	**1,00**	0,38	0,40	0,38	0,40	0,40	0,40	0,38	0,11
3TSw 0.2	**0,37**	0,10	0,11	0,10	0,10	0,11	0,11	0,10	0,08
3TSw 0.8	**0,32**	0,11	0,12	0,11	0,11	0,12	0,12	0,11	0,08

Tabelle 6.2.5: F-Werte für Verläufe, die mit dem Algorithmus DZG ermittelt wurden.

Wie man anhand der Tabellen 6.2.4 und 6.2.5 sehen kann, zeigt sich wieder der Anstieg in der Bewertung für Zentralitätsmaße, die nur die Struktur bewerten. Für den Algorithmus DZG verdoppelt sich die Bewertung der Stress-Zentralität und der Shaffer-Zentralität. Die Werte dieser Zentralitäten unterscheiden sich aber nicht signifikant für die Werte der Verläufe, die mit dem Graphenalgorithmus GMT erstellt wurden. Deshalb werden die F-Werte für den Algorithmus GMT nicht in einer eigenen Tabelle dargestellt.

Im Allgemeinen ist die Betweenness-Zentralität der am besten bewertete Zentralitätsindex. Die drei Verläufe *CT 0.2*, *CT 0.8* und *3TS* werden korrekt erkannt und werden auch nicht durch andere Themen gestört. Die beiden Verläufe *3TSw 0.2* und *3TSw 0.8* werden nicht sehr hoch bewertet. Dies liegt, wie bei den SCY-Daten, an für den aktuellen Zeitabschnitt nicht relevanten Themenverläufen, die als falsch positiv gewertet werden. Die visuelle Inspektion der Verläufe zeigt auch hier wieder, dass die relevanten Themen erfasst werden, jedoch zeigen die Verläufe nicht die erwartete Sägezahnkurve, wie sie es bei den SCY-Daten tun. Die relevanten Themen

sind jedoch noch gut von den nicht relevanten Themen zu unterscheiden. In Abbildung 6.2.1 ist ein künstlicher Verlauf für die *3TSw 0.2* SCY-Daten dargestellt. Die künstlichen Verläufe für das dpa-Korpus sehen ähnlich aus. Man kann die relevanten Themen gut voneinander unterscheiden, jedoch ist die numerische Bewertung schlecht.

	1. Algorithmus VVG			2. Algorithmus DZG			3. Algorithmus GMT			
CT 0.2	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	0,99
CT 0.8	0,95	0,92	0,95	1,00	0,99	1,00	0,99	1,00	0,98	0,99
3TS	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
3TSw 0.2	0,44	0,53	0,26	0,41	0,55	0,20	0,39	0,43	0,53	0,26
3TSw 0.8	0,56	0,31	0,22	0,63	0,31	0,22	0,31	0,59	0,33	0,23

Tabelle 6.2.6: F-Werte der Betweenness-Zentralität für die drei verschiedenen Algorithmen und Modelle mit verschiedenen Parametern. Zur Erläuterung der Spalten siehe Text.

Die restlichen Werte der Betweenness-Zentralität sind in Tabelle 6.2.6 dargestellt. Die Spalten sind folgendermaßen angeordnet. Für den Algorithmus VVG zeigt die erste Spalte die Themenmodelle, die ohne Stopwortentfernung und ohne Stammformreduktion trainiert wurden, die zweite Spalte zeigt die Modelle, die mit Stopwortentfernung und mit Stammformreduktion trainiert wurden und die dritte Spalte zeigt die Modelle, die ohne Stopwortentfernung und mit Stammformreduktion erstellt wurden. Dasselbe wird beim Algorithmus DZG gezeigt. Beim Algorithmus GMT zeigen die zweite bis vierte Spalte diese Modelle. Die erste Spalte zeigt die Modelle, die mit Stopwortentfernung aber ohne Stammformreduktion trainiert wurden. Da die Werte für diese Modelle für die beiden Algorithmen VVG und DZG bereits in Tabelle 6.2.4 und 6.2.5 dargestellt wurden, werden sie hier nicht wiederholt. Der in den ersten beiden Tabellen beobachtete Trend setzt sich fort. Die Verläufe *CT 0.2*, *CT 0.8* und *3TS* weisen einen Wert nahe Eins auf, während die beiden Verläufe *3TSw 0.2* und *3TSw 0.8* zwischen 0,2 und 0,6 schwanken.

6.2.3 Schlussfolgerung

Aus der Bewertung der Zentralitätsindizes in Abhängigkeit vom verwendeten Algorithmus zur Erstellung der Graphen lässt sich folgendes feststellen.y Das geeignetste Zentralitätsmaß ist die Betweenness-Zentralität. Welcher Algorithmus benutzt werden sollte, um die Graphen aus der Themenkookkurrenz aufzubauen, hängt von der verwendeten Zentralität ab. Aufgrund der Ergebnisse der Bewertung ist es für die Betweenness und die Closeness-Zentralität unerheblich, welcher Algorithmus gewählt wurde. Die visuelle Inspektion der nachfolgenden realen Verläufe zeigt jedoch, dass die letzten beiden Algorithmen visuell bessere Ergebnisse liefern. Für die restlichen Zentralitätsindizes zeigt sich, dass der Algorithmus DZG und GMT bessere Ergebnisse liefert als der erste.

Mit welchen Parametern das Themenmodell trainiert werden soll, lässt sich aus den Bewertungen nicht ermitteln. Für die SCY-Daten weichen die Bewertungen für die verschiedenen

Modellsätze so wenig voneinander ab, dass hier gar keine Aussage getroffen werden kann und für die dpa-Text lässt sich nur feststellen, dass eine Stammformreduktion ohne Entfernen der Stopwörter die schlechtesten Ergebnisse geliefert hat. Dies lässt sich jedoch nicht verallgemeinern und hängt zu großen Teilen vom zugrundeliegenden Korpus ab.

6.3 Anwendungsphase

Ausgehend von den in Abschnitt 6.1 und Abschnitt 6.2 vorgestellten Ergebnissen werden nun die realen Verläufe aus dem SCY-Projekt und den dpa-Nachrichtenmeldungen dargestellt. Die Verläufe wurden mit verschiedener Framegröße und Überlappung generiert. Es werden exemplarisch einige Verläufe gezeigt.

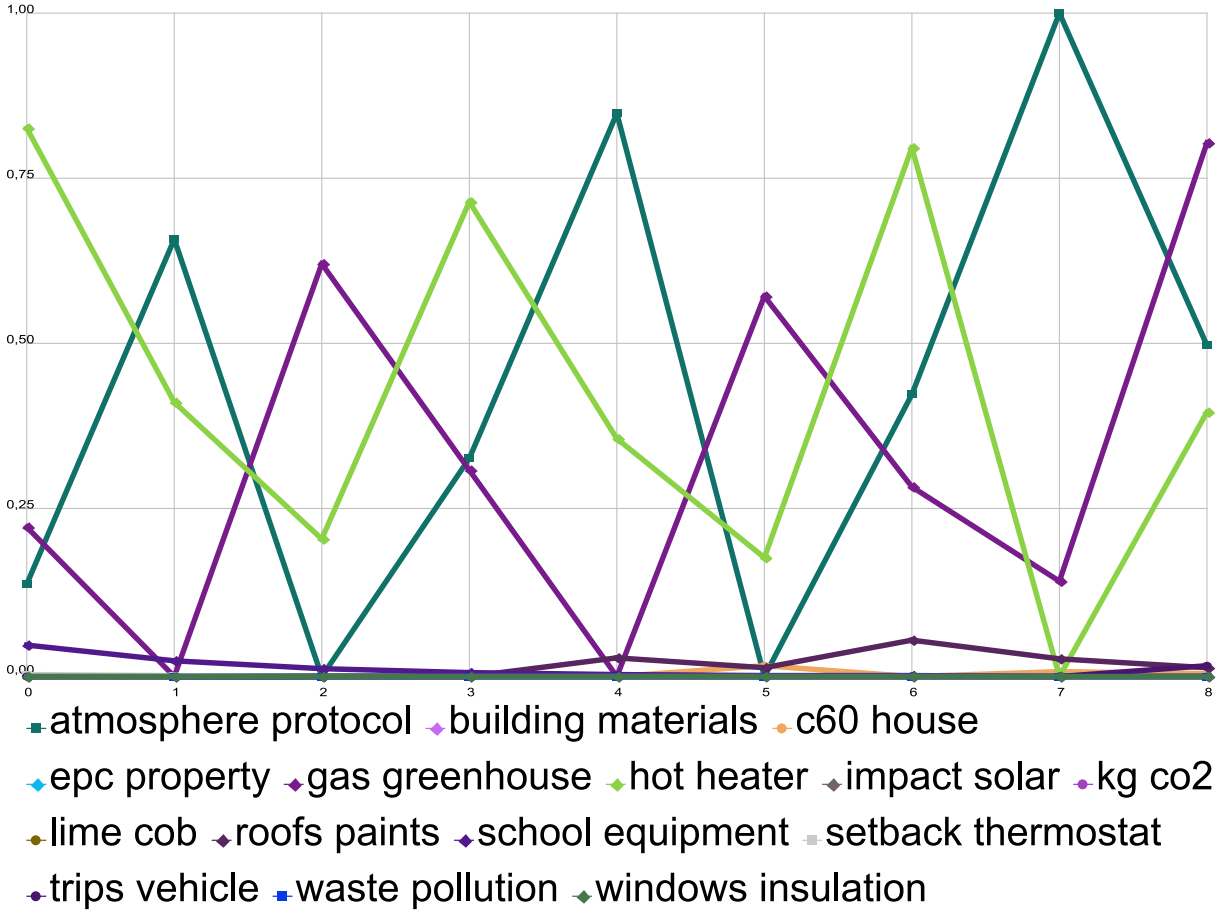


Abbildung 6.2.1: Verlauf für synthetisch erzeugten Dokumentensequenz. Der künstlich erzeugte Verlauf wurde für ein Themenmodell mit 15 Themen erstellt und ist vom Typ *3TSw 0.2*.

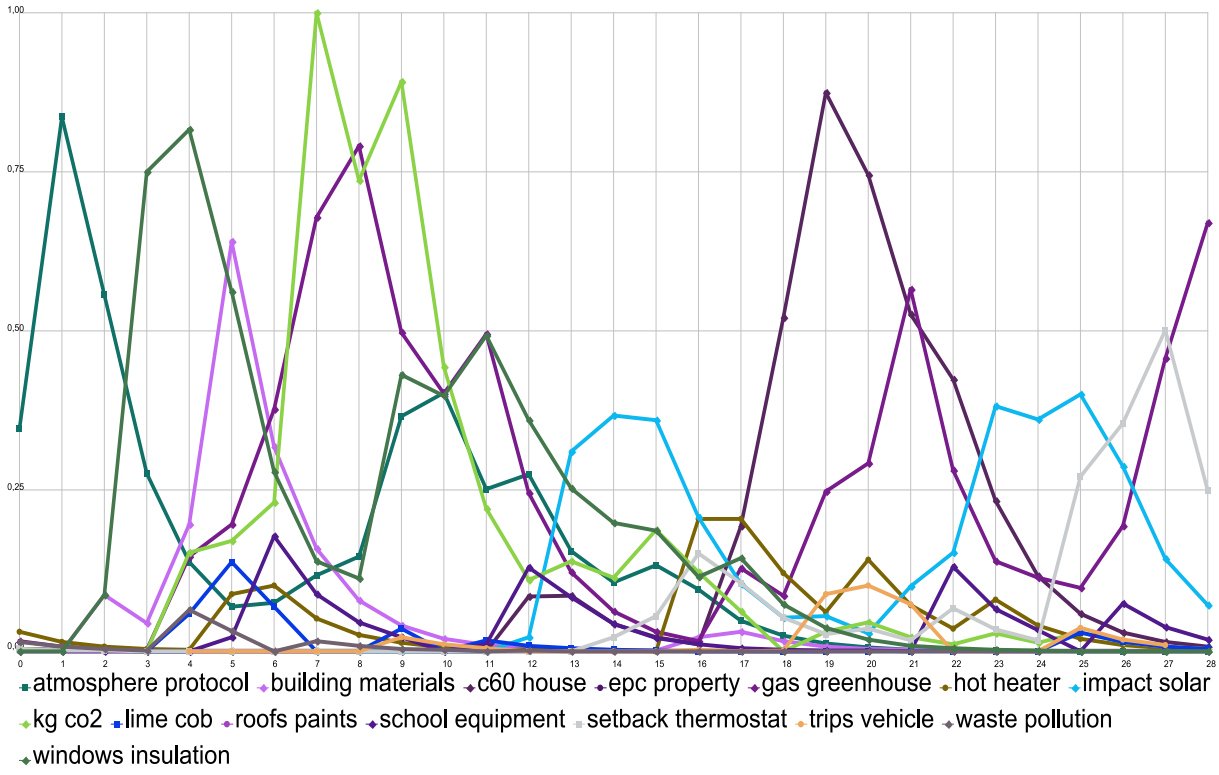


Abbildung 6.2.2: Verlauf der Betweenness-Zentralität. Das zugrundeliegende Themenmodell wurde mit 15 Themen mit Entfernung der Stopwörter aber ohne Reduktion der Terme auf ihre Stammform trainiert. Der Verlauf wurde mit dem Algorithmus GMT erzeugt.

6.3.1 SCY Chatdaten

Für die SCY-Chatdaten wurde eine Framegröße von sieben und eine Überlappung von zwei benutzt. Diese Werte wurden experimentell ermittelt und ergaben die besten Ergebnisse. In allen Abbildungen ist auf der X-Achse die Framenummer abgetragen und auf der Y-Achse die normalisierten Zentralitätswerte. Es wird nur ein Verlauf für beide Methoden der Visualisierung dargestellt. Dies zeigt die Unterschiede und Vor- und Nachteile der beiden Arten der Visualisierung.

In Abbildung 6.2.2 sind die Verläufe der Themen für die SCY-Chatdaten dargestellt. Durch die Natur der Chattexte wechseln sich viele Themen sehr schnell ab. Man kann aber erkennen, dass in den Frames 0 bis 10 über die Isolation von Häusern, die dazu verwendeten Baumaterialien und die Auswirkungen auf die CO_2-Neutralität gesprochen wird. In den Frames 10 bis 17 sind die Themen über Sonneneinstrahlung und Isolation vorherrschend. Ab Frame 18 wird sich hauptsächlich über die Reduzierung von Emissionen und den Zusammenhang zwischen Treibhauseffekt und Sonneneinstrahlung und die Auswirkungen auf die Temperatur gesprochen. Die restlichen Themen treten hier in den Hintergrund.

In Abbildung 6.3.1 sind die Themen einzeln dargestellt. Für diese Verläufe bietet sich die einzelne Darstellung an, da so differenzierter festgestellt werden kann, über welche Themen gesprochen wird und nicht ganz so wichtigen Themen, nicht in den Hintergrund gedrängt werden.

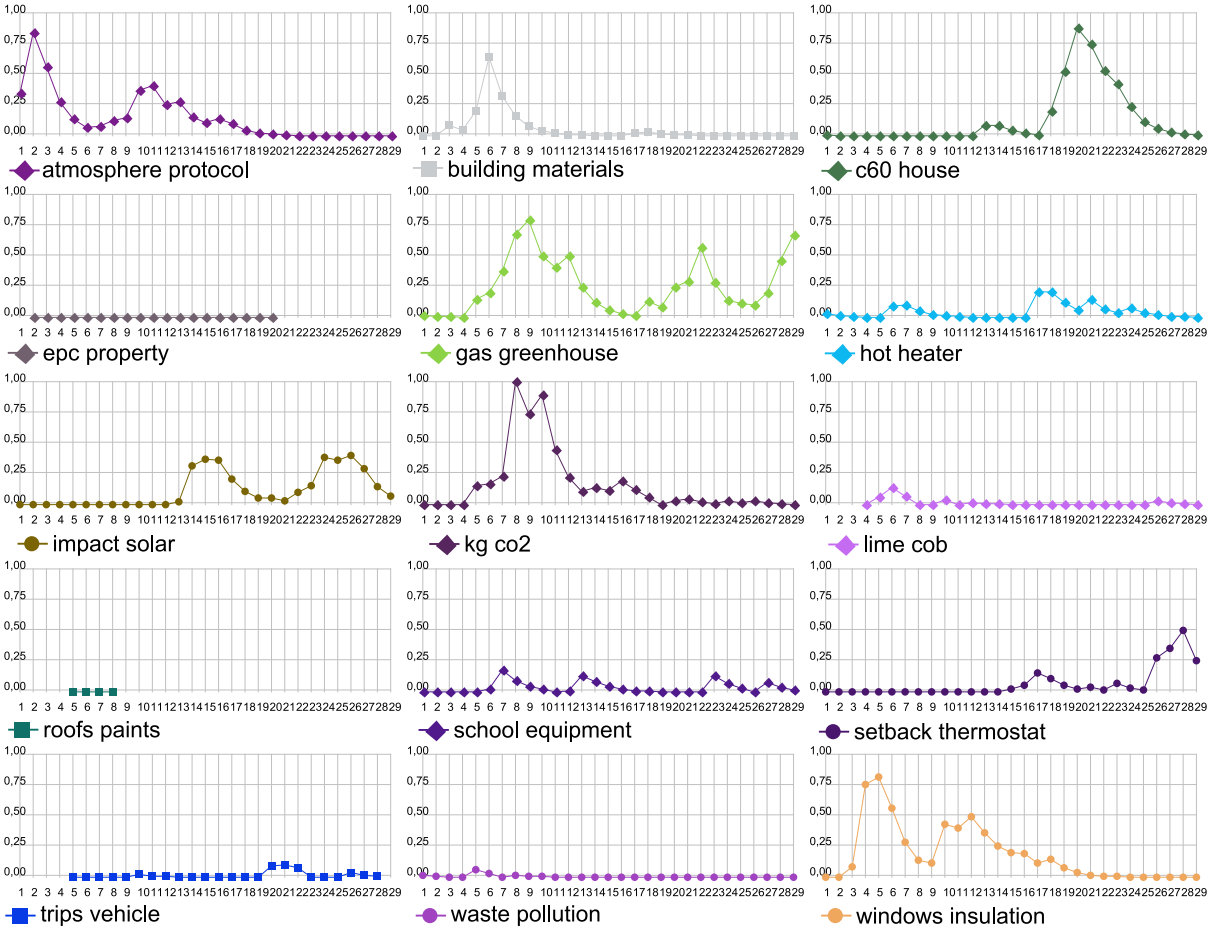


Abbildung 6.3.1: Die zu Abbildung 6.2.2 entsprechende separate Darstellung der Themenverläufe. Hier können die moderat wichtigen Themen besser erkannt werden.

6.3.2 dpa Nachrichtenmeldungen

Für die dpa-Nachrichtenmeldungen wurde zum einen eine Framegröße von 100 mit einer Überlappung von 50 gewählt und zum anderen eine Framegröße von 50 und eine Überlappung von 25. Die Verläufe für verschiedenen zugrundeliegenden Modelle unterscheiden sich nicht sehr stark und es wird deshalb auf die Präsentation der Verläufe für unterschiedliche Modelle verzichtet.

Wie erwartet, wird das Thema über das Erdbeben in Südasien in beiden Verläufen erkannt und ist sehr präsent. Am Anfang ist es wenig bis gar nicht präsent und steigt dann stark an und bleibt bis zum Ende des untersuchten Zeitraumes im Fokus. Die restlichen drei präsenten Themen sind Themen, die Terme gruppieren, die inhaltlich keinem anderen Thema zugeordnet werden

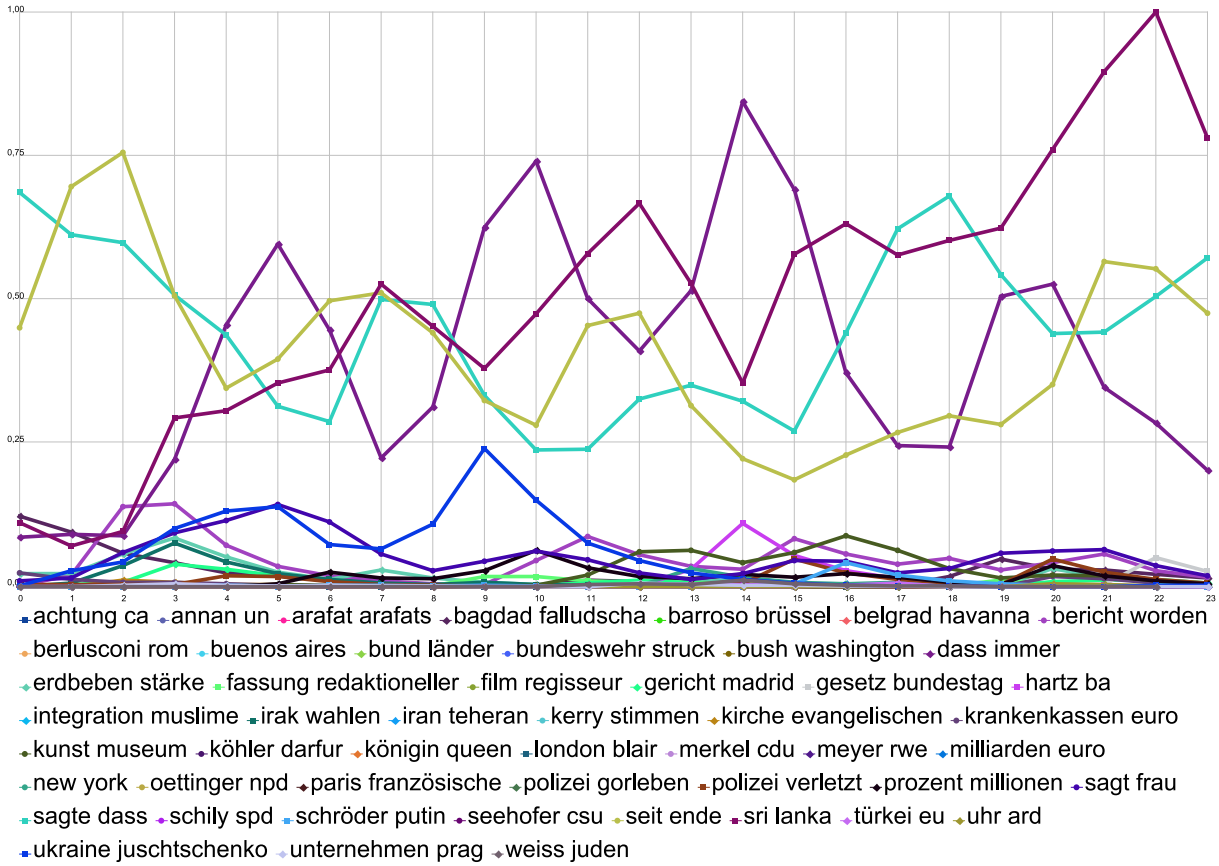


Abbildung 6.3.2: Themenverläufe der Betweenness-Zentralität für ein Themenmodell mit 50 Themen, ohne Stammformreduktion, mit Entfernung der Stopwörter. Die Framegröße wurde auf 100 gesetzt und die Überlappung auf 50. Die Verläufe wurden mit dem Algorithmus GMT erstellt.

konnten. Sie bestehen meist aus Funktionswörtern, die in allen Dokumenten auftreten. Da diese Funktionswörter auch häufig in den Texten auftreten, aus denen die Verläufe erstellt werden, wird den Themen, zu denen die Funktionswörter gehören, eine hohe Zentralität zugewiesen und sie erscheinen somit als wichtig. In Abbildung 6.4.1 wurden diese Funktionsthemen ausgeblendet, damit man den Verlauf des Themas über das Erdbeben besser erkennen kann. Dieser Verlauf unterscheidet sich vom Verlauf in Abbildung 6.3.2, da eine andere Framegröße benutzt wurde.

6.4 Zusammenfassung

Es wurden die Ergebnisse der entwickelten Methode und deren Anwendung auf verschiedene Korpora dargestellt. Im ersten Teil wurden Themenmodelle bewertet und die am besten bewerteten wurden für die spätere Analyse ausgewählt. Anhand der ausgewählten Modelle, wurden synthetische Dokumentströme erzeugt anhand derer Themenverläufe bestimmt wurden. Diese künstlichen Verläufe konnten darauf untersucht werden, wie gut sie mit den verschiedenen Zen-

tralitätsindizes in Kombination mit den Graphenalgorithmen erfasst werden. Dies ist möglich, da die Dokumentensequenz aus dem Themenmodell generiert wurden und somit das Thema und dessen Wahrscheinlichkeit innerhalb eines Dokuments bekannt sind. Anhand der Textströme konnte man vorhersagen, wie die Verläufe aussehen und somit konnte auch bewertet werden, wie gut sie durch die Zentralitäten erfasst werden.

Die Betweenness-Zentralität zusammen mit den Algorithmen DZG und GMT hat sich dabei als am besten geeignet erwiesen. Mit dieser Zentralität und dem Graphenalgorithmus GMT wurden für reale Dokumentströme aus dpa-Nachrichtenmeldungen und SCY-Chats Themenverläufe erstellt. Die dpa-Nachrichtenmeldungen enthielten ein bekanntes Ereignis, welches auch korrekt erkannt wurde. Für die SCY-Chatdaten war im Voraus kein einzelnes Ereignis bekannt, jedoch konnten die Diskussionsthemen verfolgt werden.

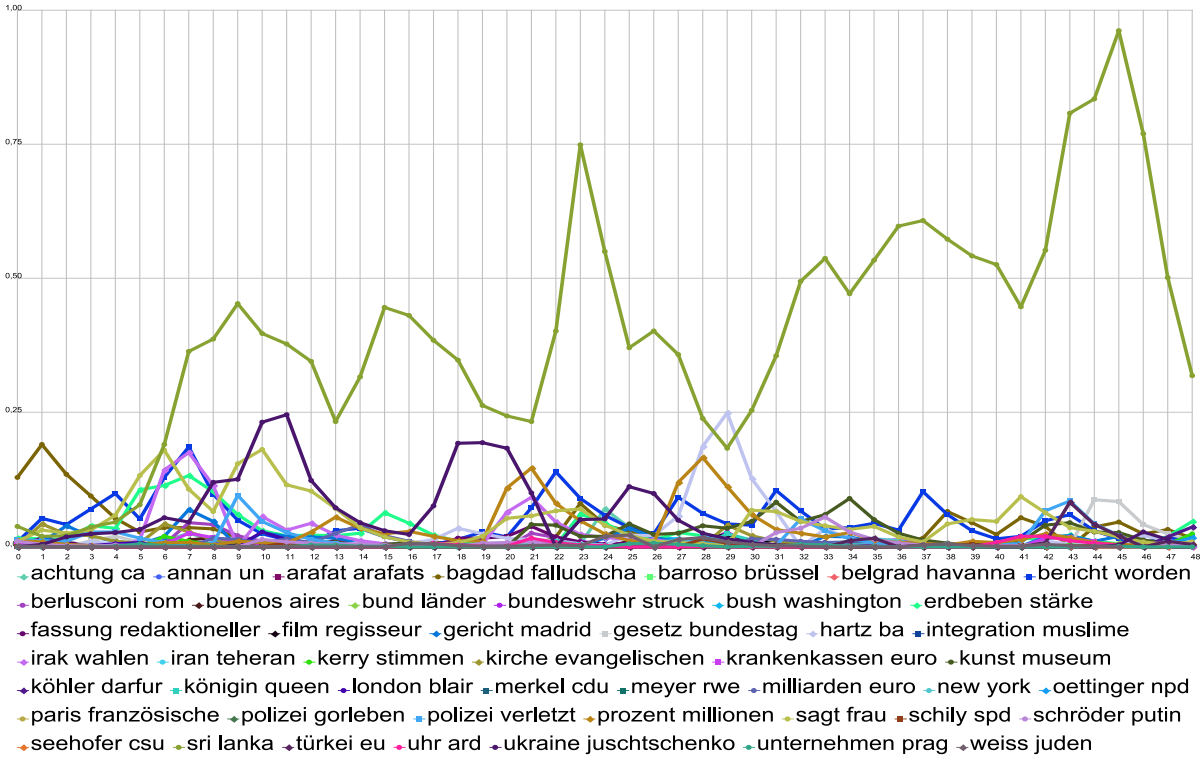


Abbildung 6.4.1: Themenverläufe der Betweenness-Zentralität für ein Themenmodell mit 50 Themen, ohne Stammformreduktion, mit Entfernung der Stopwörter. Die Framegröße ist 50 und die Überlappung 25. Die Verläufe wurden mit dem Algorithmus GMT erstellt.

7 Diskussion

In dieser Diplomarbeit wurde eine Methode entwickelt, um zeitliche Veränderungen in der Thematik von Texten zu erfassen. Dazu wurden Themenmodelle aus Hintergrundtexten trainiert, um die Themen der Texte erfassen zu können. Ausgehend von diesen Themenmodellen wurde die zeitliche Veränderung der Themen in Textsequenzen erfasst, indem diese zuerst in Zeitabschnitt aufgeteilt wurden. Anschließend wurde dann die Kookkurrenz der Themen der Dokumente, die in einem Zeitabschnitt enthalten sind, in einem Graphen kodiert. Auf jeden Graphen eines Zeitabschnittes wurde dann ein Zentralitätsmaß angewendet, um die Prominenz der Themen über die Zeit zu verfolgen.

Dazu mussten verschiedene Aufgaben gelöst und Algorithmen entwickelt werden. So musste eine Methode entwickelt werden, wie die Textsequenzen in Zeitabschnitte unterteilt werden können. Die Themenkookkurrenz der in diesen Zeitabschnitten vorkommenden Dokumente mussten als Graphen kodieren werden. Zu diesem Zweck wurden die drei Algorithmen aus Abschnitt 5.2 entwickelt. Für die Graphen mussten die verschiedenen Zentralitätsmaße implementiert werden. Schlussendlich musste eine geeignete Visualisierung der Themenverläufe gefunden und implementiert werden. Zusätzlich wurden die Verfahren zur Bestimmung der optimalen Themenanzahl und der Evaluation der Themenverläufe entwickelt, da es bisher keine geeigneten Verfahren gab, um die optimale Themenanzahl von Themenmodellen nur anhand der gelernten Modelle ohne Zurückhalten von Dokumenten zu bestimmen. Eine numerische Evaluation von Themenverläufen wurde bisher auch noch nicht durchgeführt, deshalb wurde das hier dargestellte Evaluationsverfahren entwickelt.

Mit der Entwicklung und Implementierung der Methoden und Algorithmen konnten verschieden Fragestellungen untersucht werden. Die zentrale zu lösende Fragestellung der Diplomarbeit war es, zu untersuchen, welche Zentralitätsmaße, auf die verschiedenen erstellten Graphen angewandt, die Themenveränderung am besten erfassen. Zusätzlich wurde der Einfluss verschiedener Parameter auf die Erfassung der Themenänderungen untersucht. Zum einen mussten in der Lernphase die Themenmodelle bestimmt werden, die das zugrundeliegende Korpus am besten repräsentieren, bzw. die optimale Anzahl von Themen musste bestimmt werden. Zum anderen wurden Themenmodelle mit unterschiedlichen Parametern trainiert und der Effekt dieser Parametervariation auf die Erfassung der Themenverläufe wurde untersucht. So wurden Modelle mit Stopwortentfernung und Stammformreduktion trainiert, Modelle, für die nur eine Stopwortentfernung oder Stammformreduktion durchgeführt wurde und ein Modell, welches weder Stopwortentfernung noch Stammformreduktion im Vorverarbeitungsschritt enthält. Die Größe der

Frames und die Größe der Überlappung spielt auch eine wesentliche Rolle bei der Erfassung der Themenverläufe. Diese wurden jedoch nur empirisch für die realen Textsequenzen ermittelt. Eine genauere Untersuchung zum Einfluss der Framegröße auf die Erfassung der Themenveränderung wurde aus Zeitgründen nicht mehr durchgeführt.

Die so entwickelte Methode, um die Veränderung in der Prominenz von Themen über die Zeit zu erfassen, zeigte gute Ergebnisse für die Betweenness-Zentralität auf Graphen die mit den Algorithmen DZG und GMT erstellt wurden. Die synthetisch erzeugten Verläufe und die realen Verläufe wurden korrekt erfasst und wiedergegeben. Die Variation der Parameter zum Training der Themenmodelle zeigte keinen signifikanten Einfluss auf die Qualität der Themenerfassung. Es hat sich jedoch gezeigt, dass die numerische Bewertung teilweise nicht mit der manuell visuellen Bewertung übereinstimmt. So wurden Verläufe schlecht bewertet, obwohl sie visuell klar erkennbar und gut zu unterscheiden waren. Ein weiterer Nachteil der hier entwickelten Methoden stellt die Wahl der Framegröße dar. Die Framegröße und die Überlappung werden fest gewählt und müssen je nach betrachteter Textsequenz manuell ermittelt werden, um Zeitabschnitte zu repräsentieren. Es wäre denkbar die Framegröße und Überlappung automatisch bestimmen zu lassen und die Framegröße variieren zu lassen.

Auch ist es nötig für die Bestimmung der optimalen Themenanzahl für die Themenmodelle viele Themenmodelle zu trainieren um die Bewertung zu vergleichen. Es kann kein absoluter Wert angegeben werden, ab dem ein Themenmodelle als geeignet betrachtet werden kann. Die optimale Anzahl von Themen ist immer nur im direkten Vergleich verschieden parametrisierter Modelle zu ermitteln. Es wäre zu untersuchen, ob die Anzahl der zu trainierenden Modelle durch Ziehen von Stichproben gesenkt werden kann.

Dies beschränkt sich jedoch auf die Lernphase, die offline durchgeführt werden kann. In der Anwendungsphase müssen die Modelle nicht aufwändig angepasst oder neu trainiert werden. Die entwickelte Methode kann somit als Online-Algorithmus verwendet werden, um kontinuierliche Textströme zu untersuchen. Weiterhin berücksichtigt die hier entwickelte Methode, im Gegensatz zu anderen Ansätzen wie zum Beispiel [15], die Wahrscheinlichkeit der Themen; einerseits implizit, wie im Algorithmus DZG beschrieben und andererseits explizit, wie im Algorithmus GMT.

7.1 Ausblick

Die hier entwickelte Methode lässt sich nicht nur dazu einsetzen, Themenverläufe in Nachrichtenmeldungen oder Chats von Schülern zu überwachen. Es wäre zusätzlich denkbar, die Überwachung auf bestimmte Themen einzuschränken, um das Aufkommen bestimmter Themen zu entdecken. Im Kontext des SCY-Projektes könnte der Fokus von Diskussionen von einem Lehrer gesteuert werden oder es wäre denkbar anhand der behandelten Themen einer Diskussion Rückschlüsse auf den Lernfortschritt der Schüler zu ziehen. Hält sich die Diskussion sehr

lange mit einem Thema auf, könnte dies darauf hinweisen, dass das Thema nicht richtig verstanden wurde und zusätzlicher Lernbedarf besteht. Andererseits kann über die Prominenz der Themen festgestellt werden, dass die Schüler ein Thema verstanden haben. Erscheint ein Thema am Anfang einer Diskussion sehr wenig und wird über die Zeit prominenter könnte man daraus schließen, dass die Schüler das Thema besser verstehen als am Anfang, da mehr Fachvokabular benutzt wird, so dass das Thema öfter als prominent gewertet wird. Ob sich die hier entwickelte Methode dazu eignet, solche Lernmuster zu entdecken muss jedoch noch untersucht werden.

Werden zusätzlich zu den Termen der Dokumente andere Merkmale der untersuchten Dokumente hinzugezogen, können auch andere Verläufe untersucht werden. So ist es möglich die syntaktische Struktur der untersuchten Dokumente als Merkmal mit einzubeziehen. Trainiert man ein Themenmodell mit der Syntax der Dokumente als Merkmal, könnten Themen entstehen, die bestimmte rhetorische Muster gruppieren. Anhand dieser rhetorischen Themen kann dann unter Umständen die Argumentationsstruktur einer Diskussion verfolgt werden. Dies gilt es aber noch zu untersuchen.

Zusätzlich wäre es interessant zu untersuchen, inwieweit die Nutzung von adaptiven Themenmodellen, wie sie in [16] oder [6] dargestellt wurden, die Erfassung von Themen in Nachrichtentexten verändert. Für die Anwendung im SCY-Projekt ist das statische Themenmodell ausreichend, da sich die Hintergrundtexte nicht verändern und die Schüler kein neues fachliches Vokabular entwickeln. Für die Anwendung in realen Szenarien wäre es zusätzlich denkbar, die Themen, die nur Funktionswörter gruppieren und keinen semantische Inhalt automatisch aufweisen, speziell zu kennzeichnen. So könnte man diese, wenn sie, wie im Falle der dpa-Nachrichtenmeldungen unerwünscht sind, ausblenden. Im Falle der SCY-Chattexte könnten die Funktionswortthemen speziell überwacht werden und festgestellt werden, wann die Schüler von der zu bearbeitenden Aufgabe abweichen und etwas anderes tun.

Literaturverzeichnis

[1] Alsumait, L. ; Barbará, D. ; Domeniconi, C.: On-line LDA: Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking. In: *IEEE International Conference on Data Mining* (2008), S. 3–12

[2] Anjewierden, A. ; Kollöffel, B. ; Hulshof, C.: Towards educational data mining: Using data mining methods for automated chat analysis to understand and support inquiry learning processes. In: *Proceedings of: International Workshop on Applying Data Mining in e-Learning, at the 2nd European Conference on Technology Enhanced Learning.*, (2007)

[3] Blei, D. M. ; Lafferty, J. D.: Dynamic topic models. In: *Proceedings of the 23rd International Conference on Machine Learning*, (2006)

[4] Blei, D. M. ; Ng, A. Y. ; Jordan, M. I.: Latent dirichlet allocation. In: *Journal of Machine Learning Research* 3 (2003), S. 993–1022

[5] Brandes, U. (Hrsg.) ; Erlebach, T. (Hrsg.): *Lecture Notes in Computer Science*. Bd. 3418: *Network Analysis*. Springer-Verlag, (2005)

[6] Canini, K. R. ; Shi, L. ; Griffiths, T. L.: Online Inference of Topics with Latent Dirichlet Allocation. In: *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, (2009)

[7] Chung, S. ; Mcleod, D.: Dynamic Topic Mining from News Stream Data. In: *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*. (2003) (Lecture Notes in Computer Science), S. 653–670

[8] Feng, D. ; Kim, J. ; Shaw, E. ; Hovy, E.: Towards Modeling Threaded Discussions through Ontology-based Analysis. In: *Proceedings of National Conference on Artificial Intelligence*, (2006)

[9] Glance, N. S. ; Hurst, M. ; Tomokiyo, T.: BlogPulse: Automated trend discovery for weblogs. In: *WWW 2004 Workshop on the Weblogging Ecosystem*, (2004)

[10] Griffiths, T. L. ; Steyvers, M.: Finding Scientific Topics. In: *Proceedings of the National Academy of Sciences of the United States of America* 101 (2004)

[11] Heinrich, G.: Parameter estimation for text analysis. / University of Leipzig. (2008). – Forschungsbericht

[12] Hofmann, T.: Probabilistic latent semantic indexing. In: *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, (1999), S. 50–57

[13] Kim, J. ; Shaw, E. ; Ravi, S. ; Tavano, E. ; Arromratana, A. ; Sarda, P.: Scaffolding On-line Discussions with Past Discussions: An Analysis and Pilot Study of PedaBot. In: *Intelligent Tutoring Systems* Bd. 5091. Springer Verlag, (2008)

[14] Kleinberg, J. M.: Authoritative sources in a hyperlinked environment. In: *Journal of the ACM* 46 (1999), Nr. 5, S. 604–632

[15] Linstead, E. ; Lopes, C. ; Baldi, P.: An Application of Latent Dirichlet Allocation to Analyzing Software Evolution. In: *Fourth International Conference on Machine Learning and Applications* (2008), S. 813–818

[16] McCallum, A. ; Wang, X.: Topics over Time: A Non-Markov Continuous-Time Model of Topical Trends. In: *Proceedings of the 12th Conference on Knowledge Discovery and Data Mining*, (2006)

[17] Mclaren, B. M. ; Scheuer, O. ; Laat, M. de ; Hever, R. ; Groot, R. de ; Rose, C. P.: Using Machine Learning Techniques to Analyze and Support Mediation of Student E-Discussions. In: *Proceedings of the 13th International Conference on Artificial Intelligence in Education*, (2007), S. 331–338

[18] Miksatko, J. ; Mclaren, B.: What's in a Cluster? Automatically Detecting Interesting Interactions in Student E-Discussions. In: *Proceedings of the 9th International Conference on Intelligent Tutoring Systems*, (2008), S. 333–342

[19] Page, L. ; Brin, S. ; Motwani, R. ; Winograd, T.: *The PageRank Citation Ranking: Bringing Order to the Web.* (1999)

[20] Poulin, R. ; Boily, M. C. ; MÂsse, B. R.: Dynamical systems to define centrality in social networks. In: *Social Networks* 22 (2000), Nr. 3, S. 187 – 220

[21] Scheuer, O. ; Mclaren, B. M.: Helping Teachers Handle the Flood of Data in Online Student Discussions. In: *Proceedings of the 9th International Conference on Intelligent Tutoring Systems*, (2008), S. 323–332

[22] Shaffer, D.W. ; Hatfield, D. ; Svarovsky, G. N. ; Nash, P. ; Nulty, A. ; Bagley, E. ; Franke, K. ; Rupp, A. A. ; Mislevy, R.: Epistemic Network Analysis: A prototype

for 21st Century assessment of learning. In: *The International Journal of Learning and Media (in press)* (2009)

[23] Snowball web site: *[http://snowball.sourceforge.net].* August (2009)

[24] Steyvers, M. ; Griffiths, T.: Probabilistic Topic Models. In: *Latent Semantic Analysis: A Road to Meaning.* Erlbaum, (2006)

[25] Tuulos, V. H. ; Tirri, H.: Combining Topic Models and Social Networks for Chat Data Mining. In: *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, (2004), S. 206–213

[26] Walsh, B.: Markov Chain Monte Carlo and Gibbs Sampling. (2004). – Forschungsbericht