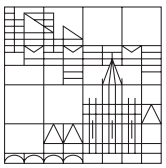

Universität Konstanz
Fachbereich Informatik und Informationswissenschaft

Bachelorarbeit

Netzwerkanalytische Methoden zur Text-Exploration

Network-Analytical Means of Text Exploration

zur Erlangung des akademischen Grades eines Bachelor of Science (B.Sc.)

Studiengang: Information Engineering
Themengebiet: Angewandte Informatik

von
Simon Endele
simon.endele@uni-konstanz.de
Matrikelnummer.: 01/551125

Erstgutachter: Prof. Dr. Ulrik Brandes
Zweitgutachter: Prof. Dr. Daniel A. Keim
Einreichung: März 2007

Zusammenfassung

Bei dem Verfahren *Centering Resonance Analysis* (CRA) von Corman et al. [CKMD02] wird aus einem natürlich sprachlichen Text ein Netzwerk erstellt, in dem die Knoten die Wörter oder Begriffe im Satz repräsentieren und die Kanten die Zusammenhänge dieser Begriffe im Text abbilden sollen. Auf dieses Netzwerk lassen sich netzwerkanalytische Methoden anwenden, um die Knoten bezüglich ihrer strukturellen Wichtigkeit im Netzwerk zu bewerten.

Am Beispiel von Nachrichtentexten, die nach den Terroranschlägen auf das World Trade Center in New York am 11. September 2001 veröffentlich wurden, sollen hier einige textexplorative Ansätze zur Verwendung solcher Netzwerke beschrieben und untersucht werden.

Abstract

The *Centering Resonance Analysis* (CRA) developed by Corman et al. [CKMD02] is an approach which describes how to build a network out of a natural language text. Nodes represent the words or terms and the links in the network shall express their coherence in the text. To get a stuctural rating of the nodes, network analytical means can be used.

Taking news texts as example that were published after the terrorist attack on the World Trade Center in New York on 11th of September 2001, some text explorative approaches for processing such networks will be described and analyzed here.

Inhaltsverzeichnis

1 Einleitung

1.1 Ziele einer Textanalyse

Nach [CKMD02] sind die drei großen Ziele der automatischen Textanalyse *inference*, *positioning* und *representation*.

Bei der *inference* wird versucht, aus einem Text Informationen abzuleiten, die dort nicht explizit gegeben sind. Beim *positioning* wird ein Text mit anderen verglichen, um ihn über die Ähnlichkeit mit einer Sammlung von Texten in einen Kontext, ein Fachgebiet einordnen zu können. Das erklärte Ziel der *representation* ist es, aus einem Text relevante Informationen zu extrahieren, um eine komprimierte Darstellung des Inhalts zu erzeugen.

Wir werden sehen, dass das hier vorgestellte Verfahren sowohl im Bereich *positioning* als auch *representation* Anwendung findet.

1.2 Abgrenzung von anderen Techniken

Als alternative Technik im Bereich *representation* sei an dieser Stelle nur beispielhaft *TF-IDF* (*term frequency–inverse document frequency*) [BR99] erwähnt. Mit diesem Maß kann ein Term, also ein Wort oder ein längerer Begriff, innerhalb eines Textes gewichtet werden, wobei der Term dabei in den Bezug einer repräsentativen Sammlung von Dokumenten gesetzt wird. Diejenigen Terme mit den höchsten Gewichten sollen schließlich eine stichwortartige Zusammenfassung des Textes ergeben.

Dabei wird die Auftrittswahrscheinlichkeit (*term frequency*) des zu untersuchenden Terms (Anzahl Vorkommen des Terms im Text dividiert durch die Gesamtvorkommen aller Terme im Text) mit der sogenannten *inversen Dokumenthäufigkeit* (*inverse document frequency*) multipliziert. Die inverse Dokumenthäufigkeit gibt dabei den Logarithmus vom Kehrwert des Anteils an Dokumenten in ei-

nem Textkorpus an, in denen das Wort vorkommt, und soll somit die globale Relevanz des Wortes wiedergeben.

Pseudo-formal lässt sich die TF-DIF für einen Term t eines Dokuments D wie folgt notieren:

$$tf\text{-}idf(t \in D) = \frac{\text{Häufigkeit}(t)}{\sum_{k \in D} \text{Häufigkeit}(k)} \cdot \log\left(\frac{\text{Anz. aller Dokumente}}{\text{Anz. Dokumente mit } t}\right)$$

Kommt ein Term in relativ wenigen Referenztexten vor, aber sehr häufig in dem untersuchten Text, wird seine Häufigkeit als überdurchschnittlich gewertet und der Term somit als relevant.

Der erste Nachteil dieser Technik, den wir hier festhalten wollen, ist die Abhängkeit von einem Korpus. Einerseits muss eine solche Textsammlung zunächst einmal gegeben sein, andererseits muss sie exakt den Kontext des untersuchten Dokuments wiedergeben, da sonst die Standardisierung durch die inverse Dokumenthäufigkeit fehlschlägt. Der zweite Nachteil besteht darin, dass die Analyse auf reiner Statistik beruht und in keiner Weise auf die Struktur des Textes eingegangen wird.

Da Messung und Berechnung jedoch leicht zu bewerkstelligen sind, ist TF-IDF trotz allem ein viel verwendetes Maß, vor allem im *Information Retrieval* und *Text Mining*.

1.3 Motivation Text-Netzwerke

Die elementare Neuerung in der Netzwerk-Textanalyse besteht darin, einen Zwischenschritt einzubauen, in dem aus dem Text ein Netzwerk konstruiert wird, wobei die Knoten des Netzwerks die Wörter im Text repräsentieren. Durch die komplexe Struktur eines Netzwerks lassen sich die innere Struktur des Textes und die Beziehungen zwischen den Termen exakt darstellen. Dieses mathematisch-logische Modell bildet die Grundlage für netzwerkanalytische Methoden, welche die Knoten im Netzwerk strukturell bewer-

ten. Wie wir in Kapitel 5 sehen werden, kann man diese Netzwerke für verschiedenste Zwecke nutzen.

In dieser Arbeit wird vereinfacht davon die Rede sein, dass Wörter durch Knoten repräsentiert werden. Stellvertretend sind damit jedoch auch längere Begriffe gemeint, die nach Ermessen des Anwenders ebenso als Knotenbasis gewählt werden können.

Wir wollen in dieser Arbeit die beiden Verfahren *Word-Network Analysis* (WNA) und *Centering Resonance Analysis* (CRA) näher untersuchen. Da der Vorteil von CRA gegenüber WNA stärker linguistisch begründet ist und das Verständnis detailliertere Kenntnisse über die genaue Funktionsweise dieser beiden Verfahren erfordert, werden hier erst einige grundlegende Definitionen festgelegt, bevor die genaue Ausführung und der Vergleich in Kapitel 2 folgen.

1.4 Grundlegende Definitionen[1]

Definition 1.1 (Graph)

Eine endliche Knotenmenge V und eine Kantenmenge $E \subseteq V \times V$ bilden gemeinsam einen (gerichteten) Graphen $G = (V, E)$. *G ist* vollständig, *falls $E = V \times V$, also alle möglichen Kanten existieren.*
Ein Graph $G = (V, E)$ mit $E \subseteq \{\{v, w\} : v, w \in V\}$ ist ungerichtet.

In den folgenden Definitionen schreiben wir für eine Kante nur (v, w), meinen aber auch analog $\{v, w\}$ in ungerichteten Graphen.

Definition 1.2 (Multigraph)

Wenn zu einem Graphen $G = (V, E)$ zusätzlich eine Vielfachheit $\#_G : E \to \mathbb{N}$ *der Kanten definiert ist, sprechen wir von einem* Multigraphen. *Die Kantenmenge E ist also eine Multimenge und $\#_G(e) = \#e$ die Vielfachheit der Kante e in G.*

[1] teilweise entnommen aus [MNa07]

Ein Graph $G = (V, E)$ *mit* $\forall e \in E : \#e = 1$ *wie in der obigen Definition nennt sich* gewöhnlich.
Wir vereinbaren weiter, $n = |V|$ *sei die Anzahl der Knoten und* $m = |E|$ *die Anzahl der Kanten in einem Graphen.*

Die folgenden Definitionen, speziell für Multigraphen, können analog auf gewöhnliche Graphen übertragen werden.

Definition 1.3 (Knotengrad)

Für einen Multigraphen $G = (V, E)$ *und einen Knoten* $v \in V$ *seien*

- $d_G^-(v) = d^-(v) = \sum_{(w,v)\in E} \#(w, v)$ Eingangsgrad,
- $d_G^+(v) = d^+(v) = \sum_{(v,w)\in E} \#(v, w)$ Ausgangsgrad,
- $d_G(v) = d(v) = d^-(v) + d^+(v)$ Knotengrad *oder kurz* Grad *von* v.

Definition 1.4 (Nachbarschaft)

Für zwei Knoten $x, v \in V$ *in einem Multigraphen* $G = (V, E)$ *sei*

- x adjazent *(benachbart) zu* v*, falls* $(x, v) \in E$,
- e inzident *zu* v *sowie* v *inzident zu* e*, falls œ* $e = (u, v) \in E$,
- e' inzident *zu* e*, falls œ* $e' = (u, v),\ e = (v, w) \in E$.

Definition 1.5 (Schleife)

Eine Kante (v, w) *mit* $v = w$ *nennen wir* Schleife. *Ein Multigraph, der keine Schleife enthält, heißt* schleifenfrei.

Definition 1.6 (Teilgraph)

Ein Multigraph $G = (V, E)$ enthält *einen Multigraphen* $G' = (V', E')$, *falls* $V' \subseteq V$ *und* $E' \subseteq \{(v, w) \in E : v, w \in V'\}$. *Wir nennen* G' *auch* Teilgraph *von* G *und schreiben* $G' \subseteq G$.

Definition 1.7 (Weg)

Ein (gerichteter) Weg *(auch:* Pfad*) der Länge* k *in einen Multigraphen*

$G = (V, E)$ ist ein Teilgraph $P = (V', E')$ mit $V' = \{v_1, \ldots v_{k+1}\} \subseteq V$ und $E' = \{(v_1, v_2), \ldots, (v_k, v_{k+1})\} \subseteq E'$, wobei $|E'| = k$ gefordert wird.

Definition 1.8 (Clique)

In einem ungerichteten Graphen $G = (V, E)$ heißt eine Knotenteilmenge $C \subseteq V$ Clique *(der Größe $|C|$ bzw. $|C|$-Clique), falls der von C knoteninduzierte Teilgraph $G[C]$ vollständig ist.*

Definition 1.9 (Netzwerk)

Unter der etwas unscharfen Definition eines Netzwerks *soll ein Gebilde verstanden werden, dessen zu Grunde liegende Struktur einen Graphen darstellt, das jedoch über zusätzliche Eigenschaften oder Erweiterungen verfügt, beispielsweise einer Interpretation der Knoten; z. B. Wörter im Falle von Text-Netzwerken.*

Die Begriffe Netzwerk und (Multi-)Graph werden teilweise äquivalent verwendet, weil damit meistens dasselbe Objekt in unterschiedlichen Abstraktionstiefen betrachtet wird.

Desweiteren werden im Zusammenhang von Text-Netzwerken Wörter und die sie repräsentierenden Knoten des Netzwerks äquivalent verwendet.

2 Erstellung von Text-Netzwerken

Zuerst soll hier erwähnt sein, dass in dieser Arbeit von einer Aufbereitung der Texte abstrahiert wird. Die spezielle Behandlung von Synonymen[2], Homographen[3], mehrwortigen Begriffen wie „*World Trade Center*“, unterschiedlichen Schreibweisen[4] oder Rechtschreibfehlern würde hier den Rahmen sprengen und ist auch nicht Ziel dieser Arbeit.[5] Hier soll viel mehr die Nutzung und Weiterverarbeitung der Netzwerke im Vordergrund stehen.

Als Beispiel zur Veranschaulichung der Vorgehensweisen wird uns folgender Satz durch den Prozess begleiten:

The two planes crashed into the towers of
the World Trade Center in New York.

Der Satz stammt aus einem der Texte, die in 5.1 genauer untersucht werden.

2.1 WNA-Netzwerke

2.1.1 Aufbau

Die *Word-Network Analysis* (WNA) wurde von James A. Danowski entwickelt [Dan82], [Dan93]. Dabei wird der Text als eine Kette von Wörtern aufgefasst, die jeweils paarweise miteinander in Verbindung stehen, wenn sie innerhalb eines bestimmten Abstandes zueinander im Text vorkommen. Bei diesem Abstand werden allerdings irrelevante Wörter ignoriert.

Demnach findet ein Vorverarbeitungsschritt statt, in dem die Wörter im Text gefiltert werden, sodass nur aussagekräftige Be-

[2] Wörter mit gleicher Bedeutung aber unterschiedlicher Schreibweise

[3] Wörter mit gleicher Schreibweise aber unterschiedlicher Bedeutung

[4] z. B. amerikanisches/britisches Englisch oder Dialekt

[5] mit diesen Fällen beschäftigt sich [Blu06]

griffe in das Netzwerk übernommen werden. Dies kann am besten durch die Anwendung einer Stoppwortliste[6] und/oder durch Selektion von Wörtern bestimmter Wortarten realisiert werden. Das heißt, Wörter wie Artikel, Konjunktionen und Pronomen werden nicht in das Netzwerk übernommen. Zur Vorverarbeitung gehört auch das *Stemming*: Jedes Wort sollte in seine Grundform gebracht werden, damit alle flektierten[7] Vorkommen eines Wortes – beispielsweise *plane* und *planes* – mit dem selben Knoten identifiziert werden.

Nun wird über die Kette vorverarbeiteter Wörter ein Fenster einer vom Anwender festzulegenden Größe über den Text geschoben und sukzessive alle Wörter innerhalb des Festers mit dem ersten verbunden. Abbildung 1 zeigt unseren Beispielsatz nach der Vorverarbeitung und den hergestellten Verbindungen im ersten Schritt mit einer Fenstergröße von fünf.

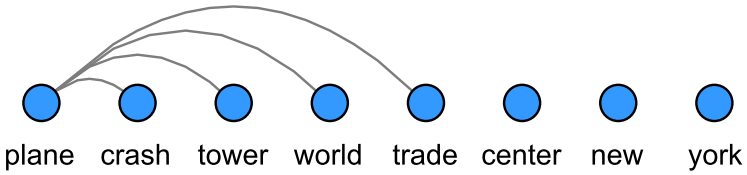

Abbildung 1: Erster Schritt zu einem WNA-Netzwerk mit Fenstergröße 5

Darin ist erkennbar, dass zwischen zwei Wörtern genau dann eine Kante existiert, falls der Abstand dieser Wörter im vorverarbeiteten Text kleiner oder gleich fünf beträgt. Danowski macht keine Aussage darüber, ob die Kanten gerichtet sein müssen. Wir definieren jedoch alle Kanten als ungerichtet mit der Begründung, dass

[6] Ausfilterung häufiger Wörter wie *by, but, done, every, have, how, it etc.*

[7] durch grammatikalische Beugung (Flexion) entstanden, d. h. durch Deklination bei Substantiven, Konjugation bei Verben und Komparation (Steigerung) bei Adjektiven und Partizipien

eine Richtung des Bezugs zweier Wörtern nicht als sinnvoll angesehen wird. Ein Wort soll zu sich selbst nicht in Verbindung stehen, auch wenn dies durch die Aufbauvorschrift theoretisch möglich ist. Da zwei Wörter im (vorverarbeiteten) Text mehrfach mit einem Abstand, der maximal die Fenstergröße beträgt, auftreten können, sind auch Multikanten möglich. Wir erhalten somit als WNA-Netzwerk einen schleifenfreien, ungerichteten Multigraphen. Abbildung 2 zeigt ein fertiges WNA-Netzwerk unseres Beispielsatzes, allerdings mit Fenstergröße drei.

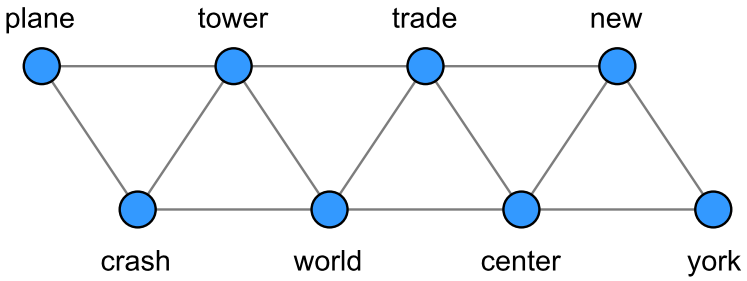

Abbildung 2: WNA-Netzwerk mit Fenstergröße 3

Hier könnte der Eindruck entstehen, das Netzwerk bilde für jeden Text nur einen langen „Schlauch“, dessen Dicke von der Wahl der Fenstergröße abhängt. Dabei muss allerdings beachtet werden, dass weitere Vorkommen eines Wortes – durch Stemming auch in flektierter Form – mit demselben Knoten identifiziert werden und somit für einen längeren Text ein enges Geflecht entsteht. Das gleiche gilt auch für CRA-Netzwerke, die wir in 2.2 betrachten werden.

Begründed wird der hier dargestellte Ansatz damit, dass alle Begriffe, die innerhalb eines bestimmten Abstandes – eben der gewählten Fentergröße – im Text auftreten, in einem Sinnzusammenhang zueinander stehen. Die Vielfachheit einer Kante gibt wieder, wie oft jeweils Vorkommen der beiden Begriffe, deren Knoten inzident zueinander sind, in einem Abstand, der höchstens die Größe des Fensters betragen darf, im Text auftreten. Hierbei ist wieder zu

beachten, dass diese Begriffe gestemmt sind und sich der Abstand auf den vorverarbeiteten Text bezieht, in dem einige Wörter fehlen.

2.1.2 Nachteile

Die Kritik an diesem Verfahren setzt genau an dieser Begründung an: Die Verwendung eines Fensters mit einer statischen Größe lässt sich linguistisch nicht begründen. Welche Fenstergröße für einen Text geeignet ist, hängt stark vom Aufbau und Genre des Textes, Formulierungsstil und Intention des Autors ab. Es lässt sich also keine allgemeingültige Parameterbelegung finden. Was jedoch noch entscheidender ist: Die erforderte Größe kann sogar innerhalb eines Textes variieren.

2.2 CRA-Netzwerke

2.2.1 Motivation

Im Hinblick auf die soeben erwähnten Nachteile der WNA sollte ein Verfahren entwickelt werden, das folgende drei Kriterien erfüllt:

- Es soll auf einem Netzwerk basieren, um die komplexe Struktur von Texten widerspiegeln zu können.
- Die Regeln zum Aufbau des Netzwerks sollen theoretisch begründet sein und auf die Struktur des Textes eingehen.
- Es soll flexibel und kontextübergreifend sein, d. h. die Unabhängigkeit von Wörterbüchern, Korpora oder Textsammlungen soll es ermöglichen, verschiedene Gruppen, Personen, Zeitspannen und Kontexte miteinander zu vergleichen.

Die *Centering Resonance Analysis* (CRA), die von Corman et al. [CKMD02] entwickelt wurde, basiert auf einer linguistischen Kohärenztheorie, die den logischen Aufbau und Zusammenhalt natürlich sprachlicher Texte beschreibt:

Die *Centering Theory* [GWJ95], [WJP98] besagt, dass ein Sprecher oder Autor Äußerungen bildet, deren Aussagen sich auf sogenannte *centers* konzentrieren. Diese *centers* sind *Nominalphrasen* (NPs), welche aus einem Substantiv (oder einem stellvertretenden Pronomen) bestehen, das durch Artikel, Adjektive oder sogar weitere Nominalphrasen oder Präpositionalphrasen erweitert sein kann. Sie bilden das Subjekt und die Objekte einer Äußerung und werden durch Verben, Pronomen, Artikel etc. aneinandergekettet. In der Regel werden dadurch Entitäten wie Dinge, Ereignisse oder Personen dargestellt [GGG93].

Logisch verknüpft werden die einzelnen Äußerungen (oder Sätze) über sogenannte „*backward-looking centers*“. In einem geschriebenen Text beispielsweise besitzt jeder Satz – abgesehen von dem ersten – ein solches, das sich auf ein „*forward-looking center*“ in einer vorhergehenden Äußerung bezieht. Verben, beziehungsweise ganze *Verbalphrasen*, bilden dagegen keine *Centers*, da sie in dieser Theorie die „Aktion“ darstellen, die die Nominalphrasen miteinander verknüpfen.

In der CRA wird nun versucht, diesen Aufbau der natürlichen Sprache nachzubilden und somit die Beziehung von Begriffen nicht durch ihren Abstand zueinander zu definieren, sondern durch die zu Grunde liegende grammatikalische Struktur des Textes.

2.2.2 Aufbau

Die Erstellung eines CRA-Netzwerkes besteht im Grunde aus drei Schritten: Zuerst muss der Text in Sätze und Wörter zerlegt sowie die Struktur der einzelnen Sätze analysiert und die Wortarten bestimmt werden. Anhand der Wortarten werden gewisse Wörter ausgewählt und mit einem Knoten identifiziert. Im letzten Schritt werden diese nach Regeln, die aus dem oben erläuterten theoretisch-sprachlichen Aufbau eines Textes abgeleitet sind, durch ent-

sprechende Kanten verbunden.

Die Struktur der einzelnen Sätze wird mit Hilfe eines natürlich sprachlichen Parsers analysiert. Nähere technische Details dazu werden in 4.2 gegeben. An dieser Stelle soll genügen, dass ein solcher Parser einen sogenannten „*Syntaxbaum*“ (oder *Ableitungsbaum*, *parse tree*) ausgibt. Abbildung 3 zeigt den entsprechenden Syntaxbaum für unseren Beispielsatz.

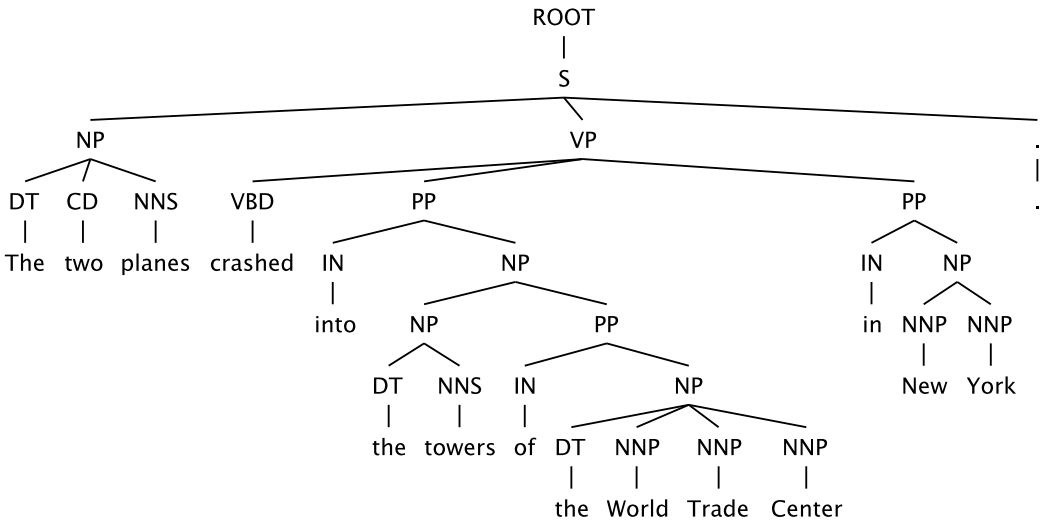

Abbildung 3: Ausgabe eines Parsers: Syntaxbaum

Ein Syntaxbaum stellt vereinfacht gesagt eine rekursive Unterteilung eines Satzes nach grammatikalischen Regeln dar. Die Wurzel des Baumes ist der komplette Satz. Im Beispiel wurde darauf eine Ableitungsregel angewendet, die besagt, dass ein Satz aus einer Nominalphrase (NP), einer Verbalphrase (VP) und einem Punkt besteht. Die Verbalphrase wird wiederum in ein Verb im Präteritum (VBD) und zwei Präpositionalphrasen (PP) unterteilt und so weiter.

Die Blätter repräsentieren schließlich die einzelnen Wörter des Satzes. Der einzige direkte Elternknoten eines jeden Blattes ist das sogenannte „*tag*“ des Wortes, eine Annotation, die die Wortart

des Wortes beschreibt. Die hier verwendeten Abkürzungen sind im *Penn Treebank Tag Set* definiert, wie es beispielsweise in [MSM93] beschrieben wird.

Tag	**Abkürzung (engl.)**	**Beschreibung (deutsch)**
FW	foreign word	Fremdwort
JJ	adjective	Adjektiv im Positiv
JJR	adjective, comparative	Adjektiv im Komparativ
JJS	adjective, superlative	Adjektiv im Superlativ
NN	noun, singular or mass	Substantiv, singular
NNS	noun, plural	Substantiv, plural
NNP	proper noun, singular	Eigenname, singular
NNPS	proper noun, plural	Eigenname, plural

Tabelle 1: Tags für CRA-Netzwerk

Diese *tags* werden nun im zweiten Schritt verwendet, um – ähnlich wie bei der WNA – Wörter anhand ihrer Wortart zu selektieren. Da Verben – wie in 2.2.1 begründet wurde – ebenso wie aussagelose Wörter wie Artikel, Konjunktionen und dergleichen entfernt werden, ergeben sich die *tags*, wie sie in Tabelle 1 aufgelistet sind, als Filter für selektierte Wörter.

Das Wort „*two*“ in unserem Beispielsatz wird durch den Wortartfilter herausgefiltert, weil es ein Zahlwort (CD = *cardinal number*) und damit im eigentlichen Sinne kein Adjektiv ist.

Die selektierten Wörter werden gestemmt und jeweils mit einem Knoten identifiziert.

Der letzte Schritt, das Einfügen der Kanten, besteht wieder aus zwei Unterschritten: Da eine Nominalphrase ein *center* und damit eine in sich abgeschlossene logische Einheit bildet, werden nun alle Knoten innerhalb einer Nominalphrase zu einer Clique ver-

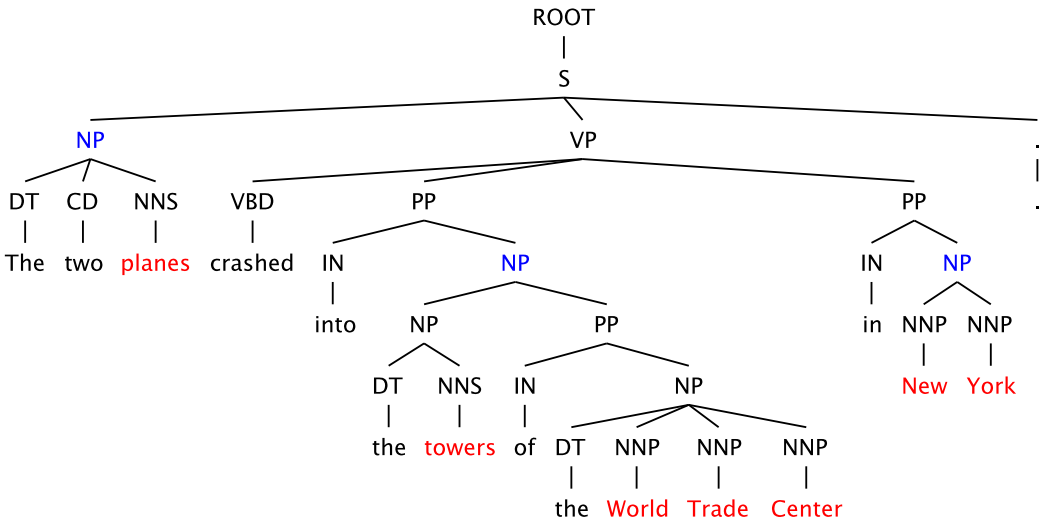

Abbildung 4: Syntaxbaum mit angewandtem Tag-Filter und identifizierten Nominalphrasen

bunden (Abbildung 5). Die Aneinanderkettung der Nominalphrasen wird dadurch erreicht, dass der jeweils letzte Knoten einer NP mit dem ersten Knoten der nächsten verbunden wird.

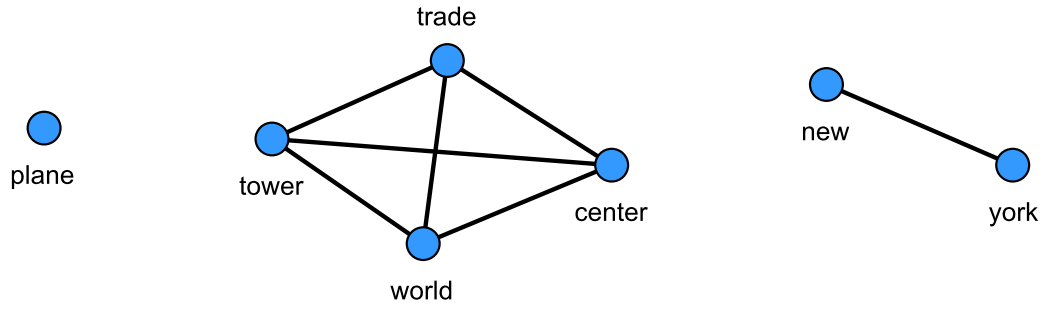

Abbildung 5: NPs zu Cliquen verbunden

Offen gelassen wird hierbei allerdings, auf welcher Ebene die Nominalphrasen ausgewählt werden. Wie Abbildung 4 zeigt, können diese nämlich hierarchisch ineinander geschachtelt auftreten.

Da zwei Wörter in einem Text mehrfach innerhalb einer Nominalphrase auftreten können, sind auch hier Multikanten mög-

lich. Entsprechend dem Argument bei den WNA-Netzwerken soll erneut ein Wort zu sich selbst nicht in Verbindung stehen können. Alle Kanten sind per Definition ungerichtet, sodass wir wieder einen schleifenfreien, ungerichteten Multigraphen erhalten, wie er für unseren Beispielsatz in Abbildung 6 gezeigt wird.

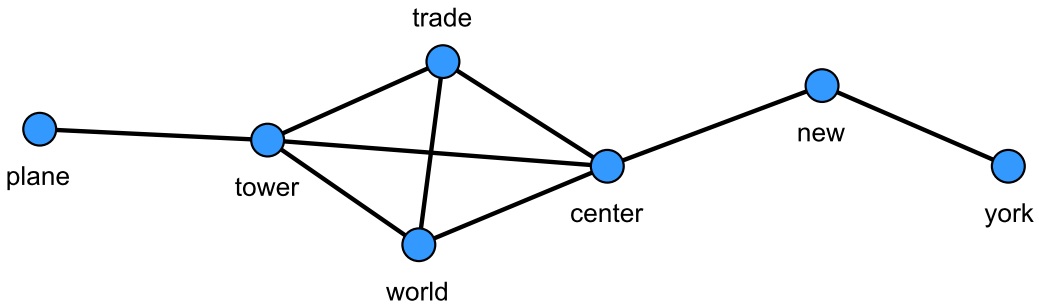

Abbildung 6: Fertiges CRA-Netzwerk mit verbundenen Cliquen

3 Analyse eines Text-Netzwerks

In diesem Kapitel wollen wir ausschließlich den zugrundeliegenden Multigraphen eines einzelnen Netzwerks betrachten. Die ursprüngliche Struktur des Textes und die Bedeutung der Knoten und Kanten wird hier ausgeblendet.

Der Grund dafür ist, dass sich die Netzwerktheorie schon seit Jahrzehnten mit der Frage beschäftigt, wie man die Knoten, allein auf der Struktur des Netzwerks beruhend, bewerten kann und welcher Knoten der zentralste oder wichtigste in einem Netzwerk.

Trotz dass eine solche Bewertung zweifellos subjektiv und abhängig von Kontext und Interpretation des Graphen ist, gibt es eine große Anzahl von Maßen, die die strukturelle Wichtigkeit der Knoten mit unterschiedlichen Kriterien beurteilen. Ferner existieren entsprechende Algorithmen, die diese Maße auch für sehr große Graphen schnell berechnen können.

Für eine konkrete Anwendung kann aus diesem Fundus geschöpft werden. Eine kleine Auswahl sogenannter *Zentralitäten* werden nun hier vorgestellt.

3.1 Zentralitäten[8]

Zunächst wollen wir die Grundform einer Zentralität definieren.

Definition 3.1 (Zentralität)

Eine (Knoten-)Zentralität *ist eine Abbildung* c*, die einen Graphen* $G = (V, E)$ *mit* $n = |V|$ *auf einen Vektor* $c(G) \in \mathbb{R}^n_{\geq 0}$ *abbildet.*
Der Eintrag $c(G)_v$ *im Vektor steht also für den* Zentralitätswert *oder auch kurz die* Zentralität *des Knotens* v *in* G*.*
Eine Zentralität c *ist* normiert*, falls* $\sum_{v \in V} c(G)_v = 1$*, d. h. falls die*

[8] die meisten Definitionen sind entnommen aus [MNa07]

Summe der Zentralitätswerte aller Knoten in G eins ergibt.

Je größer $c(G)_v$ ist, desto wichtiger oder zentraler wird ein Knoten v in G durch die Zentralität c eingestuft. Ein Knoten kann hier per Definition keine negative Wichtigkeit haben.

Die wohl einfachste Knoten-Zentralität ist folgende.

Definition 3.2 (Grad-Zentralität)
Die Grad-Zentralität c_D *ist definiert durch*

$$c_D(G)_v = d_G(v)$$

für alle Multigraphen $G = (V, E)$.

Das Problem bei diesem Zentralitäts-Maß ist, dass es nur die lokale Wichtigkeit eines Knotens bestimmt. So kann ein Knoten, der am Rand eines Netzwerks liegt, aber sehr viele inzidente Kanten besitzt, sehr hoch bewertet werden, obwohl er global gesehen keine große Rolle im Netzwerk spielt.

Eine Möglichkeit, ein umfassenderes Maß zu erhalten, ist die Einbeziehung von Abständen im Graphen.

Definition 3.3 (Abstand)
Für einen Multigraphen $G = (V, E)$ sei der (graphentheoretische) Abstand *oder die* Distanz $d_G(s, t) = d(s, t)$ *von einem Knoten $s \in V$ zu einem anderen Knoten $t \in V$ definiert durch die Länge des kürzesten Weges von s nach t.*

Beispielsweise haben die Knoten *york* und *tower* in unserem CRA-Netzwerk in Abbildung 6 den Abstand drei.

Eine Idee ist, an einem Knoten die Abstände zu allen anderen aufzuaddieren. Ein Knoten ist umso weniger zentral, je höher diese Summe ist. Durch Kehrwertbildung der Summe erhalten wir folgendes Zentralitäts-Maß.

3.1.1 Closeness

Definition 3.4 (Closeness)[9]

Die Closeness-Zentralität c_C *ist definiert durch*

$$c_C(G)_v = \frac{1}{\sum_{t \in V} d(v,t)}$$

für alle Multigraphen $G = (V, E)$*, wobei* $\frac{1}{0} = 1$ *gelte.*

Durch eine Breitensuche von jedem Knoten aus kann die Closeness-Zentralität sehr effizient in $\mathcal{O}(nm)$ Zeit und mit $\mathcal{O}(n + m)$ Speicher berechnet werden.

Eine weitere Möglichkeit ist, die Wichtigkeit eines Knotens über seine Rolle als „Vermittler“ zu definieren. Zwischen je zwei Knoten fließe ein Fluss durch den Graphen. Dies könnte beispielsweise ein Datenstrom in einem Computer-Netzwerk oder die Verhältnisse zwischen Personen in einem Beziehungs-Netzwerk sein. Ein Knoten kann unter dieser Auffassung des Netzwerks nun als zentral angesehen werden, wenn er auf möglichst vielen Verbindungswegen zwischen anderen Knoten platziert ist und somit einen Vermittler für viele Knotenpaare darstellt.

Die daran anknüpfende Annahme, dass dieser Fluss gezielt nur über die kürzesten Wege zwischen zwei Knoten verläuft, führt uns zu folgender Zentralität.

[9] nach [Bea65]

3.1.2 Betweenness

Definition 3.5 (Betweenness)[10]

Die (Shortest-Path) Betweenness-Zentralität c_B *ist definiert durch*

$$c_B(G)_v = \sum_{s,t \in V} \frac{\sigma_G(s,t|v)}{\sigma_G(s,t)}$$

für alle Graphen $G = (V, E)$*. Dabei bezeichne* $\sigma_G(s,t)$ *die Anzahl der kürzesten Wege von* s *nach* t*,* $\sigma_G(s,t|v)$ *die Anzahl der kürzesten* (s,t)*-Wege, die* v *als inneren Knoten enthalten (d. h.* v *liegt auf dem Weg, aber* $v \neq s,t$*), und es gelte* $\frac{0}{0} = 0$*.*

Zum Berechnen der Betweenness-Zentralität gibt es heute ebenfalls einen sehr effizienten Algorithmus [Bra01], ähnlich wie der Closeness-Algorithmus auf Breitensuchen basiert und nur $\mathcal{O}(nm)$ Zeit und $\mathcal{O}(n+m)$ Speicher benötigt.

Abbildung 7 zeigt unser Beispielnetzwerk mit angeschriebenen, prozentualen Betweenness-Werten. Wie intuitiv erwartet, erhält der Knoten *center* treffenderweise den höchsten Betweenness-Wert, da ausschließlich über ihn alle kürzesten Wege von der linken Seite des Graphen zur rechten verlaufen (und umgekehrt).

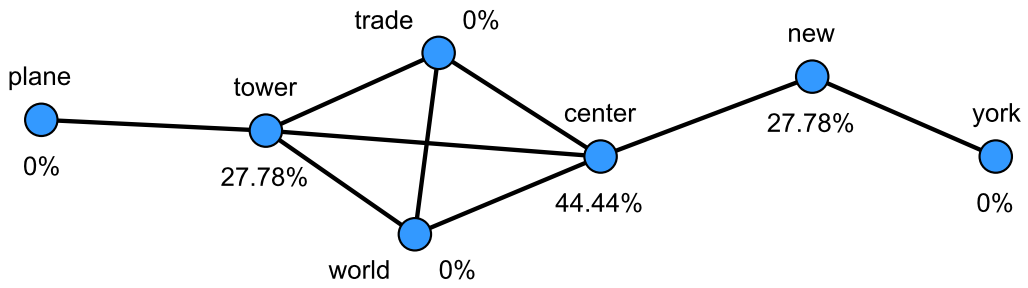

Abbildung 7: CRA-Netzwerk mit Betweenness-Werten

Wie man hier erkennen kann, gehen zum einen alle Knoten mit

[10] nach [Ant71], [Fre77]

Grad 1 (*plane* und *york*) leer aus. Zum anderen erhalten die Knoten *world* und *trade* ebenfalls keine Betweenness, da sie zwar einen relativ hohen Grad haben, jedoch auf keinem kürzesten Weg zwischen zwei Knoten liegen, da die Kante zwischen *tower* und *center* immer eine Abkürzung bildet. Existierte diese Kante nicht, hätten wir ein völlig anderes Ergebnis. Der Grund für diese Diskrepanz liegt darin, dass einzig und allein kürzeste Wege betrachtet werden.

Mit der Behebung dieses Mankos, befasst sich folgende Definition: Bei der *Current-Flow Betweenness* [BF05] wird ein Graph $G = (V, E)$ als ein Netzwerk aufgefasst, in dem elektrischer Strom fließt. Es wird sozusagen an alle Knoten paarweise eine Spannung angelegt, d. h. ein Knoten ist die Quelle (*source*) s, von dem aus Strom zum Ziel (*target*) t fließt. Die Kanten im Netzwerk stellen in dieser Analogie elektrische Leitungen dar, die einen gewissen Widerstand besitzen. Die Vielfachheit einer Kante kann als Querschnittsfläche des Leiters interpretiert werden, der sich umgekehrt proportional zum Widerstand verhält.

Der Fluss im Netzwerk verläuft nun nicht mehr lediglich über die kürzesten Wege zwischen zwei Knoten, sondern auch über Umwege mit entsprechend vermindertem Volumen.

Zur Berechnung der Current-Flow Betweenness gibt es einem Algorithmus, der eine Laufzeitkomplexität von $\mathcal{O}(mn \log n + n^\omega)$ hat, wobei ω der Exponent der Matrixmultiplikation ist (momentan $\omega < 2.376$). Die Implementierung des Algorithmus ist jedoch nicht trivial.

3.2 Wahl der geeigneten Zentralität

Die Grad-Zentralität ist für unsere Anwendung wie erwähnt ungeeignet, da sie für die Bewertung der Wichtigkeit eines Knotens im Bezug auf das gesamte Netzwerk aufgrund mangelnder Globalität nicht taugt.

Die Closeness hat den entscheidenden technischen Nachteil, dass sie nur auf *zusammenhängenden* Graphen definiert ist, das heißt, dass jeder Knoten von jedem anderen über einen Weg im Graphen erreichbar sein muss. Während diese Voraussetzung bei WNA-Netzwerken gegeben ist, gilt dies für CRA-Netzwerke in der Regel jedoch nicht. Denn schon ein einzelner Satz, der ausschließlich (gefilterte) Wörter beinhaltet, die in keinem anderen Satz im Text vorkommen, erzeugt einen losgelösten Teilgraphen. Neben dieser technischen Einschränkung für unsere Anwendung, wird in in [CKMD02] argumentiert, dass speziell Knoten, die zwei Teilgraphen miteinander verbinden, unterbewertet und Knoten in großen Clustern meist überbewertet werden.

Nach [CKMD02] stellt die (Shortest-Path) Betweenness die günstigste Wahl einer Zentralität für CRA-Netzwerke dar, da sie am besten misst, wie stark ein Knoten die Information, die durch das Netzwerk fließt, („*rush of meaning*“) kanalisiert.

Die Current-Flow Betweenness hat sich trotz der erwähnten Vorteile (noch) nicht durchgesetzt. Dies könnte an der aufwändigen Implementation oder aber daran liegen, dass das Verfahren noch neu ist. Zum anderen besitzt diese Zentralität wie die Closeness den technischen Nachteil, dass sie nur auf *zusammenhängenden* Graphen definiert ist.

Definitiv entschieden kann die beste Wahl der geeigneten Zentralität hier nicht, da sie zu stark in den linguistischen Bereich ragt. Schlussendlich kommt es auch auf den Anwendungskontext an und welche Eigenschaft des Text-Netzwerks untersucht werden soll.

4 Implementation

4.1 Visone

Entwickelt wurde das Tool zur Erstellung von Text-Netzwerken, das in dieser Bachelorarbeit vorgestellt wird, als ein Teil von *Visone* [Vis07], ein Langzeit-Forschungsprojekt zur Analyse und Visualisierung von sozialen Netzwerken. Visone basiert auf der Java-Graphenbibliothek *yFiles* der Firma *yWorks* [yWo07] und stellt bereits einige sinnvolle Funktionen zur Verfügung wie die visuelle Darstellung und Möglichkeit zur manuellen Modifikation von Netzwerken oder die Berechnung von etlichen Zentralitäts-Maßen und vieles mehr.

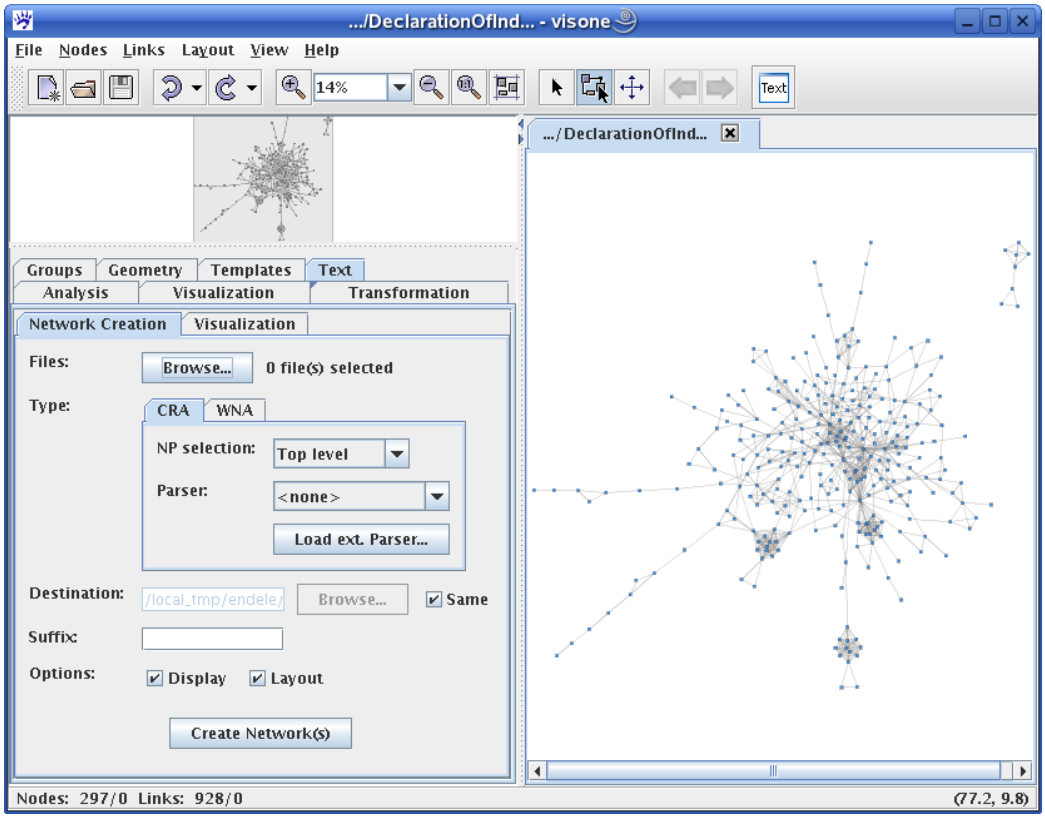

Abbildung 8: Screenshot von Visone mit geöffnetem Text-Tab

Desweiteren ist es möglich, die Software durch Module zu erweitern, für welche Steuerelemente in einem Tab[11] und Buttons in der Toolbar platziert werden können und die Zugriff auf die geöffneten Netzwerke haben.

Abbildung 8 zeigt das Visone-Fenster mit dem geöffnetem Tab „Text“ auf der linken Seite, dessen Inhalt im Rahmen dieser Arbeit implementiert wurde.

4.2 Parser und Tagger

Wie in Kapitel 2 dargelegt wurde, wird für die Erstellung eines WNA- oder CRA-Netzwerks ein Programm zur Erkennung von Wortarten, beziehungsweise ein natürlich sprachlicher Parser benötigt. Diese Funktionalitäten bieten der *Part-of-Speech Tagger (POS-Tagger)* und der darauf aufbauende *Natural Language Parser*, die von der *Stanford Natural Language Processing Group* entwickelt wurden. Dabei handelt es sich um Open Source Java-Bibliotheken, die unter der *GNU General Public License* [GPL07] stehen und auf der Website der Entwickler [Sta07] unter *software* kostenlos heruntergeladen werden können.

Neben der Wortarterkennung und dem Parsen natürlicher Sprache, von deren genauer Funktionalität in dieser Arbeit abstrahiert werden soll, sind in den Software-Paketen einige nützliche Tools enthalten, die im Kapitel 2 schon als gegeben vorausgesetzt wurden: Die Klasse *DocumentPreprocessor* nimmt uns die undankbare Arbeit ab, einen Text in Sätze und Wörter zu unterteilen. Dies ist keine so leichte Aufgabe, da Abkürzungen wie „*U. S.*“, direkte Rede, Aufzählungen und Ähnliches dabei Probleme bereiten können. Außerdem ist ein *Stemmer* vorhanden, der Wörter auf ihre grammatikalische Grundform reduziert.

[11] Registerkarte

4.3 Aufbau der Implementation

4.3.1 Klassenbeschreibungen

Für die Text-Netzwerk-Erstellung wurde ein *Package* innerhalb von Visone angelegt mit dem Namen `nlp`, der für *Natural Language Processing* stehen soll. Das UML-Diagramm in Abbildung 9 zeigt ein einen Überblick über die elementaren Klassen im Package.

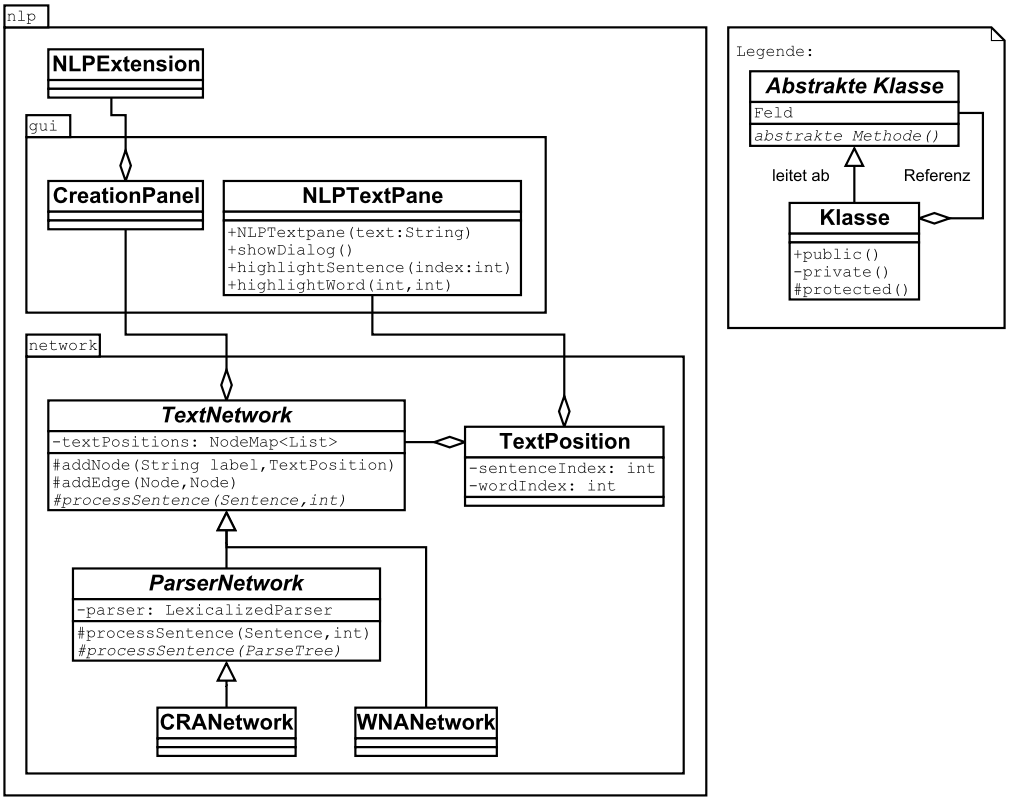

Abbildung 9: UML-Diagramm

Die Klassen besitzen folgende Funktionalitäten:

`NLPExtension` und **`CreationPanel`** bilden die Schnittstelle zu Visone und zum Benutzer durch grafische Steuerelemente.

`TextNetwork` repräsentiert ein abstraktes Text-Netzwerk, das heißt,

sie verwaltet Attribute wie den Text, aus dem das Netzwerk aufgebaut wurde, die Abbildung der Knoten auf Wörter oder Begriffe und umgekehrt, sowie die Positionen im Text, an denen ein Wort vorkommt, in Form von Instanzen der Klasse **`TextPosition`**. Über die Methode `addNode(String,[..])` wird dabei die nötige Abstraktionsebene geschaffen. Sie gibt zu einem Wort den repräsentierenden Knoten zurück, nachdem er nötigenfalls erstellt wurde. Der detaillierte Aufbau des Netzwerks, also das Erstellen von Kanten, wird jedoch der ableitenden Klasse überlassen, die die abstrakte Methode `processSentence(Sentence,[..])` implementieren muss.

`ParserNetwork` erweitert `TextNetwork` durch die Funktionalität des Parsers. Die Methode `processSentence(Sentence,[..])` wird implementiert, indem der Satz geparst und der Syntaxbaum der Methode `processSentence(Tree)` übergeben wird, die wiederum abstrakt ist.

`CRANetwork` ist von `ParserNetwork` abgeleitet und baut mit Hilfe des Syntaxbaums ein CRA-Netzwerk auf. Dabei kann eine der beiden NP-Auswahlstrategien *top level* und *lowest level* gewählt werden, wobei *top level* Standard ist.

`WNANetwork` ist von `TextNetwork` abgeleitet und baut mit Hilfe des POS-Taggers und einer Stoppwortliste (im Package enthalten unter `/nlp/resource/stop_words.txt`) ein WNA-Netzwerk auf. Im Kontruktor ist die zu verwendende Fenstergröße festzulegen; Standardwert ist fünf.

4.3.2 Text-Rückverfolgung

Die Klasse **`NLPTextPane`** bietet die Funktionalität der Text-Rückverfolgung. Durch Anklicken des Text-Buttons in der Toolbar wird ein Dialogfenster geöffnet, das den Text anzeigt, aus dem das Netz-

werk erstellt wurde. Mit Hilfe der abgespeicherten Textstellen ist es möglich, durch Selektieren eines Knotens alle Vorkommen im Text zu markieren, vorausgesetzt die entsprechende CheckBox ist angewählt. Bei Selektion eines Satzes im Dialogfenster wiederum werden die entsprechenden Knoten der Wörter in diesem Satz im Graphen farblich hervorgehoben.

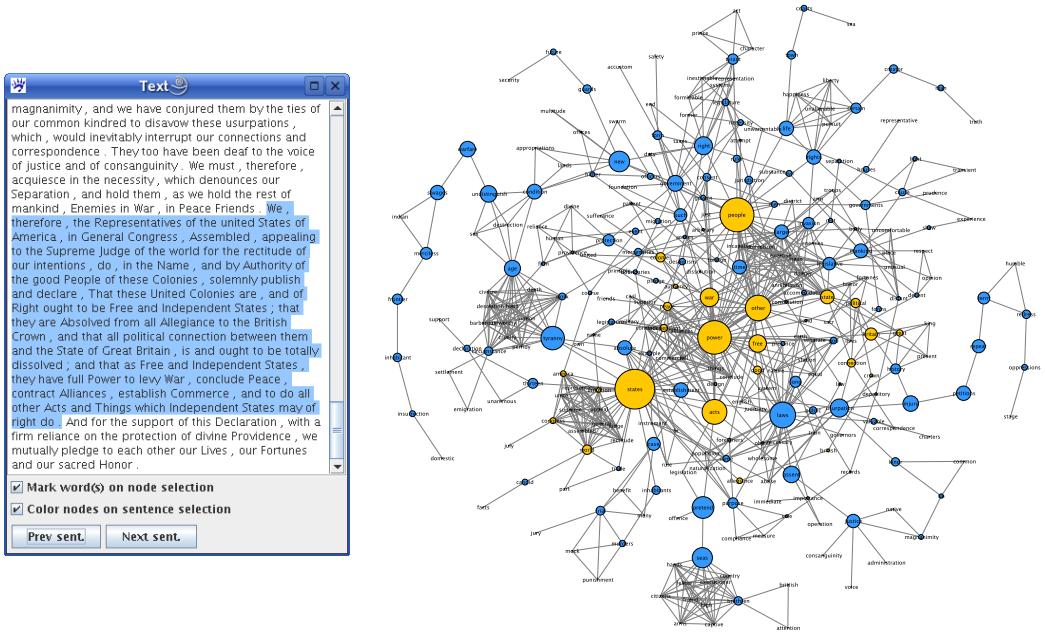

Abbildung 10: Textdialog und Netzwerk mit markiertem Satz und entsprechend hervorgehobenen Knoten

Ein solches Dialogfenster mit einem markierten Satz ist in Abbildung 10 zu sehen. In der größten Zusammenhangskomponente[12] des entsprechenden CRA-Netzwerks, das aus der amerikanischen Unabhängigkeitserklärung erstellt wurde,[13] sind die Wörter im Satz gelb gefärbt.

[12] ein maximaler, knoteninduzierter, zusammenhängender Teilgraph

[13] In einer Aufzählung mussten einige Punkte gesetzt werden, um sie in Sätze zu unterteilen, da sie sonst als ein extrem langer Satz erkannt wurden und der der Parsers den Arbeitsspeicher sprengte.

Diese Funktionalität unterstützt das Explorieren des Netzwerks, weil sich besser zurückverfolgen lässt, woher ein Knoten stammt und welche Teile des Netzwerks durch welchen Satz zustande gekommen sind.

4.3.3 Synonym-Behandlung

Trotz dass dies nicht Ziel der Arbeit ist, wurde eine rudimentäre Synonym- und Mehrwortbegriffs-Behandlung eingebaut. Aus einer Textdatei werden zeilenweise Regeln gelesen, die Ersetzungen im Originaltext beschreiben. Ein Vorkommen des rechten Teils einer Regel wird mit dem linken Teil ersetzt, der durch einen Schrägstrich abgetrennt wird. Beispielsweise bewirkt die Regel

```
World Trade Center / World_Trade_Center
```

die Ersetzung des Ausdrucks *World Trade Center* im Originaltext durch *World_Trade_Center*, wobei hier erwähnt sein muss, dass durch Unterstriche verbundene Wörter vom Parser als eines erkannt werden. Durch die Einführung einer zweiten Regel wie etwa

```
WTC / World_Trade_Center
```

lassen sich Synonyme definieren.

Auf der rechten Seite einer Regel sind auch reguläre Ausdrücke erlaubt, wie sie in der Klasse `java.util.regex.Pattern` in der Java-Standardbibliothek definiert sind. Dadurch lassen sich mehrere Regeln zusammenfassen, beispielsweise

```
(World )?Trade Center / World_Trade_Center
```

ersetzt sowohl *World Trade Center* als auch *Trade Center*. Das Fragezeichen ist ein „*greedy quantifier*“, bevorzugt wird also der längere Ausdruck.

Die ersetzten Wörter werden beim Netzwerk-Aufbau in Kleinbuchstaben umgewandelt, es sollten jedoch trotzdem Großbuchstaben verwendet werden, damit der Parser Sätze korrekt erkennt.

Der Implementation ist eine solche Synonym-Datei beigefügt (im Package unter `/nlp/resource/synonyms.txt`), die jedoch spezifisch auf die in Kapitel 5 untersuchten Texte zugeschnitten ist.

4.4 Benutzung der Software

Starten von Visone

Visone muss mit dem Parameter `-nlp` gestartet werden, damit beim Start die NLP-Erweiterung geladen wird. Außerdem sollte die *Java Virtual Machine* mit erhöhtem Arbeitsspeicher gestartet werden (z. B. mit `-Xmx500m` für 500 MB), da vor allem der Parser einen sehr hohen Speicheraufwand hat.

Am besten sollten die ausführbaren Dateien `visone.sh` bzw. `visone.bat` verwendet werden, die auf der CD beiliegen.

Erstellen eines Text-Netzwerks

Zuerst müssen eine oder mehrere Text-Dateien ausgewählt werden. Aus jeder Datei wird ein einzelnes Netzwerk erstellt.

Als nächstes muss die Art der Text-Netzwerk-Erstellung festgelegt werden. Verfügbar sind CRA und WNA, für die jeweils verschiedene Optionen eingestellt werden können. Bei der CRA ist wählbar, ob die Nominalphrasen auf oberster oder auf unterster Ebene aggregiert werden sollen. Bei der WNA ist die Fenstergröße zwischen 2 und 30 einstellbar.

Vor der Erstellung eines CRA-Netzwerks muss eine Parser-Datei geladen werden, die ein Wörterbuch, Ableitungsregeln und andere Daten enthält. Dazu kann eine der beiliegenden Dateien in der

Combobox neben *Parser* ausgewählt werden oder auch eine externe Datei durch Klick auf den nebenstehenden Button geöffnet werden. Die benötigten Dateien für den POS-Tagger werden beim Erstellen eines WNA-Netzwerks automatisch geladen, können aber im Voraus eingelesen werden.

Alle erstellten Netzwerke werden automatisch im selben Verzeichnis wie die Textdateien gespeichert. Durch Deselektieren der CheckBox unter *Destination* kann jedoch auch ein anderes Zielverzeichnis gewählt werden.

Das Abwählen der Checkbox *Display* verhindert, dass die erstellten Netzwerke im Editor dargestellt werden. Somit wird der Speicheraufwand herabgesetzt und der Prozess beschleunigt. Die Checkbox *Layout* gibt an, ob das Netzwerk nach der Erstellung mit einem Layoutverfahren[14] anschaulicher dargestellt werden soll. Bei Deselektieren dieser Funktion liegen alle Knoten auf einem Punkt.

Zentralitätsberechnung

Im Tab *Analysis* kann eine gewünschte Knoten-Zentralität berechnet werden, die unter *Index* auswählbar ist. Die berechnete Zentralität wird als Attribut zum Graphen gespeichert und ist über den Namen, der unter *Result Attribute* angegeben wurde, erreichbar, beispielsweise im Tab *Visualization*. Bei Speicherung des Graphen im *graphml*-Format werden die Zentralitäts-Werte mit abgespeichert.

Unter *Apply to* kann auch *Selected files* ausgewählt und somit eine Zentralität für eine Reihe von Graphen berechnet werden, die sukzessiv geladen, analysiert und deren Attribute (inklusive Zentralität) im CSV-Format[15] abgespeichert werden.

---

[14] es handelt sich dabei um einem *Spring-Embedder*

[15] CSV = *comma seperated values*

5 Verwendung von Text-Netzwerken

Im diesem Kapitel wollen wir davon ausgehen, dass aus einem oder einer Reihe von Texten ein, bzw. mehrere CRA-Netzwerke aufgebaut wurden und auf diesen eine geeignete Zentralität berechnet wurde. Hier sollen einige Ansätze zur weiteren Untersuchung und Visualisierung der Netzwerke und Zentralitäts-Daten dargestellt werden mit dem Ziel, den Inhalt der ursprünglichen Texte zu explorieren.

5.1 Testdaten

Zum Testen wurden Nachrichtentexte der Agentur Reuters verwendet, die in den ersten 66 Tagen ab den Anschlägen auf das World Trade Center in New York, also vom 11. September bis zum 15. November 2001, veröffentlich wurden.[16]

Die Texte sind deshalb gut geeignet, weil das Vokabular der Reuters-Redakteure stark reglementiert ist. Daher sind die in den Texten verwendeten Begriffe sehr differenziert und scharf definiert, wodurch die Repräsentation der Begriffe durch Knoten eine exaktere Wiedergabe der Information darstellt und besser untereinander vergleichbar ist als bei gewöhnlichen Texten.

Auch wenn sicherlich kein Nachrichtentext politisch ungefärbt ist, so können doch durch die Exploration der Texte und der entsprechenden Netzwerke die Ereignisse des Weltgeschehens in diesem Zeitabschnitt untersucht werden.

Abbildung 11 gibt einen kleinen Überblick über die Daten bezüglich der Anzahl an Sätzen, Wörtern und Schriftzeichen in den Texten und die Anzahl an Knoten und Kanten in den entsprechenden Texten. Festzustellen ist dabei eine starke Nachrichtenflut in den ersten sechs Tagen. Beim Vergleich verschieden langer Texte

[16] zur Herkunft der Daten siehe [DC02]

fällt auf, dass sich die Anzahl der Knoten nicht proportional zur Anzahl der Sätze, Wörter und Zeichen verhält, sondern in einem geringeren Maße steigt. Dies spricht für eine höhere Redundanz der vorkommenden Wörter in den größeren Texten.

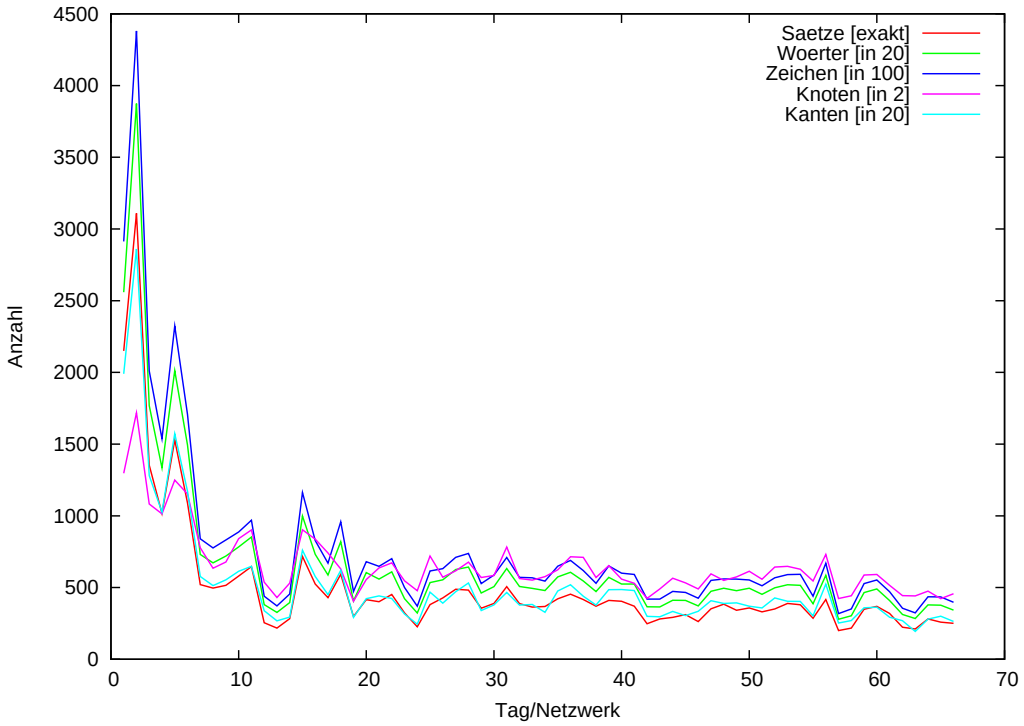

Abbildung 11: Einfache Eigenschaften der 66 Nachrichtentexte bzw. CRA-Netzwerke

Die CRA-Netzwerke aller 66 Nachrichtentexte im *graphml*-Format sowie die darauf berechneten, normierten Betweenness-Werte als CSV-Dateien liegen auf der CD bei. Bedauerlicherweise war es nicht möglich, die Originaltexte auf die CD zu brennen, da sie urheberrechtlich geschützt sind. Aus demselben Grund ist das Text-Rückverfolgungs-Tool auf den Netzwerken nicht funktionsfähig.

Um dies jedoch testen zu können, liegt auf der CD ein bereits in 4.3.2 erwähntes CRA-Netzwerk der amerikanischen Unabhängigkeitserklärung bei.

5.2 Pruning

Die Visualisierung von Text-Netzwerken macht es möglich, einige Eigenschaften des Netzwerks optisch abzulesen, beispielsweise mit welchen anderen Wörtern ein Wort in Verbindung gebracht wird oder wie oft zwei konkrete Wörter miteinander in Verbindung stehen. Auch gibt es die Möglichkeit, eine Zentralität über die Farbe oder Größe der Knoten zu visualisieren.

Die Voraussetzung dafür ist jedoch ein Layout für den Graphen, das heißt eine Einbettung der Knotenmenge in die zweidimensionale Ebene. Am einfachsten lassen sich Kanten als gerade Linien repräsentieren, wobei die Vielfachheit einer Kante durch die Dicke der Linie visualisiert werden kann.

Leider besitzen CRA-Netzwerke keine spezielle Eigenschaft, beispielsweise eine Baumform oder Planarität[17] oder eine zugrunde liegende Hierarchie, wie sie für einige Layoutalgorithmen Voraussetzung ist. Es gibt jedoch einen sehr einfachen aber effektiven Algorithmus, den es heute in vielen Variationen gibt: Beim sogenannten *Spring Embedder* [Ead84] wird als Kante – bildhaft gesprochen – eine Feder zwischen je zwei adjazente Knoten gespannt, die eine gewisse Länge im entspannten Zustand hat und die Knoten zusammenzieht, beziehungsweise auseinander treibt. Damit sich die Knoten gleichmäßiger in der Ebene verteilen, wird zusätzlich eine abstoßende Kraft zwischen allen Knotenpaaren definiert. In einem iterativen Prozess wird nun die „Spannung“ im Netzwerk minimiert.

Das grafische Darstellen eines gesamten CRA-Netzwerks wirft allerdings auf Grund der Größe häufig Probleme der Unübersichtlichkeit auf. So ist CRA-Netzwerk 23 mit 1093 Knoten und 5261 Kanten (Mehrfachkanten nur einzeln gezählt) noch passabel, auch wenn dabei Zooming unverzichtbar ist. Bei CRA-Netzwerk 02, dem

[17] das heißt, es gibt eine Einbettung, sodass sich keine Kanten überschneiden

größten der 66, mit seinen 3428 Knoten und 28 837 Kanten lässt sich dagegen im Kantenwirrwarr nichts mehr erkennen.

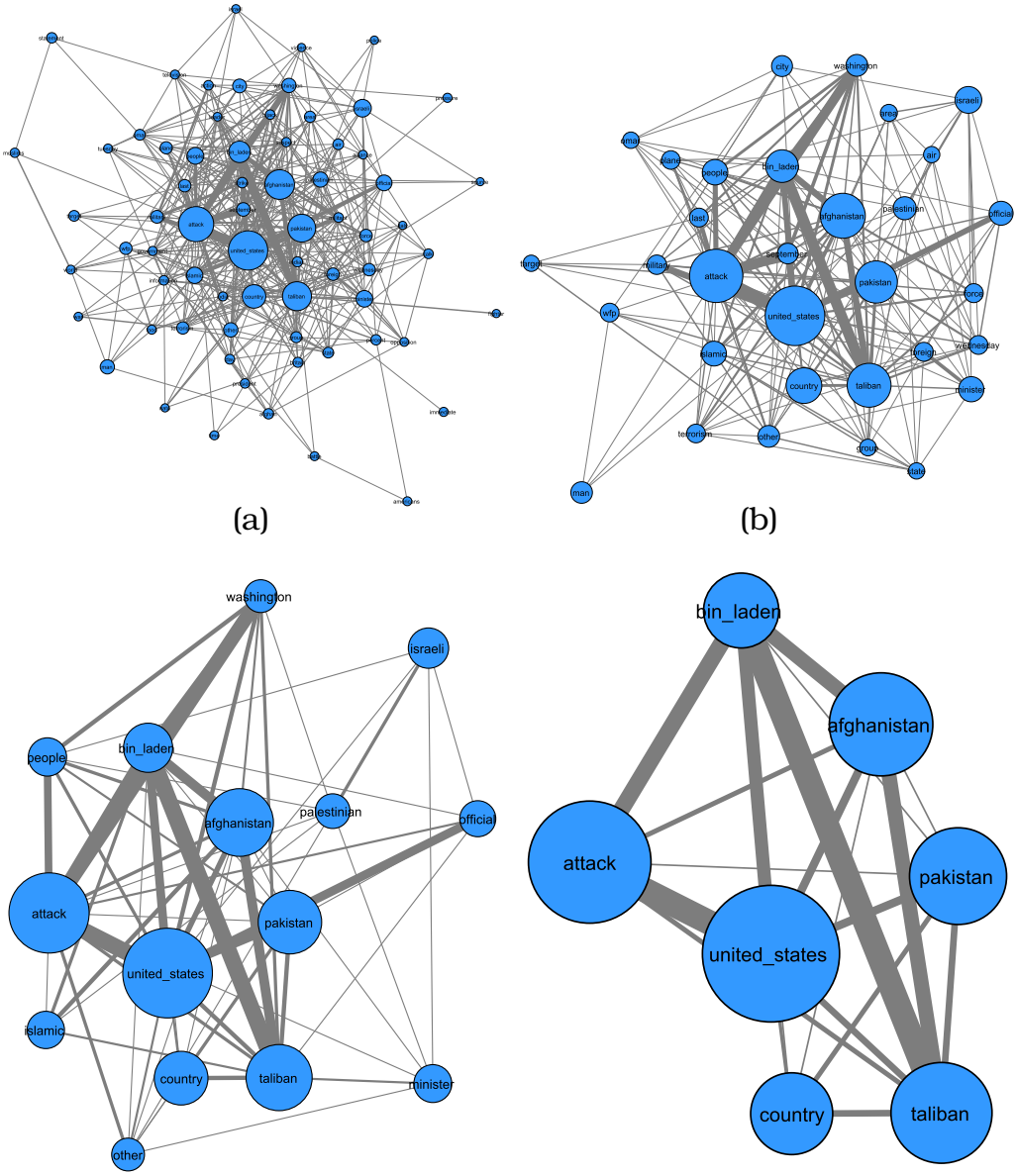

Abbildung 12: Geprunte CRA-Netzwerke des Textes 23 mit mehr als (a) 0,0031, (b) 0,0055, (c) 0,01 und (d) 0,02 Betweenness

Des Weiteren ist der Betrachter ohnehin meist nur an der oder den Kernaussagen des Textes interessiert und nicht an jedem einzelnen Begriff. Daher ist es sinnvoll, ein Text-Netzwerk zu *prunen* (beschneiden), das heißt einige Knoten aus dem Netzwerk zu entfernen, selbstverständlich inklusive ihrer inzidenten Kanten. Dazu kann eine Zentralität verwendet werden, um alle unwichtigen Knoten, also Knoten, deren Zentralitäts-Werte unterhalb einer bestimmten Schranke liegen, auszusondern.

Abbildung 12 zeigt vier geprunte Versionen des CRA-Netzwerks von Text 23. Durch die Dicke der Kanten wird deren Vielfachheit dargestellt und der Flächeninhalt eines Knotens ist proportional zum jeweiligen Betweenness-Wert. Zum Layouten wurde hier die Klasse `SmartOrganicLayouter` aus *yFiles* verwendet, die einen Mindestabstand der Knoten unter Berücksichtigung ihrer Größen garantiert.

Auf der CD liegen weitere Versionen und andere Netzwerke als Vektorgrafiken im PDF-Format bei, an die beliebig nah herangezoomt werden kann.

5.3 Zeitreihenanalyse

Dadurch dass wir im Besitz mehrerer Netzwerke sind, welche die Folgenglieder einer zeitlichen Abfolge darstellen, lässt sich die Berichterstattung zu einem bestimmten Thema über den Zeitverlauf analysieren. Zu diesem Zweck lassen sich *Zeitreihen* aus den Zentralitäts-Daten der Netzwerke extrahieren, indem für ein Wort der Zentralitäts-Wert des entsprechenden Knotens in jedem Netzwerk nachgeschlagen wird.

5.3.1 Identifizierung interessanter Zeitreihen

Ähnliche Verläufe in der Berichterstattung zu zwei bestimmten Wörtern lassen sich über die Korrelation ihrer Zeitreihen identifizieren.

Definition 5.1 (Korrelation)[18]

Die Korrelation *zwischen zwei Messreihen* $X = (x_1, \ldots, x_n)$ *und* $Y = (y_1, \ldots, y_n)$ *ist definiert durch*

$$\mathrm{Kor}(X, Y) = \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{Var}(X)} \cdot \sqrt{\mathrm{Var}(Y)}},$$

wobei $\mathrm{Var}(X)$ *die Varianz der Messreihe* X *und* $\mathrm{Cov}(X, Y)$ *die Covarianz von* X *und* Y *ist. Somit erhalten wir*

$$\mathrm{Kor}(X, Y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}},$$

wobei $\bar{x} := \frac{1}{n}\sum_{i=1}^{n} x_i$ *der Erwartungswert von* X *ist (analog für* $\bar{y}$*).*

Der Maximalwert 1 von $\mathrm{Kor}(X, Y)$ wird erreicht, wenn X und Y stark korrelieren, der Minimalwert -1 bei negativer Korrelation. Ein Wert von 0 entspricht keiner Korrelation.

Bei den 14 072 unterschiedlichen Wörtern, die in allen CRA-Netzwerken vorkommen, wären dies jedoch $\binom{14\,072}{2} = 99\,003\,556$ Wortpaare, deren Zeitreihen es zu vergleichen gilt. Dies verlangt nach einer Vorfilterung geeigneter Kandidaten von Wortpaaren.

Dazu kann die Entropie verwendet werdem, die den Informationsgehalt einer Messreihe wiedergibt.

[18] empirischer Korrelationskoeffizient nach Bravais, Pearson [Pea84]

Definition 5.2 (Entropie)[19]

Die Entropie *einer Messreihe* $X = (x_1, \ldots, x_n)$ *ist definiert durch*

$$H(X) = -\sum_{i=1}^{n} x_i \cdot \log_2(x_i).$$

Eine nach der Entropie ihrer Zeitreihen sortierte Liste aller Wörter und eine Liste mit Korrelationen aller Kombinationen der 150 Wörter mit der höchsten Entropie liegen auf der CD bei.

Einige starke Korrelationen sind weniger verblüffend, wie zum Beispiel *afghanistan* und *taliban* (0,612) oder *office* und *anthrax* (0,585). Interessanter ist jedoch das untere Ende der Liste, auf der sich Wörter mit einer negativen Korrelation finden. So scheinen *bin_laden* und *anthrax* (Korrelation -0,3598) in einem entgegengesetztem Verhältnis zueinander aufzutreten.

Um genauere Informationen über den zeitlichen Verlauf zu erhalten, wollen wir die Zeitreihen visualisieren.

5.3.2 Visualisierung

Wir werden für ein Wort jeweils die Zeitreihe der Betweenness- und der Grad-Zentralität des entsprechenden Knotens visualisieren. Dabei werden wir sehen, dass Betweenness- und Grad-Zentralität sich zwar an manchen Stellen ähnlich verhalten, jedoch oft stark voneinander abweichen. Um diese Unterschiede deutlich zu machen, soll hier eine Visualisierung-Technik entwickelt werden, die ein solches Paar von Wertereihen in einer Darstellung vereint.

Eine naheliegende Darstellung ist ein Streifen, dessen Ordinate die eine und deren Schnitt-Intervall mit der Parallelen zur y-Achse an der entsprechenden Abszisse die zweite Wertereihe wiedergibt. Dieses Layout wollen wir im Folgenden *Polygon*-Darstellung nennen. Ein Bespiel dafür ist in Abbildung 14 zu sehen. Das schlichte

[19] nach Shannon [SW48]

Antragen des zweiten Wertes in y-Richtung hat jedoch den signifikaten Nachteil, dass Ausschnitte der Kurve mit einer hohen Steigung optisch dünner erscheinen, da der Intervall-Abschnitt nicht orthogonal zur Kurve angetragen wird.

Das führt uns zu einer Darstellung, in der Kreise verwendet werden, um die zweite Zeitreihe zu visualisieren. Diese sollten dabei proportional zum Flächeninhalt der Kreise sein. Diese Darstellung wollen wir Kreisdarstellung nennen. Wie in Abbildung 13 gezeigt, können die Kreise durch tangential anliegende Trapeze verbunden werden, um so den Verlauf der Kurve besser nachvollziehbar zu gestalten. Mit Hilfe der Verbindungstrapeze können sowohl die Größe als auch die Ordinate von Kreisen, welche durch die Kreise einer anderen Kurve (teil-)verdeckt sind, optisch abgeschätzt werden.

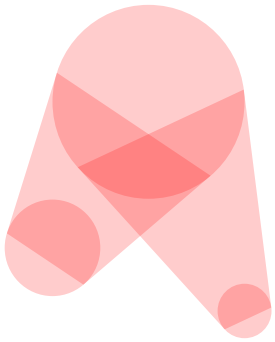

Abbildung 13: Tangential anliegende Verbindungstrapeze

Ein Nachteil der Kreisdarstellung ist, dass das Diagramm eine gewisse Breite haben muss, damit die Kreisflächen erkennbar sind. Dies lässt sich kompensieren, indem die Maximalbreite der Kreise so hoch gesetzt wird, dass sie sich zwar überlappen, ihre Größe jedoch erkennbar bleibt. Abbildung 15 zeigt dieselben Zeitreihen wie Abbildung 14 in der Kreisdarstellung. Trotz allem lässt sich die Polygondarstellung platzsparender umsetzen, da beide Wertereihe in y-Richtung angetragen werden.

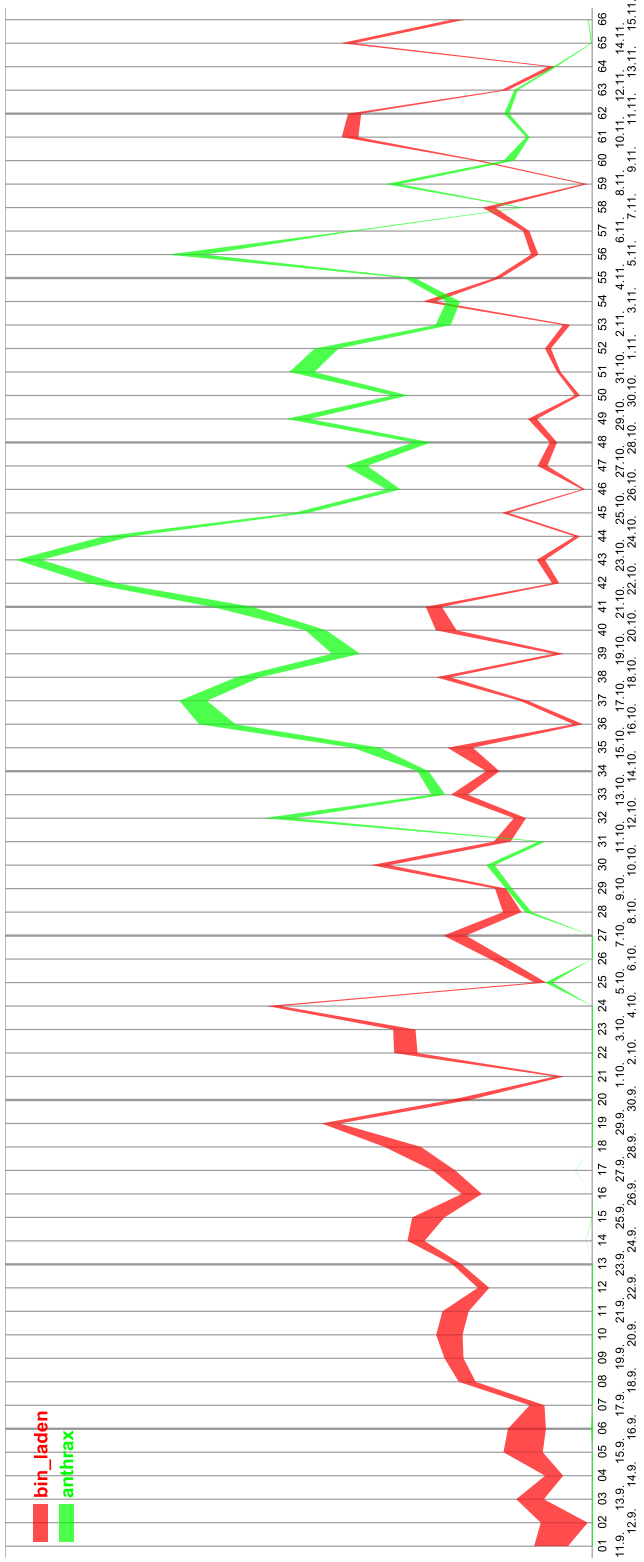

Abbildung 14: Zeitreihen von *bin_laden* und *anthrax* in der Polygondarstellung. Die Ordinate der Kurven repräsentiert die Betweenness und die Schnitt-Intervalle mit den Parallelen zur y-Achse geben die Gradzentralität wieder.

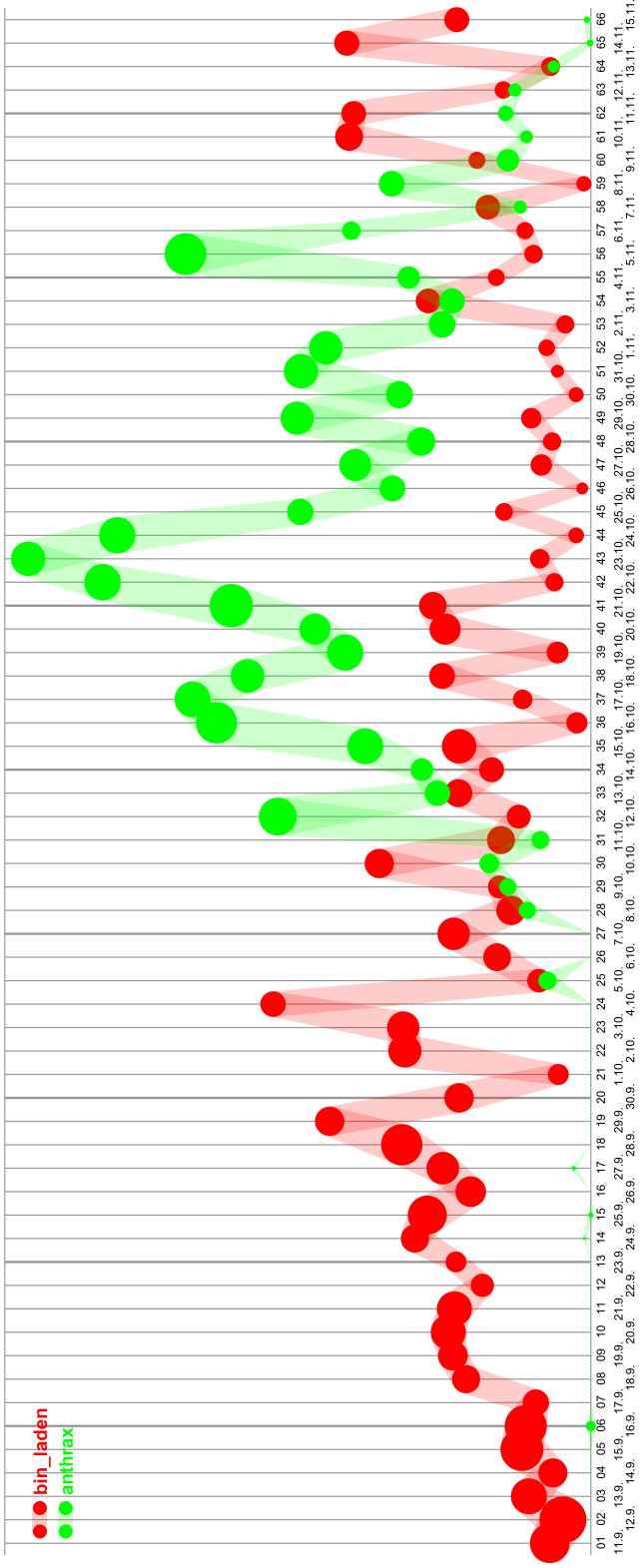

Abbildung 15: Zeitreihen von *bin_laden* und *anthrax* in der Kreisdarstellung. Die Ordinate der Kurven repräsentiert die Betweenness und die Flächeninhalte der Kreise geben die Gradzentralität wieder.

5.4 Satzfilterung

Eine denkbare Anwendung eines CRA-Netzwerks ist die Filterung von Sätzen mit einer möglichst hohen Relevanz, um so eine inhaltliche Kurzfassung des ursprünglichen Textes zu erhalten.

Die beiden Herausforderungen dieser Anwendung sind das Ermitteln einer geeigneten Gewichtsfunktion, um die Sätze sortieren zu können, und das Eliminieren von Sätzen mit der gleichen oder einer ähnlichen Aussage, um nicht eine redundante Zusammenfassung des Textes zu erhalten, die womöglich nur eines von mehreren Hauptthemen des Textes wiedergibt.

Der Text bestehe also aus p Sätzen und die Knotenmengen $V_1, \ldots, V_p \subset V$ beinhalten jeweils diejenigen Wörter, die in einem Satz vorkommen und auf einen Knoten abgebildet wurden. Das Gewicht eines Satzes V_i wollen wir mit $w(V_i)$ bezeichnen.

5.4.1 Gewichtsfunktionen

Maximums-Gewichtung Hier wird als Gewicht des Satzes die Zentralität des Knotens im Satz mit der höchsten Zentralität gewählt:

$$w(V_i) = \max_{v \in V_i} \{c(v)\}$$

Das Problem hierbei ist, dass der Knoten mit dem höchsten Zentralitäts-Wert im gesamten Netzwerk bei einer geeigneten Zentralität meist in entsprechend vielen Sätzen auftritt. All jene Sätze bekommen bei diesem Ansatz das höchste und vor allem identische Gewicht zugewiesen. Bei den Beispieltexten liegt der Anteil jener Sätze in der Größenordnung $\frac{1}{8}$ bis zu $\frac{1}{3}$.

Dadurch ergibt sich das technische Problem, wie diese große Menge gleichgewichteter Sätze wiederum untereinander gerankt werden soll.

Durchschnitts-Gewichtung Eine naheliegende Gewichtsfunktion ist die Durchschnittsbildung aller Zentralitätswerte im Satz:

$$w(V_i) = \frac{\sum_{v \in V_i} c(v)}{|V_i|}$$

Bei Anwendung dieser Gewichtsfunktion lässt sich beobachten, dass die am höchsten gewichteten zwei oder drei Sätze fast immer sehr kurz sind, z. B. „*We did not expect that they would enter Kabul.*“ (Satz 145, Text 64). Dieser Satz erhält, neben einem zweiten (Satz 280), mit großem Abstand das maximale Gewicht im zugehörigen Text, da der einzige Knoten *kabul* eine sehr hohe Betweenness besitzt und es keine weiteren Substantive oder Adjektive im Satz gibt. Dieser Satz ist für eine Zusammenfassung jedoch gänzlich ungeeignet; sein Informationsgehalt ist sehr niedrig, da er komplett aus dem Zusammenhang gerissen ist. Wir wissen nicht (bzw. können nur vermuten), wer sich hinter *we* und *they* verbirgt.

Eine Idee zur Lösung des Problems wäre das Bestrafen von Pronomen, da deren Vorkommen den Informationsgehalt des Satzes senkt, wie wir im vorangegangenen Beispiel sehen können. Allerdings kann ein Satz wie zum Beispiel „*But the U.S. has said it will continue the war.*“ (Satz 46, Text 66) trotz Pronomen eigenständig und aussagekräftig sein, da sich das Pronomen *it* auf *U.S.* im selben Satz bezieht. Vielleicht könnte hierbei eine Einschränkung auf diejenigen Pronomen, die nur im Hauptsatz vorkommen, Abhilfe schaffen. Welcher Teil des Satzes der Hauptsatz ist, lässt sich leicht aus dem Syntaxbaum ableiten.

TopK-Gewichtung In einem Hybridverfahren ließe sich der Durchschnitt über die k wichtigsten Knoten im Satz bilden, wobei an k fehlende Knoten mit 0 bestraft werden.

Es sei also $V_i = \{v_1, \ldots, v_l\}$ absteigend nach Zentralitäten sortiert und $c(v_j) = 0$ für ein $j > l$. Da k für alle Sätze gleich ist, ist die

Division durch k für den Vergleich der Gewichte der Sätze hinfällig:

$$w(V_i) = \sum_{j=1}^{k} c(v_j)$$

Wenn k zu klein gewählt wird, treten Probleme auf, wie sie bei der Maximums-Gewichtung geschildert wurden. Wenn k zu groß ist, werden kurze Sätze zu stark bestraft und die Zusammenfassung besteht nur aus sehr langen Sätzen. Eine Wahl von $k = 3$ scheint für die Beispieldaten eine geeignete Wahl zu sein.

Einen weiteren Parameter, den es noch einzustellen gilt, ist die Behandlung von mehrfachen Vorkommen eines Wortes in einem Satz. Vor allem bei der TopK-Gewichtung spielt es eine entscheidende Rolle, ob die V_i als Multimengen definiert sind, in denen ein Wort mehrfach vorkommen kann.

5.4.2 Redundanzbehandlung

Das Identifizieren und Auswählen von inhaltlich möglichst unterschiedlichen Sätzen entspricht einem Clustering und der Bestimmung von geeigneten Repräsentanten für jedes Cluster. Gesucht ist hier demnach eine Abstandsfunktion, die die inhaltliche Unähnlichkeit zweier Sätze formalisiert.

In [CKMD02] wird ein Maß basierend auf CRA-Netzwerken vorgeschlagen, das zum inhaltlichen Vergleich zweier Texte dient, die *Word Resonance*. Damit kann die CRA einen Beitrag im Bereich des *positioning* leisten.

Definition 5.3 (Word Resonance)

Seien die beiden Texte T_1 und T_2 gegeben sowie die zugehörigen CRA-Netzwerke $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$ und eine Zentralität c. Außerdem erhalte der Indikator α für zwei Knoten $v \in V_1$ und

$w \in V_2$ *den Wert* 1, *falls beide dasselbe Wort repräsentieren, andernfalls* $\alpha_{v,w} = 0$. *Dann ist die* Word Resonance *von* T_1 *und* T_2 *definiert durch*

$$\mathrm{WR}_{T_1T_2} = \sum_{v \in V_1} \sum_{w \in V_2} c(G_1)_v \cdot c(G_2)_w \cdot \alpha_{vw}.$$

Die Formel entspricht dem Skalarprodukt und damit dem Cosinus des Winkels der beiden Zentralitäts-Vektoren in einem hochdimensionalen Raum, der von den Knoten der beiden CRA-Netzwerke aufgespannt wird. Zur Veranschaulichung zeigt Abbildung 17 eine symmetrische Matrix, in der die Word-Resonance-Werte zwischen je zwei Texten farblich eingetragen sind. Hohe Werte sind rot, niedrige blau.

Abbildung 16: Word-Resonance-Matrix der 66 Nachrichtentexte

Nach der obigen Definition ist das Maß jedoch nicht standardisiert. Zwei Texte werden in Unabhängigkeit der Anzahl an Knoten

in ihren CRA-Netzwerken verglichen. Dadurch sind die Ähnlichkeiten zweier Textpaare schlechter vergleichbar. Dies motiviert, die Word Resonance wie folgt zu standardisieren:

$$\mathrm{WR}'_{T_1T_2} = \mathrm{WR}_{T_1T_2} \cdot \left(\sqrt{\sum_{v \in V_1} c(G_1)_v^2 \cdot \sum_{w \in V_2} c(G_2)_w^2} \right)^{-1}$$

Abbildung 17 zeigt die Matrix mit standardisierter Word Resonance.

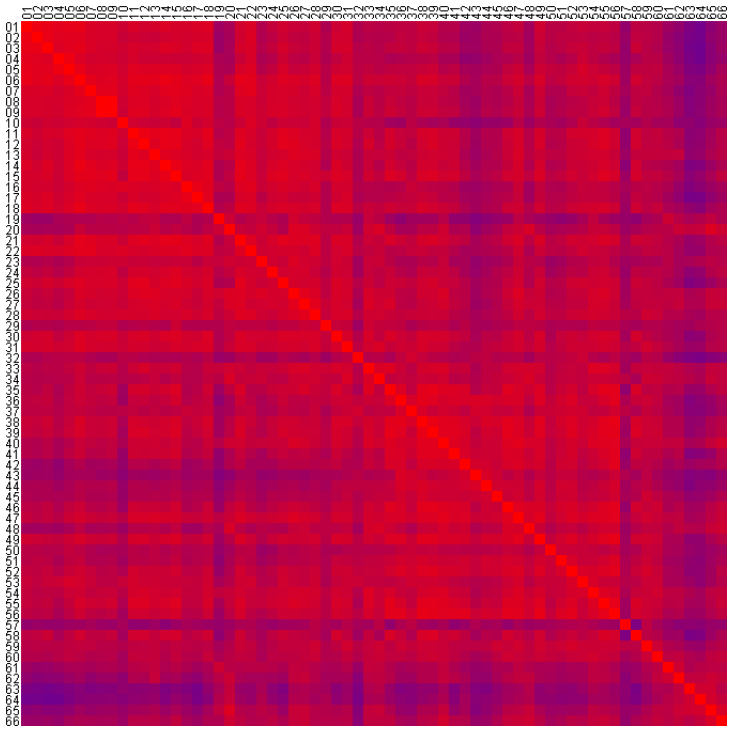

Abbildung 17: Standardisierte Word-Resonance-Matrix der 66 Nachrichtentexte

Die Word Resonance lässt sich jedoch nicht ohne Weiteres auf den Vergleich von Sätzen übertragen. Aus einem einzelnen Satz ein CRA-Netzwerk aufzubauen und darauf eine Zentralität zu berechnen, ist nicht besonders sinnvoll. Ein zu kleines Netzwerk bildet in der Regel eine Kette, was zur Folge hat, dass die Wörter, die exakt in der Mitte des Satzes liegen, am zentralsten bewertet werden.

Stattdessen wäre es möglich, für die Knoten in V_i als Zentralitäts-Werte diejenigen im CRA-Netzwerk des gesamten Textes zu verwenden.

Ziel der weiteren Forschung auf diesem Gebiet ist die Validierung der Ergebnisse. Dazu müssten manuelle Zusammenfassungen von repräsentativen Texten erstellt werden und mit den maschinell erzeugten Ergebnissen verglichen werden. Auch bedarf es einer Weiterentwicklung der Satzauswahl-Strategien und geeigneter Satzdistanzfunktionen.

6 Zusammenfassung und Ausblick

In dieser Arbeit wurde das Verfahren *Centering Resonance Analysis* zur Erstellung von Text-Netzwerken aus natürlich sprachlichen Texten dargelegt und mit einem älteren Ansatz, der *Word-Network Analysis*, verglichen. Des weiteren wurde gezeigt, dass sich Zentralitäten verwenden lassen, um die Knoten im Netzwerk strukturell zu bewerten. Eine Implementation der beiden Verfahren wurde vorgestellt, sowie einige Ansätze zur Verwendung von CRA-Netzwerken mit dem Ziel, die ursprünglichen Texte zu explorieren.

Verbesserungspotential besteht in der Geschwindigkeit der Implementation. Beim Aufbau eines WNA-Netzwerks interessieren nur die Wortarten und somit konnte die Erstellung aller 66 Netzwerke mit Hilfe des POS-Taggers in nur 1 Stunde 45 Minuten durchgeführt werden. Die Erstellung aller CRA-Netzwerke benötigte hingegen mehr als 15 Stunden. Der entscheidende zeitliche Faktor dabei ist das Parsen der Texte. Eine Möglichkeit zur Optimierung in diesem Punkt wäre eine Art *NP-Chunker*, der nur die Nominalphrasen im Satz auf einer vorgegebenen Ebene identifiziert und nicht die komplette Struktur des Satzes ermittelt. In der Implementation des Stanford Parsers sind diese beiden Funktionen jedoch zu stark miteinander verwoben, als dass eine solche Funktionalität kurzerhand extrahiert werden könnte.

Weiterer Forschungsbedarf in diesem Gebiet bleibt die Justierung des Verfahrens. So steht beispielsweise die Frage offen, auf welcher Ebene die Nominalphrasen ausgewählt werden sollen. Ebenso bedürfen die vorgestellten Ansätze zur Visualisierung von Netzwerken und Zeitreihen sowie die Identifikation und Filterung von inhaltlich repräsentativen Sätzen einer Weiterentwicklung und Parameterjustierung.

Darüber hinaus werden repräsentative Testdaten und Methoden zur empirischen Validierung und Qualitätsmessung benötigt. Das in dieser Arbeit vorgestellte Rückverfolgungstool für Text-Netzwerke wäre dabei eine Unterstützung, indem auftretende Phänomene direkt an den Textstellen überprüft werden können.

Referenzen

[Ant71] Anthonisse, J. M. (1971). The rush in a directed graph. Technical Report BN 9/71, Stichting Mathematisch Centrum, 2e Boerhaavestraat 49 Amsterdam.

[BR99] Baeza-Yates, R., Ribeiro-Neto, B. A. (1999). Modern information retrieval. *Addison-Wesley Longman*, Harlow, England.

[Bea65] Beauchamp, M. A. (1965). An improved index of centrality. *Behavioral Science*, 10:161–163.

[Blu06] Blumenthal, J. (2006). Netzwerk-Textanalyse. Bachelorarbeit, Universität Konstanz, Fachbereich Informatik und Informationswissenschaft.

[Bra01] Brandes, U. (2001). A Faster Algorithm for Betweenness Centrality. *Journal of Mathematical Sociology* 25(2):163–177.

[BF05] Brandes, U. & Fleischer, D. (2005). Centrality Measures Based on Current Flow. *Proc. 22nd Symp. Theoretical Aspects of Computer Science (STACS '05)*. LNCS 3404, S. 533–544. Springer-Verlag.

[CKMD02] Corman, S. R., Kuhn, T., McPhee, R. D. & Dooley, K. J. (2002). Studying Complex Discursive Systems. Centering Resonance Analysis of Communication. *Human Communication Research*, Vol. 28, No. 2.

[Dan82] Danowski, J. A. (1982). A network-based content analysis methodology for computer-mediated communication: An illustration with a computer bulletin board. *Communication yearbook 6*, S. 904–925.

[Dan93] Danowski, J. A. (1993). Network analysis of message content. In W. D. Richards & G. A. Barnett (Hrsg.), *Progress in Communication sciences* (Vol. 12, S. 197–221). Norwood, NJ: Ablex.

[DC02] Dooley, K. & Corman, S. (2002). The dynamics of electronic media coverage. In B. Greenberg (Hrsg.) *Communication and Terrorism* (S. 121–136). Creskill, NJ: Hampton Press.

[Ead84] Eades, P. (1984). A heuristic for graph drawing. *Congressus Numerantium 42*, 149–160.

[Fre77] Freeman, L. C. (1977). A set of measures of centrality based upon betweeness. *Sociometry*, 40:35–41.

[GPL07] GNU General Public License (Stand März 2007) `http://www.gnu.org/copyleft/gpl.html`

[GGG93] Gordon, P. C., Grosz, B. J., & Gilliom, L. A. (1993). Pronouns, names, and the centering of attention in discourse. *Cognitive Science, 17*, 311–347.

[GWJ95] Grosz, B. J., Weinstein, S. & Joshi, A. K. (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21, S. 203–225.

[MNa07] Vorlesung *Methoden der Netzwerkanalyse* (Brandes) mit Skript (Stand: 18. 01. 2007) `http://www.inf.uni-konstanz.de/algo/lehre/ss05/mna`

[MSM93] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz (1993). Building a Large Annotated Corpus of English: The Penn Treebank, in Computational Linguistics, Volume 19, Number 2, S. 313–330.

[Pea84] K. Pearson, K. (1896). Mathematical contributions to the theory of evolution. III. Regression, heredity and panmixia. *Philos. Trans. Royal Soc. London Ser. A*, 187, S. 253–318.

[SW48] Shannon, C. E. & Weaver, W. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal* (Vol. 27, S. 379–423, 623–656).

[Sta07] Website der Stanford Natural Language Group `http://nlp.stanford.edu` (Stand März 2007)

[Vis07] Visone-Projekt zur Analyse, Visualisierung von sozialen Netzwerken, `http://visone.info` (Stand: 18. 01. 2007)

[WJP98] Walker, M. A., Joshi, A. K. & Prince, E. F. (Hrsg.). (1998). *Centering theory in discourse*. New York: Oxford.

[yWo07] yWorks GmbH, Entwickler von *yFiles* `http://www.yworks.com` (Stand März 2007)

Abbildungsverzeichnis

Tabellenverzeichnis

Danksagungen

Ich danke meiner Frau Kathrin, die mich beim Anfertigen dieser Arbeit immer wieder motiviert hat und jeder Zeit für mich da war. Ebenso möchte ich meiner Familie und Prof. Ulrik Brandes für ihre Unterstützung danken.

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig und nur mit den angegebenen Hilfsmitteln angefertigt habe und dass alle Stellen, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, durch Angabe der Quellen als Entlehnungen kenntlich gemacht worden sind.

Diese Arbeit wird nach Abschluss des Prüfungsverfahrens der Universitätbibliothek Konstanz übergeben und ist durch Einsicht und Ausleihe somit der Öffentlichkeit zugänglich. Als Urheber der vorliegenden Arbeit stimme ich diesem Verfahren zu.

Ort, Datum Unterschrift

Anlage: CD-ROM

Pfad	**Beschreibung**
/Bachelorarbeit.pdf	Diese Bachelorarbeit in digitaler Form
/visone/visone.jar	Visone als ausführbare JAR-Datei
/nlp/	Quelldateien des nlp-Packages
/DeclarIndep/	Amerikanische Unabhängigkeitserklärung als Text, CRA-Netzwerk und PDFs
/cra/networks/	CRA-Netzwerke der Reuters-Newstexte
/cra/Betweenness/	Betweenness-Zentralitäten der CRA-Netzwerke als CSV-Dateien
/cra/Degree/	Grad-Zentralitäten
/cra/pdf/	CRA-Netzwerke als PDF-Dateien
/cra/	Entropien, Korrelationen