

Institut für Visualisierung und Interaktive Systeme

Universität Stuttgart
Universitätsstraße 38
D-70569 Stuttgart

Bachelorarbeit

Interaktive, vergleichende Visualisierung von Figurennetzwerken

David Schütz

Studiengang:	Softwaretechnik
Prüfer/in:	Prof. Dr. Thomas Ertl
Betreuer/in:	Markus John, M.Sc. Dipl.-Math. Martin Baumann, M.A. Dr. Steffen Koch
Beginn am:	27. April 2017
Beendet am:	27. Oktober 2017
CR-Nummer:	H.5.2, J.5

Kurzfassung

Immer häufiger greifen Literaturwissenschaftler auf maschinelle Unterstützung zurück, um literarische Texte zu analysieren. Denn die automatische Analyse ist nicht nur deutlich schneller als die manuelle, Ergebnisse können auch übersichtlich mit der Hilfe von visuellen Strukturen auf dem Bildschirm dargestellt werden. Darüber hinaus geben die Visualisierungen nicht nur einen guten Überblick über die explorierte Datenmenge, sondern ermöglichen auch durch verschiedene Werkzeuge eine detaillierte Analyse des Textes. So können Figurennetzwerke, welche Entitäten und deren Relationen innerhalb eines literarischen Textes beschreiben, graphisch dargestellt werden. Häufig sind Analysten daran interessiert, wie sich Figurennetzwerke über die Handlung eines Buches hinweg verändern. Auch bietet der Vergleich zweier Figurennetzwerke forschungsrelevante Informationen. In dieser Arbeit wird ein Konzept vorgestellt, um mehrere Figurennetzwerke gleichzeitig darzustellen und vergleichen zu können. Dabei wird eine visuelle Struktur konzipiert, welche zwei Figurennetzwerke in einer hybriden Darstellung aus Superimposition und Explicit Encoding vergleicht. Durch verschiedene Filtermethoden und eine Zoomfunktion ist es möglich, die als Node-Link Diagramm oder Adjazenzmatrix dargestellten Figurennetzwerke, genauer zu untersuchen. Anschließend wird das Konzept als webbasierte Anwendung implementiert und präsentiert. In einem vorgestellten Anwendungsfall wird die Bedienung und Praktikabilität der Anwendung gezeigt. Zuletzt wurde das System verschiedenen wissenschaftlichen Mitarbeitern der Literatur- und Sozialwissenschaften vorgestellt. Die Rückmeldung der Experten war dabei durchweg positiv und es wurden verschiedene Erweiterungen der Anwendung diskutiert.

Inhaltsverzeichnis

1	Einleitung	11
2	Grundlagen	13
2.1	Digital Humanities	13
2.2	Netzwerkgraph	13
2.3	Visual Information-Seeking Mantra	15
2.4	Das Referenzmodell für Visualisierung	15
2.5	Graphenvergleich	16
2.6	Visualisierung von dynamischen Graphen	18
3	Verwandte Arbeiten	21
3.1	Verwandte Arbeiten zum Graphenvergleich anhand eines Node-Link Diagramms oder anhand einer Adjazenzmatrix	21
3.2	Verwandte Arbeit zur hybrid Darstellung aus Node-Link Diagramm und Adja- zenzmatrix	25
4	Konzept	27
4.1	Allgemeines Vorgehen	27
4.2	Importieren der Datenmenge	27
4.3	Graphdarstellung	28
4.4	Vergleich zweier Graphen	33
4.5	Entitätenliste	37
4.6	Rasteransicht	38
5	Implementierung	39
5.1	Verwendete Technologien	39
5.2	Benutzeroberfläche	39
5.3	Graph Visualisierung	43
6	Evaluation	49
6.1	Anwendungsfall	49
6.2	Expertenfeedback	54
7	Zusammenfassung und Ausblick	57
	Literaturverzeichnis	59

Abbildungsverzeichnis

2.1	Ein Netzwerkgraph als Node-Link Diagramm und als Adjazenzmatrix dargestellt aus [GFC04].	14
2.2	Das Referenzmodell für Visualisierung von Card et al. [CMS99].	16
2.3	Beispiele der verschiedene Vergleichskategorien anhand von zwei Netzwerkgraphen aus [GAW+11].	17
2.4	Time-to-time Mapping und Time-to-space Mapping gegenübergestellt [BBD13].	19
3.1	Die Darstellung zweier Graphen als Node-Link Diagramm links und rechts, so wie mittig eines vergleichenden Graphen im Semantic Graph Visualiser von Andrews et al. [AWW09].	21
3.2	Die Darstellung zweier dynamischer Graphen durch Time-to-time Mapping oben und Time-to-Space Mapping unten von Purvi et al. [SLN05].	22
3.3	Die von Beck et al [BBW16] erstellte Matrix um die Leistung zweier Sportler zu vergleichen.	23
3.4	Anordnung um zwei dynamische Graphen in einer Matrixdarstellung zu vergleichen von Burch [Bur16].	24
3.5	Vier Matrizen mit Ergebnissen eines Superbowls links in Superimposition zu einer Matrix zusammengefasst und rechts in Juxtaposition nebeneinander dargestellt von Melville et al.[MGK11].	25
3.6	Die verschiedenen Darstellungsoptionen in [ABH+13] um zwei Graphen mit Hilfe einer Adjazenzmatrix oder einem Node-Link Diagramm darzustellen.	25
3.7	Die verschiedenen Darstellungsoptionen in [ABH+13] um zwei Graphen mit Hilfe einer Adjazenzmatrix oder einem Node-Link Diagramm darzustellen.	26
4.1	Der für die Anwendung konzipierte Prozess zu einer visuellen Repräsentation, nach dem Referenzmodell von Card et al. [CMS99].	28
4.2	Entwurf einer Adjazenzmatrix um ein Figurennetzwerk darzustellen.	31
4.3	Die Anordnung der Graphdarstellung um einen Graphen genauer untersuchen zu können und mehrere Graphen in Juxtaposition zu vergleichen.	32
4.4	Verschiedene Entwürfe von Knoten- und Kantendarstellungen eines vergleichenden Node-Link Diagramms.	35
4.5	Matrixbasierte Darstellungsansätze um zwei Graphen miteinander zu vergleichen.	37
4.6	Die Rasteransicht gibt einen Überblick über alle erstellbaren Vergleichsgraphen.	38
5.1	Benutzeroberfläche mit geöffneter Graphansicht(1).	40
5.2	Ein einzelnes Node-Link Diagramm in der Großansicht dargestellt.	41
5.3	Die implementierte Rasteransicht, welche einen Überblick über alle erstellbaren Vergleichsgraphen gibt.	42
5.4	Die Graphübersichtsliste mit eingblendeter Kurzinformation.	43

5.5	Ein groß dargestelltes Node-Link Diagramm, in welchem jede Knotenbeschriftung eingeblendet ist (a) und ein Node-Link Diagramm, aus welchem herausgezoomt wurde und in welchem einzelne Beschriftungen aufgrund von Überschneidungen ausgeblendet wurden (b).	43
5.6	Die implementierte Adjazenzmatrix.	44
5.7	Vergleichsgraph zu Datensätzen von Parzival Buch drei und vier. Einmal mit Satzsegmentierung und einmal mit 30 Zeilensegmentierung. In a) werden Längenvergleichende Kanten und Knoten in Form von Balkendiagrammen dargestellt. Breitenvergleichende Kanten und Kreisdiagramme als Knoten sind in b) zu sehen.	45
5.8	Durch das Halten des Mauszeigers über einen Knoten im vergleichenden Node-Link Diagramm, werden alle in Relation stehenden Knoten und Kanten hervorgehoben	46
5.9	Die implementierte vergleichende Adjazenzmatrix nach Vorkommen der Entitäten sortiert.	47
6.1	Benutzeroberfläche mit den Figurennetzwerken zu Buch drei und vier von Parzival.	49
6.2	Der Vergleichsgraph zwischen Buch drei und vier von Parzival als Node-Link Diagramm in der Großansicht.	50
6.3	Der Vergleichsgraph zwischen Buch drei und vier von Parzival als Adjazenzmatrix in der Großansicht.	51
6.4	Der Vergleichsgraph zwischen Buch drei und vier von Parzival in dem alle Knoten und Kanten angezeigt werden, welche in beiden Büchern vorkommen.	52
6.5	Der Vergleichsgraph zwischen Buch drei und vier von Parzival, sowie die Figurennetzwerke zu den einzelnen Büchern nach Entitäten gefiltert, welche in beiden Büchern vorkommen.	52
6.6	Alle vier Datensätze über Buch vier von Parzival in der Graphansicht.	53
6.7	Die vier importierten Graphen, sowie alle Vergleichsgraphen in der Rasteransicht.	53
6.8	Der Vergleichsgraph mit geänderten Kanten und Knotendarstellungen in der Großansicht.	54
6.9	Unterschiede der Relationen in Buch vier von Parzival wenn Appellative berücksichtigt werden (blau) oder nicht berücksichtigt werden (grün).	55

Tabellenverzeichnis

4.1	Alle Attribute zu den vier gegebenen GEXF-Dateien über das Buch Parzival . . .	29
-----	--	----

1 Einleitung

Das manuelle Analysieren von Literatur ist eine zeitaufwändige Aufgabe. Aus diesem Grund wird in der heutigen Zeit immer häufiger auf maschinelle Unterstützung zurückgegriffen. Das Forschungsgebiet der Digital Humanities befasst sich mit dem Bearbeiten von geisteswissenschaftlichen Fragestellungen durch die Hilfe von computergestützten Methoden. So können automatische Analysemethoden Texte innerhalb kürzester Zeit analysieren und Informationen zu Entitäten, wie zum Beispiel Personen, und deren Relationen extrahieren. Figurennetzwerke, welche diese Daten beschreiben, können anschließend graphisch dargestellt werden. Die Visualisierung von Figurennetzwerken ist ein bereits stark erforschtes Thema und ein gängig genutztes Mittel von Geisteswissenschaftlern. Die am häufigsten verwendeten Darstellungsformen eines Figurennetzwerks sind das Node-Link Diagramm und die Adjazenzmatrix. Durch die Visualisierung der explorierten Datenmenge ist es möglich Informationen zu Entitäten, so wie deren Beziehungen zu erlangen, auch können einfacher Muster oder Besonderheiten in der Datenmenge ausfindig gemacht werden. Dabei stößt die Visualisierung solcher Datenmengen auf viele Herausforderungen. Häufig sind Figurennetzwerke komplex und daher schwer auf dem Bildschirm übersichtlich darzustellen. Aufgrund dessen ist es notwendig Filter und Analysewerkzeuge bereitzustellen, welche die dargestellten Informationen reduzieren und die Analyse durch den Benutzer vereinfachen. Neben dem Analysieren einzelner Figurennetzwerke ist auch die Entwicklung dieser über den Verlauf eines Textes interessant. Hierfür ist es notwendig, die Netzwerkgraphen dynamisch darstellen zu können. Dadurch können Literaturwissenschaftler wichtige Wendepunkte leichter in der Handlung eines literarischen Textes finden, sowie die Entwicklung von Entitäten und deren Relationen beobachten. Auch ist der Vergleich von Figurennetzwerken zu verschiedenen Abschnitten eines literarischen Textes von Interesse.

Ziel dieser Arbeit ist es, ein Konzept zur Visualisierung dynamischer Figurennetzwerke, sowie deren Vergleich vorzustellen. Dabei soll der Benutzer die dargestellte Datenmengen mit der Hilfe von ausgewählten Filtermöglichkeiten und einer Zoom-Funktion genauer untersuchen können. Auch soll eine visuelle Struktur konzipiert werden, um zwei Figurennetzwerke miteinander vergleichen zu können. Als Grundlage stehen vier Datensätze zur Verfügung, welche im Rahmen des Forschungsprojekts „Centrum für reflektierte Textanalyse“¹ (CRETA) entstanden sind. CRETA befasst sich mit dem Bereich der Digital Humanities, genauer gesagt mit allgemeinen und technischen Methoden, um Texte analysieren und visualisieren zu können. Nachdem das Konzept entwickelt und vorgestellt wurde, soll es anschließend anhand einer webbasierten Anwendung implementiert werden. Ziel der Anwendung ist es Geisteswissenschaftlern bei der Bearbeitung von Forschungsthemen zu unterstützen. Deswegen soll eng mit den Beteiligten des CRETA-Projektes zusammengearbeitet werden. Die Bedienung und Praktikabilität der Anwendung soll anhand eines Anwendungsfalles gezeigt werden. Abschließend soll der implementierte Ansatz durch Expertenfeedback evaluiert werden.

¹<https://www.creta.uni-stuttgart.de/>

Gliederung

Die Arbeit ist in sieben Kapitel gegliedert. Nach der Einleitung werden Grundlagen der Visualisierung und des Graphenvergleiches vorgestellt. Anschließend werden in Kapitel 3 verwandte Arbeiten präsentiert. In Kapitel 4 wird ein Konzept erarbeitet, um die Problemstellung zu lösen und anschließend wird eine prototypische Implementierung des Konzeptes in Kapitel 5 dargestellt. Daraufhin wird in Kapitel 6 die Implementierung anhand zweier Anwendungsfälle vorgeführt und durch eine Expertenbefragung evaluiert. Letztlich wird die vorgestellte Arbeit in Kapitel 7 zusammengefasst und es wird ein Ausblick auf Verbesserung und zukünftige Arbeit gegeben.

2 Grundlagen

In diesem Kapitel werden grundlegende Methoden und Themengebiete vorgestellt, auf welche die kommenden Kapitel aufbauen.

2.1 Digital Humanities

Digital Humanities wird in [The12] folgendermaßen definiert: „DH values collaboration, plurality, investigation of human culture, and the disruption of and reflection on traditional practices and is concerned with not just the use of digital technology for humanities projects but how the use of digital technology for humanities projects changes the user’s experience“. Die Anfänge der Digital Humanities liegen zu Beginn der fünfziger Jahre. Zu dieser Zeit wollte ein Jesuit namens Roberto Busa ein Verzeichnis aller in Texten vorkommenden Worte des Philosophen und Theologen Thomas von Aquinas erstellen [SSU08]. Da es so gut wie unmöglich war diese Aufgabe von Hand zu vollbringen, arbeitete er 1949 mit IBM, zu dieser Zeit noch Lochkartensortiermaschinenhersteller, zusammen. Durch die stetige Entwicklung der Informationstechnologie wuchs mit der Zeit auch der Aufgabenbereich der Digital Humanities. Heutzutage ist es schwierig das gesamte Sachgebiet der Digital Humanities aufzuzählen. Manfred Thaller nennt in [Tha17] folgende Aufgabenbereiche: Beschäftigung mit Text als solchem, Handhabung nicht textueller Medien, Texte und Bilder als Quellen von Belegen für historische Berichte und Projekte die neue Technologien an sich untersuchen.

2.2 Netzwerkgraph

Oftmals enthalten Datenmengen aus den Geisteswissenschaften Entitäten die untereinander in Relation stehen, diese Datenmengen werden auch Figurennetzwerke genannt. Die gängigste Art Figurennetzwerke zu beschreiben ist dabei der Netzwerkgraph. Ein Netzwerkgraph besteht aus Knoten und Kanten. Knoten repräsentieren in Figurennetzwerken Entitäten. Kanten stehen für Relationen zwischen den Entitäten. Die am häufigsten gewählten Darstellungsarten eines Netzwerkgraphen sind das Node-Link Diagramm und die Adjazenzmatrix. In Abbildung 2.1 ist beispielhaft ein Node-Link Diagramm und eine Adjazenzmatrix gegenübergestellt, welche denselben Netzwerkgraphen beschreiben. Ein Node-Link Diagramm beschreibt eine Form der Visualisierung eines Graphen, in dem üblicherweise Knoten als Kreise oder Vierecke mit oder ohne Beschriftungen dargestellt werden und Kanten als Linien oder Pfeile welche die Knoten miteinander verbinden. Die räumliche Position der Knoten und Kanten kann dabei unterschiedlich berechnet werden.

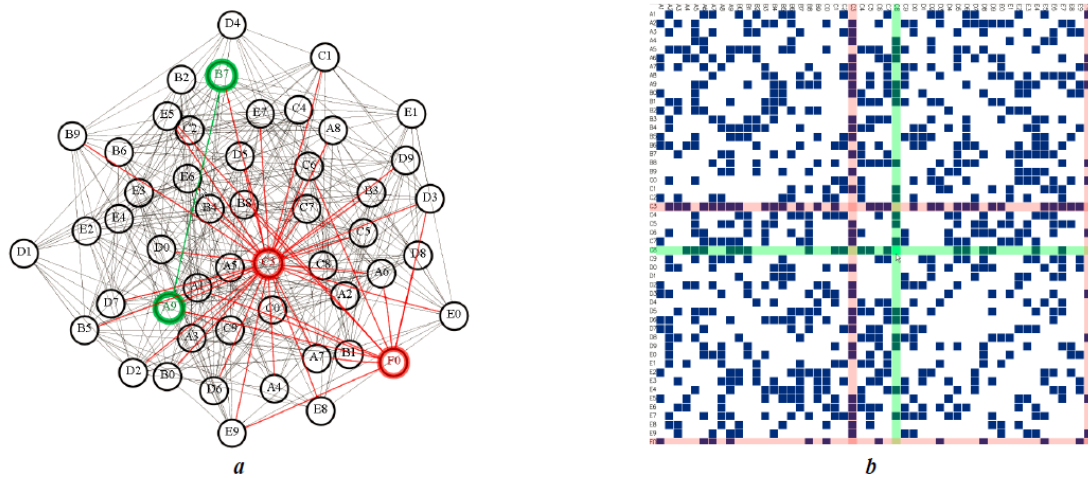


Abbildung 2.1: Ein Netzwerkgraph als Node-Link Diagramm und als Adjazenzmatrix dargestellt aus [GFC04].

Eine Adjazenzmatrix ist eine Matrixdarstellung in der es für jeden Knoten eines Graphen einen Zeilen und einen Spalteneintrag gibt. Kanten werden durch Einträge in Zellen der Matrix repräsentiert. Bei gewichteten Kanten können diese Einträge Zahlenwerte von Attributen sein, ansonsten ist die gängigste Art Wahrheitswerte zu verwenden.

2.2.1 Force-directed Graph

Bei einem Force-directed Graphen handelt es sich um eine spezielle Form des Layouts eines Node-Link Diagramms. Hierbei werden die Positionen der Knoten und Kanten des Graphen durch ein kräftebasiertes Modell berechnet. Je stärker zwei Knoten miteinander in Relation stehen, desto stärker werden diese zueinander angezogen. Dies hat zur Folge, dass Knoten welche stark miteinander in Relation stehen nah nebeneinander gezeichnet werden. Daraus folgt, dass zum Beispiel Gruppierungen von Entitäten innerhalb eines literarischen Textes leichter erkannt werden können. Auch werden durch das kräftebasierte Modell stark vernetzte Knoten näher zur Mitte der Abbildung gezeichnet, wodurch wichtige Knoten innerhalb eines Graphen schneller gefunden werden können. Darüber hinaus verhindert die Layoutform Knotenüberschneidungen, da Knoten sich stärker voneinander abstoßen je näher sie zueinander positioniert sind, was zu einer übersichtlicheren Graphdarstellung führt.

2.2.2 Vergleich zwischen Node-Link und Matrix basiertem Layout

Ghoniem et al. [GFC04] vergleicht die Lesbarkeit eines Node-Link Diagramms mit der eines Matrix basiertem Layouts. Dafür wurden in einer Studie 36 Teilnehmern sieben verschiedene Fragen gestellt, die jeweils mithilfe eines Node-Link Diagramms und einer Adjazenzmatrix beantwortet werden sollten. Die Fragen bezogen sich dabei auf die Charakteristik der Kanten, Knoten und Teilgraphen eines Netzwerkgraphen. Gemessen wurde bei jeder Frage die Zeit um diese zu beantworten, sowie die Korrektheit der Antworten.

Insgesamt wurden die Fragen mithilfe einer Matrixdarstellung deutlich effizienter und korrekter beantwortet. Eine der Hauptkenntnisse der Studie ist, dass die Matrix basierte Darstellung deutlich besser mit der Größe und Komplexität der Datenmengen skaliert. So waren bei einer Aufgabe, in der man eine Verbindung zwischen zwei Knoten in einer großen Datenmenge finden musste mit einer Matrix basierten Darstellung 92% der Antworten richtig und mit einem Node-Link Diagramm 66% der Ergebnisse korrekt. Lediglich für die Aufgabe einen Pfad zwischen zwei Knoten zu erkennen, war das Node-Link Diagramm der Matrix, außer in komplexen Datenmengen, deutlich überlegen. Ghoniem et al. empfiehlt Node-Link Diagramme nur in kleinen, einfachen Datenmengen zu verwenden, sobald die Daten größer und komplexer werden, gewinnt die Matrix deutlich die Oberhand. Für komplexere Aufgaben wie dem Finden eines Pfades zwischen zwei Knoten wird geraten zusätzliche Interaktionen bereitzustellen, um die Effektivität zu verbessern.

2.3 Visual Information-Seeking Mantra

Eine Herausforderung mit großen Datenmengen ist, dass es häufig schwierig ist diese für den Benutzer verständlich auf einem Bildschirm anzuzeigen. Mit dem Information-Seeking Mantra: „Overview first, zoom and filter, then details-on-demand“ stellte B. Shneiderman 1996 [Shn96] ein Konzept vor um diese Darstellung zu erleichtern. Zuerst soll der Benutzer einen Überblick über die gesamten Daten bekommen. Dadurch ist es schnell möglich Muster, sowie große Gemeinsamkeiten und Unterschiede der Datenmenge zu erkennen. Ebenfalls können uninteressante Datenbereiche leichter ermittelt und für die genauere Untersuchung ausgeschlossen werden. Durch eine Zoom-Funktion können einzelne Datenbereiche genauer exploriert werden, indem uninteressante Informationen außerhalb des Sichtfelds gerückt werden. Darüber hinaus ändert die Zoom-Funktion die Position und Größe der Datenelemente. Zoom kann dabei in zwei Richtungen verwendet werden. Es kann hineingezoomt werden um unwichtige Datenelementen auszublenden und im Gegensatz dazu führt Herauszoomen wieder zu einem größeren Gesamtbild der Datenmenge. Filterung erlaubt es dem Benutzer die Menge der angezeigten Daten zu modifizieren und einzuschränken. Damit können uninteressante Datenelemente aus der Visualisierung entfernt werden. Die daraus folgende Verringerung der Komplexität ermöglicht es dem Benutzer interessante Informationen leichter finden und untersuchen zu können. Häufig enthalten einzelne Datenelemente viele Nebeninformationen. Diese Informationen im Überblick anzuzeigen würde oft eine zu unübersichtliche Darstellung implizieren. Stattdessen schlägt Shneiderman vor, zusätzliche Daten erst nach einer Benutzerinteraktion ein und wieder auszublenden. Dies ermöglicht es dem Benutzer genauere Informationen zu einzelnen interessanten Datenelementen zu erhalten, welche zuvor durch Zoom und Filterung entdeckt wurden.

2.4 Das Referenzmodell für Visualisierung

Das Referenzmodell für Visualisierung von Card et al. [CMS99] beschreibt Visualisierung als Abbildung von Daten auf eine visuelle Form, mit der eine Person interagieren kann. Diese Abbildung findet dabei durch drei Zwischenschritte statt. Das Referenzmodell, zu sehen in Abbildung 2.2, beginnt mit dem Eingang von Rohdaten. Anschließend werden die Rohdaten in ein passendes Datenformat in Form von Datentabellen transformiert. Dabei werden die verschiedenen Attribute

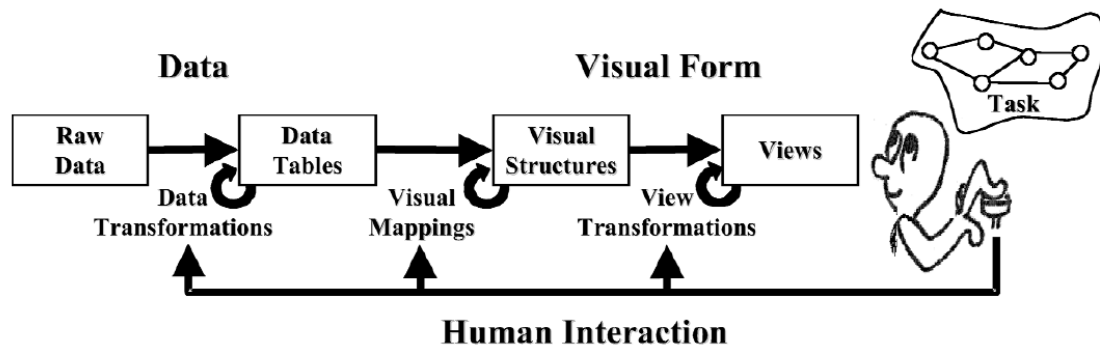


Abbildung 2.2: Das Referenzmodell für Visualisierung von Card et al. [CMS99].

der Datenelemente klassifiziert und geordnet. Liegt das passende Format vor so werden die Datenelemente anschließend in visuelle Strukturen umgewandelt. Dies ist laut Card et al. der wichtigste Schritt des Referenzmodelles. Ein Beispiel hierfür ist die Umwandlung eines Netzwerkgraphen in ein Node-Link Diagramm. Knoten werden visuellen Objekten wie zum Beispiel Kreisen oder Vierecken zugeordnet. Kanten werden zum Beispiel in Linien oder Pfeile umgewandelt. Im letzten Schritt werden anschließend die visuellen Strukturen dem Benutzer angezeigt. Der Benutzer kann nun die dargestellten Daten genauer untersuchen und analysieren. Darüber hinaus befasst sich das Modell nicht nur mit dem Erstellen einer visuellen Ansicht, sondern auch mit der Interaktion durch den Benutzer. Der Pfeil „Human Interaction“ beschreibt, dass der Benutzer in jeden Abbildungsprozess eingreifen und diesen anpassen kann. So ist es möglich die zu importierenden Rohdaten zu modifizieren, sowie auch andere Rohdaten zu importieren. Außerdem kann der Benutzer die visuelle Struktur, in der die Datenmengen angezeigt werden sollen, austauschen. So wäre es zum Beispiel möglich die Darstellung eines Kreisdiagrammes auf die eines Balkendiagrammes zu ändern. Zuletzt kann der Benutzer die Ansicht durch bereitgestellte Funktionen wie Zoom oder Filterung anpassen.

2.5 Graphenvergleich

Häufig stehen Analysten verschiedener wissenschaftlicher Bereiche, wie zum Beispiel der Neurowissenschaft, der Literaturwissenschaft oder der Bioinformatik, vor der Aufgabe zwei oder mehrere Graphen miteinander zu vergleichen. Da Graphen häufig sehr komplex sind, ist dieses Anliegen nicht immer trivial. Ziel des Vergleiches ist es meistens grundlegende Unterschiede und Gemeinsamkeiten der Graphen zu erkennen. Es können aber auch weitere Faktoren, wie der zeitliche Verlauf eines dynamischen Graphen, untersucht werden.

Gleicher et al. [GAW+11] untersucht wie zwei komplexe Objekte miteinander verglichen werden können. Ein komplexes Objekt ist hierbei eine Darstellungsart einer komplexen Datenmenge. Die Vielzahl der Vergleichsmöglichkeiten wird in drei Kategorien eingeteilt. Darüber hinaus gibt es auch hybride Darstellungsarten, welche zwei der drei verschiedenen Vergleichsarten miteinander kombinieren.

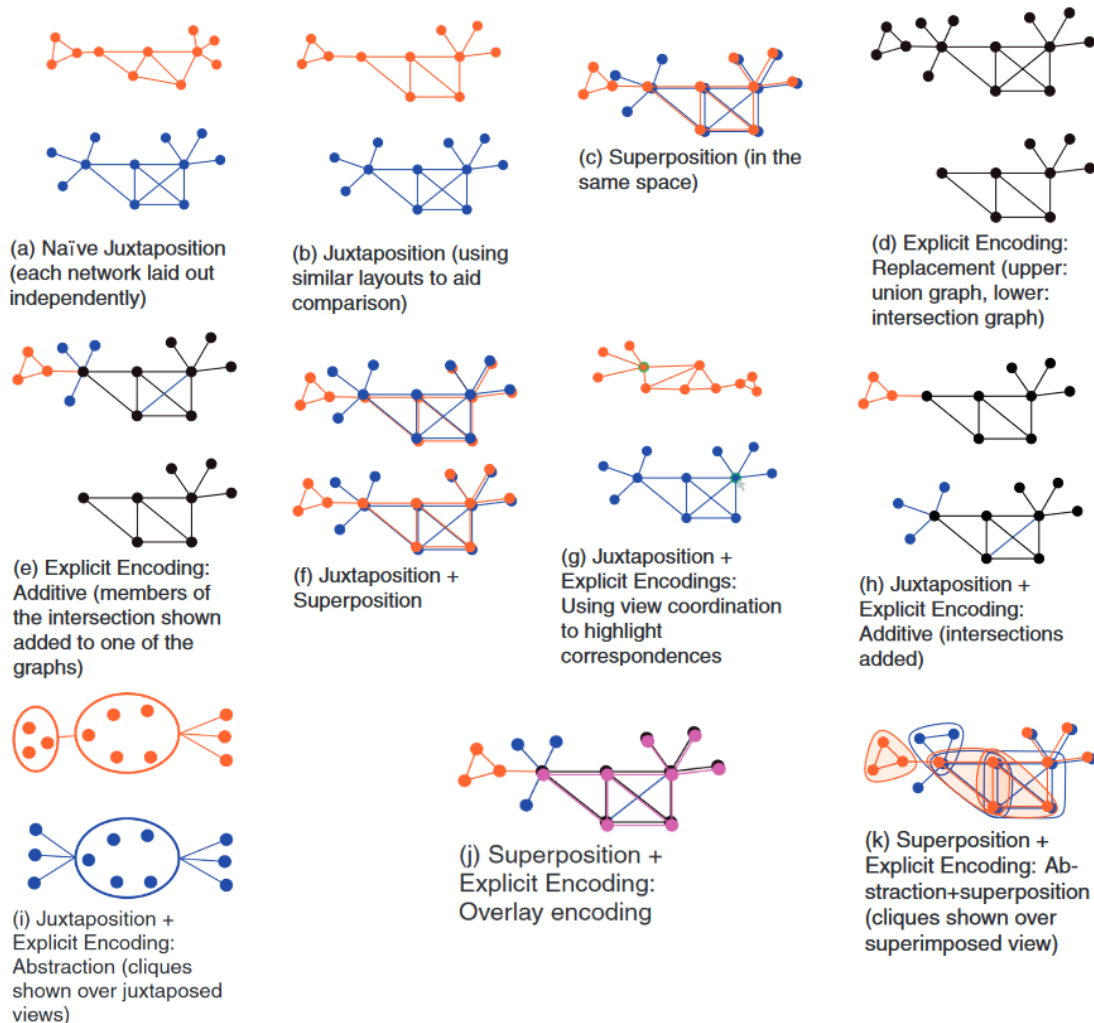


Abbildung 2.3: Beispiele der verschiedenen Vergleichskategorien anhand von zwei Netzwerkgraphen aus [GAW+11].

Ein Vergleich in Juxtaposition stellt beide Objekte separat dar. Dabei bleibt die ursprüngliche Darstellungsart beider Objekte meist vorhanden und kann sich auch zwischen den beiden unterscheiden. Die Gegenüberstellung funktioniert jedoch am besten, wenn beide Darstellungsarten Ähnlichkeiten zueinander haben. Juxtaposition ist meist einfach zu implementieren und gibt einen guten Kontext zu den gefundenen Erkenntnissen. Jedoch hängen Resultate und die Effektivität des Vergleiches stark von der Analysefähigkeit des Benutzers ab. Abschnitte von Graphen müssen im Kopf behalten werden, um diese anschließend mit einem anderen Graphen vergleichen zu können. Darüber hinaus skaliert Juxtaposition meist schlecht mit wachsender Größe und Komplexität der Datenmenge, da es für den Analysten zunehmend schwieriger wird Muster zu erkennen und den Überblick zu behalten. Ein Beispiel zur Juxtaposition anhand zweier Node-Link Diagramme ist in Abbildung 2.3 (a) zu sehen.

Im Gegensatz zu Juxtaposition erstellt Superimposition eine Darstellungsform aus beiden Objekten. Dabei werden beide Graphen auf derselben Fläche dargestellt und nicht wie in Juxtaposition

separiert. Dafür müssen beide Objekte zuvor jedoch auf eine kompatible Darstellungsform gebracht werden, um anschließend überlagert dargestellt werden zu können. Ein Beispiel hierfür ist in Abbildung 2.3 (c) zu sehen. Die Überlagerung kann unterschiedlich umgesetzt werden. Ein Ansatz besteht daraus, eines der beiden Objekte semi-transparent zu machen. Eine andere Möglichkeit ist die Objekte durch die Änderung bestimmter Darstellungsmerkmale, wie zum Beispiel der Farbe, zu unterscheiden. Welche Umsetzung gewählt wird hängt stark von der Art, Größe und Komplexität der zugrunde liegenden Datenmengen ab. Superimposition wird oft verwendet, wenn räumliche Abhängigkeiten verglichen werden sollen oder wenn die Darstellungsart beider Objekte genug Ähnlichkeiten miteinander besitzen um Unterschiede und Gemeinsamkeiten zu erkennen. Eine der Schwierigkeiten der Superimposition ist, dass durch das Übereinanderlegen beider Graphen die Darstellung noch komplexer wird.

Explicit Encoding berechnet Unterschiede bestimmter Relationen beider Objekte bereits im Voraus, diese werden anschließend graphisch dargestellt. Die Visualisierungsform kann dabei auch stark von der ursprünglichen Objekte abweichen. Damit Explicit Encoding umgesetzt werden kann, muss bereits im Voraus klar sein, welche Unterscheidungsmerkmale beachtet werden sollen. In Abbildung 2.3 (d) ist ein Netzwerkgraph zu sehen, welcher die Schnittmenge beider Grundgraphen darstellt. Einer der Vorteile von Explicit Encoding ist, dass Relationen zwischen komplexen Objekten einfacher dargestellt werden können, da viele Attribute der Datenmenge nicht berücksichtigt werden. Durch diese Vereinfachung werden die angezeigten Daten jedoch oft aus dem Kontext gerissen, weitere Beobachtungen als Unterschiede zu Relationen, die berechnet wurden können häufig nicht gemacht werden. Auch ist es schwer, dargestellte Informationen nachzuvollziehen, da die ursprünglichen Objekte nicht angezeigt werden.

Eine hybride Darstellung aus Juxtaposition und Superimposition stellt die zwei zu vergleichenden Objekte sowohl in Superimposition als auch in Juxtaposition nebeneinander dar. In Abbildung 2.3 (f) werden etwa zwei durch Superimposition visualisierte Graphen separat dargestellt. Dabei stellt jeweils ein vergleichender Graph ein anderes Objekt in den Vordergrund. Dadurch wird es erleichtert weitere Unterschiede zu finden, die Darstellung verbraucht dafür jedoch auch mehr Platz.

Die hybride Darstellung von Juxtaposition und Explicit Encoding ermöglicht es mehrere errechnete Relationen von zwei Objekten durch mehrere Graphen in Juxtaposition anzuzeigen. Dadurch können verschiedene Relationen in den Datenmengen verglichen und analysiert werden. In Abbildung 2.3 (g), (h) und (i) sind Beispiele für die hybride Darstellung von Juxtaposition und Explicit Encoding zu sehen.

Die Kombination aus Superimposition und Explicit Encoding legt mehrere Graphen in Superimposition übereinander. Einzelne Graphen können dabei errechnete Unterschiede darstellen. Die Verringerung der Komplexität der einzelnen übereinandergelegten Graphen ermöglicht es einfacher Muster in den überlagerten Daten zu finden, wie auch in zu Abbildung 2.3 (k) sehen ist. Dort werden Cliques in der überlagerten Ansicht farblich hervorgehoben.

2.6 Visualisierung von dynamischen Graphen

Eine Graphstruktur, die sich über die Zeit hinweg verändert wird dynamischer Graph genannt [BBD09]. Einen dynamischen Graphen anzuzeigen bietet neben den üblichen Herausforderung der

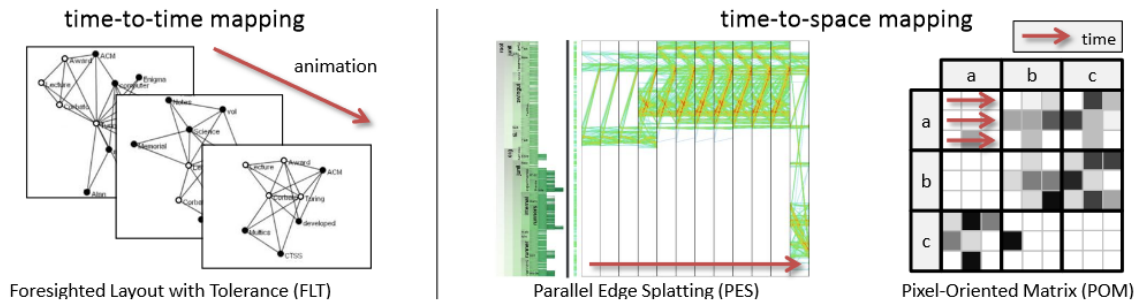


Abbildung 2.4: Time-to-time Mapping und Time-to-space Mapping gegenübergestellt [BBD13].

Graphvisualisierung die weitere Schwierigkeit, den Zeitfaktor mit in die Darstellung zu integrieren. Nach Beck et al [BBD13] [BBDW14] existieren zwei Hauptansätze um einen dynamischen Graph zu visualisieren.

Time-to-time Mapping stellt den dynamischen Graphen animiert über den Ablauf der Zeit dar. Dafür werden einzelne statische Graphen zu bestimmten Zeitsegmenten des dynamischen Graphen der Zeitabfolge nach angezeigt. Daraus ergibt sich eine Animation aus Bildabfolgen der verschiedenen statischen Graphen. Eine solche Animation wird in Abbildung 2.4 links bildlich dargestellt. Time-to-time Mapping wird hauptsächlich für die Darstellung in Form eines Node-Link Diagramms verwendet. Es gibt zwei Möglichkeiten die Knoten und Kanten in den einzelnen statischen Graphen zu positionieren. Dabei können für die aktuelle Darstellung des Graphen nur Daten der Vergangenheit berücksichtigt werden, auch Online-Approach genannt. Bei jedem neuen Zeitsegment müssen deswegen die Knotenpositionen des statischen Graphen neu berechnet werden. Dies hat zur Folge, dass jeder Graph den zugesicherten Platz komplett ausnutzen kann und die Knotenpositionen jedes Graphen für den aktuellen Datensatz idealerweise ausgewählt werden können. Da beim Online-Approach die zukünftigen Datenelemente nicht bekannt sind, kann es jedoch dazu kommen, dass Knoten überlappen. Auch müssen Änderungen des Benutzers in alle zukünftigen Graphen mitberechnet werden. Ein zweiter Ansatz ist, alle vergangenen sowie zukünftigen Daten mit in die Knotenpositionen einzukalkulieren, auch Offline-Approach genannt. Ein gängige Art ist dabei, zuerst einen sogenannten Supergraphen zu berechnen, welcher alle Datenelemente über den gesamten Zeitabschnitt des dynamischen Graphen beinhaltet. Anhand dieses Graphen werden die Positionen aller Knoten festgelegt. Alle statischen Graphen zu spezifischen Zeitsegmenten behalten diese berechneten Positionen bei. Ein Problem des Offline-Approach ist, dass die Lesbarkeit einzelner statischer Graphen nicht so optimal wie bei einem Online-Approach ist. Dies liegt daran, dass vor allem bei großen dynamischen Graphen über lange Zeitabschnitte, die Knotenpositionen von einzeln kleinen Graphen zu einem Zeitsegment den vollen gegebenen Platz nicht ausnutzen. Auch können die Positionen von Knoten sehr nah oder weit auseinanderliegen. Dagegen können Knoten und Kanten in den einzelnen Zeitabschnitten wiedergefunden werden, da sie ihre Position nicht ändern.

Time-to-space Mapping zeigt mehrere statische Graphen zu verschiedenen Zeitpunkten der dynamischen Datenmenge gleichzeitig an. Dabei können die einzelnen Graphen in Form einer Zeitleiste als Node-Link Diagramm nebeneinander dargestellt werden, wie in Abbildung 2.4 mittig zu sehen ist. Während die Position der Diagramme in Juxtaposition eine intuitive Lösung darstellt, können die Graphen auch überlagert in Superimposition, mit einem integrierten Zeitstrahl oder in einer hybriden Darstellung aus zwei der drei vorher genannten Möglichkeiten angezeigt werden.

Im Gegensatz zu Time-to-time Mapping findet Time-to-space Mapping auch viel Anwendung mit Adjazenzmatrizen. So kann der Zeitverlauf eines Datenattributes innerhalb der einzelnen Zellen der Matrix dargestellt werden, wie in Abbildung 2.4 rechts zu sehen ist. Darüber hinaus können aber auch mehrere Matrizen zu spezifischen Zeitsegmenten nebeneinander in Juxtaposition oder auf einem Zeitstrahl angezeigt werden.

3 Verwandte Arbeiten

In diesem Kapitel werden einige wichtige verwandte Arbeiten zur Darstellung und dem Vergleich von Graphen vorgestellt. Dabei wird sich auf die Visualisierung und den Vergleich von Netzwerkgraphen als Node-Link Diagramm oder Adjazenzmatrix konzentriert.

3.1 Verwandte Arbeiten zum Graphenvergleich anhand eines Node-Link Diagramms oder anhand einer Adjazenzmatrix

Ein häufig gewählter Ansatz um den Vergleich von Graphen zu vereinfachen ist, die verschiedenen Graphen zu einem Graph in Superimposition zusammenzufassen. In [AWW09] wird mit dem Semantic Graph Visualiser (SGV) von Andrews et al. ein Programm vorgestellt, welches aus zwei zueinander ähnlichen Graphen einen einzelnen fusionierten Graph erstellt. Dabei wird in sechs Schritten vorgegangen. Zuerst werden zwei Graphen G_1 und G_2 an den Semantic Graph

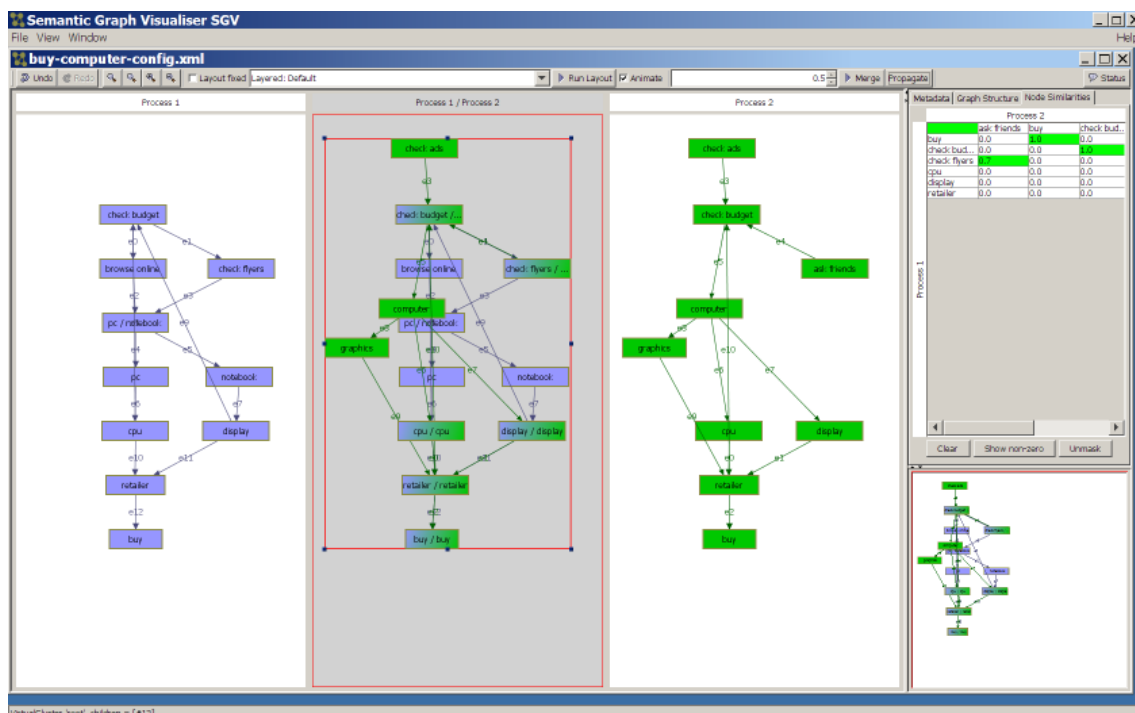


Abbildung 3.1: Die Darstellung zweier Graphen als Node-Link Diagram links und rechts, so wie mittig eines vergleichenden Graphen im Semantic Graph Visualiser von Andrews et al. [AWW09].

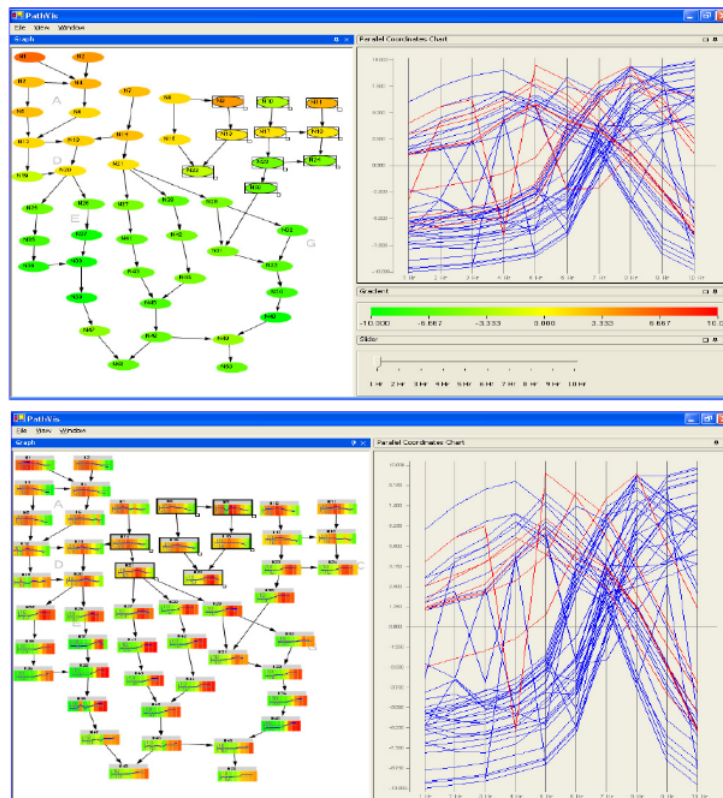


Abbildung 3.2: Die Darstellung zweier dynamischer Graphen durch Time-to-time Mapping oben und Time-to-Space Mapping unten von Purvi et al. [SLN05].

Visualiser übergeben. Anschließend werden beide Graphen nach gleichen Knoten untersucht, dabei kann jeweils nur ein Knoten aus G_1 zu einem Knoten aus G_2 passen und umgekehrt. Wurden alle gleichen Knoten gefunden wird nun der Vergleichsgraph erstellt und anschließend dargestellt. Dabei wird die Position gleicher Knoten in G_1 und G_2 angepasst, so dass diese an gleicher Stelle im jeweiligen Node-Link Diagramm sind. Zuletzt kann der Benutzer die Graphen manuell bearbeiten, indem er Knotenpositionen oder Beschriftungen ändert, dies ermöglicht ihm die Graphen für seine Forschungszwecke individuell anzupassen. In Abbildung 3.1 ist die Graphische Benutzeroberfläche des Semantic Graph Visualiser zu sehen. Links wird in grün G_1 und rechts in lila G_2 dargestellt. In der Mitte ist der Vergleichsgraph zu sehen. Zusammengefasste Knoten sind jeweils zur Hälfte in den Farben grün und lila gezeichnet. Knoten die jeweils nur in einem der beiden Graphen vorkommen werden in der Farbe des jeweiligen Graphen angezeigt.

Oftmals sind Datenmengen multidimensional. Auch ist Zeit unter anderem in der Bioinformatik häufig ein entscheidender Faktor. Viele Datenmengen werden über einen Zeitraum gesammelt, wie zum Beispiel der Ablauf einer viralen Infektion. Anhand dieser Daten kann ein dynamischer Graph erstellt werden. Wichtige Informationen offenbaren sich durch den Vergleich zweier dynamischer Graphen über verschiedene Zeiträume. Wie in Abschnitt 2.6 vorgestellt, können dynamische Graphen anhand von Time-to-time Mapping oder Time-to-space Mapping dargestellt werden. Von Purvi et al. werden in [SLN05] drei Visualisierungsarten vorgestellt um multidimensionale dynamische Graphen zu vergleichen. Während der erste Ansatz Time-to-time Mapping

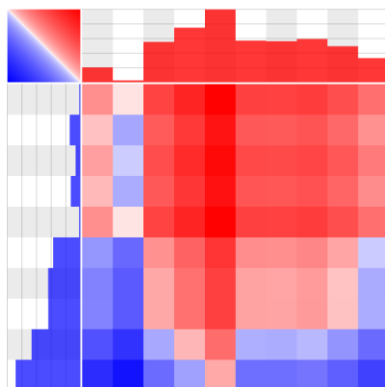


Abbildung 3.3: Die von Beck et al [BBW16] erstellte Matrix um die Leistung zweier Sportler zu vergleichen.

verwendet um zwei Graphen durch Explicit Encoding zu vergleichen, stellen die letzten zwei Ansätze einen vergleichenden Graphen mit Time-to-space Mapping dar. Zusätzlich wurde neben der Visualisierung des Graphen ein Diagramm gezeichnet um den Verlauf weiterer Datenattribute anzeigen zu können.

Der erste vorgestellte Ansatz besteht daraus, einen dynamischen Graphen zu einen bestimmten Zeitraum darzustellen, welcher über einen Schieberegler ausgewählt werden kann. Der Graph vergleicht dabei die Datensätze beider dynamischen Graphen. Unterschiede der Knoten werden durch eine farbliche Skalierung dargestellt. Durch das Bedienen des Schiebereglers kann der Graph mit Daten zu den verschiedenen Messzeitpunkten angesehen und verglichen werden.

Die zweite Möglichkeit besteht darin, statt des Schiebereglers viele kleine Graphen zu den verschiedenen Zeitpunkten in Juxtaposition gleichzeitig anzuzeigen.

Während in Vorschlag eins und zwei ein Knoten ein Attribut zu einem Zeitpunkt vergleicht, wird letzters der Unterschied eines Attributes über den Verlauf der Zeit in einem Knoten angezeigt. So können über eine Heatmap im Knoten des Graphen Unterschiede und der zeitliche Verlauf eines Attributes analysiert werden. Eine Gegenüberstellung von Ansatz eins und drei ist in Abbildung 3.2 zu sehen. In einer Studie wurden anschließend die Effektivität von Ansatz eins und drei miteinander verglichen, da diese häufig genutzte Darstellungsarten in der Bioinformatik sind. Dabei stellte sich heraus, dass der erste Ansatz effektiver für Aufgaben ist, welche einen oder zwei Zeitpunkte analysieren und vergleichen. Durch Knoten mit Heatmaps war es dagegen schneller möglich Datenelemente, die starke Abweichungen über den Zeitverlauf besaßen, zu finden. Auch konnten mithilfe der Heatmaps leichter mehr als zwei Zeitabschnitte verglichen werden.

Während viele Forschungsarbeiten zum Vergleich zweier Graphen mit Hilfe eines Node-Link Diagramms existieren, ist der Vergleich durch Adjazenzmatrizen ein weniger stark erforschtes Thema. Beck et al. [BBW16] vergleicht Leistungsdaten zweier Sportler, die über einen Zeitraum hinweg gemessen wurden anhand einer Matrixdarstellung, welche in Abbildung 3.3 zu sehen ist. Der Zeitverlauf gemessener Daten der einzelnen Sportler wird dabei vertikal und horizontal auf den Achsen der Adjazenzmatrix in Form eines Balkendiagramms angezeigt. Jede Zelle der Matrix stellt einen Vergleich der Datenelemente beider Sportler an einem jeweiligen Zeitpunkt dar. Der Zeitpunkt der Datenmessung des ersten Sportler wird dabei durch die Wahl der Spalte widerspiegelt. Der Zeitpunkt des Datenausschnittes des zweiten Sportlers kann durch die Wahl der Reihe ausgewählt werden. In den Zellen wird der Vergleich beider Zeitpunkte durch einen

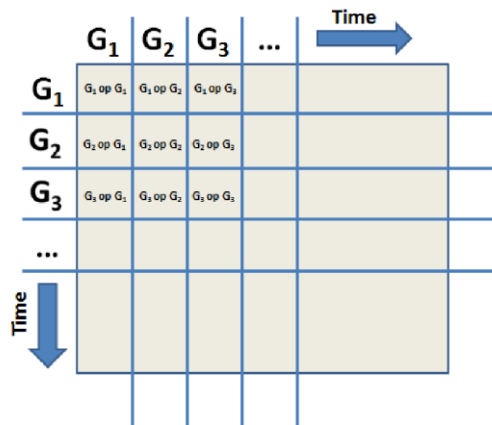


Abbildung 3.4: Anordnung um zwei dynamische Graphen in einer Matrixdarstellung zu vergleichen von Burch [Bur16].

Farbton dargestellt. Dabei wird jedem Sportler eine feste Farbe zugewiesen, in Abbildung 3.3 ist dies blau und rot. Die Farbe eines Sportlers wird mehr hervorgehoben, wenn dessen Werte höher als die des anderen Sportlers sind. Gleichgroße Werte werden weiß dargestellt.

Wie in Abschnitt 2.6 vorgestellt existieren verschiedene Möglichkeiten um einen dynamischen Graphen darzustellen. Eine Darstellungsart um zwei dynamische Graphen miteinander zu vergleichen, wird in [Bur16] vorgestellt. Als Grundlage wird hierfür im Gegensatz zu Purvi et al. eine Adjazenzmatrix benutzt. Dabei werden die dynamischen Graphen durch Time-to-Space Mapping in Juxtaposition dargestellt. Ein dynamischer Graph wird dafür vertikal und der andere horizontal auf der Matrixachse der Zeitabfolge nach abgebildet, wie in Abbildung 3.4 zu sehen ist. Pro Reihe und Spalte wird der jeweilige statische Graph zu einem anderen Zeitpunkt repräsentiert. Der Vergleich beider Graphen zu den bestimmten Zeitpunkten ist dabei in den Zellen der Matrix dargestellt. Um den Vergleich beider Graphen zu vereinfachen und übersichtlicher zu gestalten, wurden die folgenden Funktionen implementiert. Wird der Platz zu knapp um alle Teilgraphen der dynamischen Graphen anzuzeigen, so wird der zur Verfügung stehende Platz für jeden Graphen verkleinert. Somit skaliert die Darstellung auch gut mit dynamischen Graphen, welche einen langen Zeitraum repräsentieren. Auch können die Graphen in verschiedenen Darstellungsformen wie der Adjazenzmatrix oder dem Node-Link Diagramm angezeigt werden. Dadurch kann der Benutzer die für ihn passende visuelle Struktur auswählen und analysieren.

Melville et al [MGK11] stellt zwei Darstellungen vor, in welchen Datenmengen mit Resultaten verschiedener Superbowls durch eine Matrixdarstellung in Juxtaposition oder Superimposition verglichen werden, zu sehen in Abbildung 3.5. Darüber hinaus wurde eine Filterfunktion implementiert, mit welcher einzelne Graphen ein und ausgeblendet werden können. In einer anschließenden Studie wurde evaluiert, dass die in Superimposition dargestellte Matrixdarstellung deutlich genauere Ergebnisse erbrachte als die Darstellung in Juxtaposition. Darüber hinaus bevorzugten 13 von 18 Studienteilnehmern die überlagerte Darstellungsart.

Alper et al. [ABH+13] vergleicht die Verschaltung von Gehirnen mithilfe von gewichteten Graphen. Als mögliche Darstellungsart zum Vergleichen zweier neuronalen Datensätze werden dafür das Node-Link Diagramm und die Adjazenzmatrix genauer untersucht. Im Gegensatz zu Purvi et al. steht dabei der Vergleich der Kanten und nicht der Knoten im Vordergrund. Die verschiedenen

3.2 Verwandte Arbeit zur hybrid Darstellung aus Node-Link Diagramm und Adjazenzmatrix

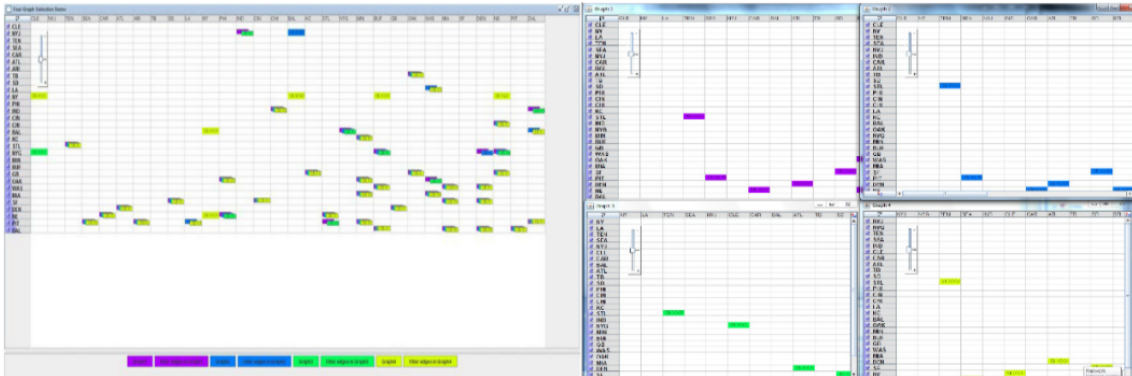


Abbildung 3.5: Vier Matrizen mit Ergebnissen eines Superbowls links in Superimposition zu einer Matrix zusammengefasst und rechts in Juxtaposition nebeneinander dargestellt von Melville et al. [MGK11].

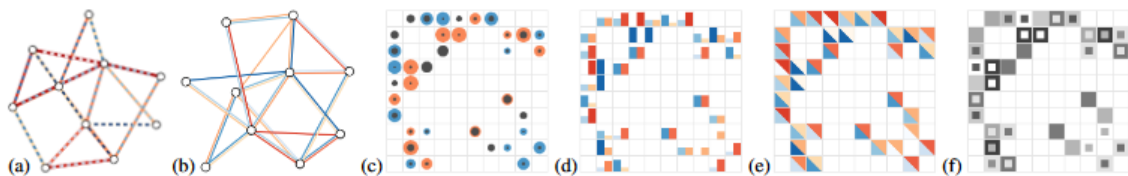


Abbildung 3.6: Die verschiedenen Darstellungsoptionen in [ABH+13] um zwei Graphen mit Hilfe einer Adjazenzmatrix oder einem Node-Link Diagramm darzustellen.

Visualisierungsoptionen sind in Abbildung 3.6 zu sehen. Für das Node-Link Diagramm werden die Kanten beider Graphen parallel nebeneinander platziert oder gestrichelt dargestellt. Für die Adjazenzmatrix werden die verschiedenen Kantengewichte zum Beispiel mit einem Balkendiagramm oder per Flächeninhalt angezeigt. Das Gewicht der Kante wird dabei immer farblich abgebildet. In einer anschließenden Studie wird das Node-Link Diagramm mit der Adjazenzmatrix verglichen. Alles in allem wurden Aufgaben mit der Adjazenzmatrix durchschnittlich 15% schneller und 20% genauer erledigt. Alper et al. empfiehlt daher Adjazenzmatrizen zu benutzen außer es handelt sich um eine sehr kleine und nicht komplexe Datenmenge.

3.2 Verwandte Arbeit zur hybrid Darstellung aus Node-Link Diagramm und Adjazenzmatrix

Eine Herausforderung der Analyse von Figurennetzwerken ist, dass Netzwerkgraphen häufig groß und komplex sind. Damit stößt die Analyse durch ein Node-Link Diagramm schnell an Grenzen, da es schwerer wird einen Überblick über die dargestellten Knoten und Kanten zu behalten. Eine Adjazenzmatrix skaliert besser mit großen Datenmengen, dafür sind Gruppierungen und Muster schwerer zu erkennen. Eine schnelle und effektive Analyse ist deshalb häufig sehr schwierig. Ein Ansatz um dieses Problem zu lösen ist NodeTrix von Henry et al. [HFM07]. NodeTrix stellt Netzwerkgraphen mit Hilfe einer hybrid Darstellung aus Node-Link Diagramm und Adjazenzmatrix dar. Wie in Abbildung 3.7 zu sehen ist werden die Vorteile beider Visualisierungsarten genutzt, indem Gruppen innerhalb der Daten als Matrizen und Beziehungen zwischen den verschiedenen

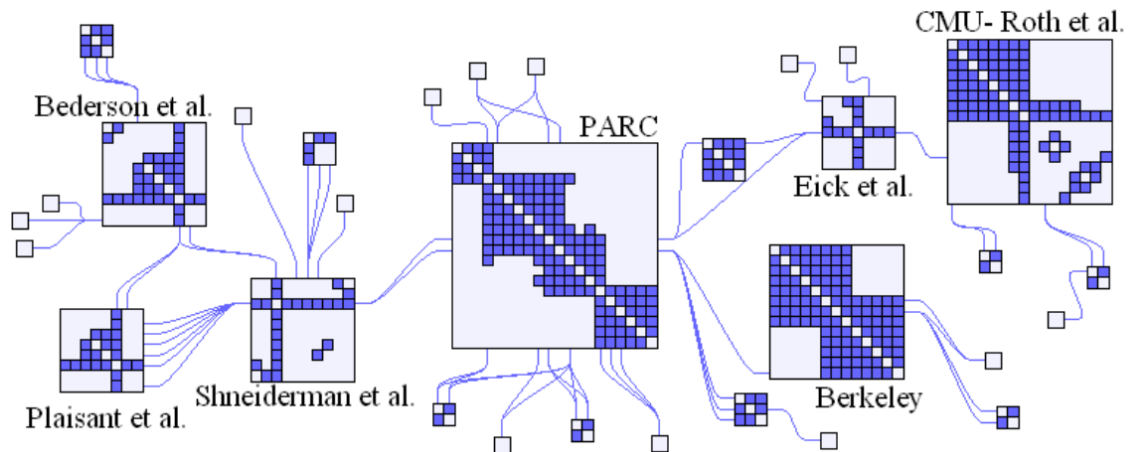


Abbildung 3.7: Die verschiedenen Darstellungsoptionen in [ABH+13] um zwei Graphen mit Hilfe einer Adjazenzmatrix oder einem Node-Link Diagramm darzustellen.

Gruppen durch ein Node-Link Diagramm dargestellt werden. Dies hat zur Folge, dass die Daten dem Benutzer übersichtlicher dargestellt werden können. Daneben werden noch mehrere Interaktionsmöglichkeiten bereitgestellt um die Darstellung besser verstehen und interpretieren zu können. So kann man Knoten und Matrizen bewegen und Entitäten in eine andere Matrix per Drag und Drop Funktion ziehen. Dies erlaubt es dem Benutzer die Layoutform so anzupassen, wie er es benötigt. Auch ist es möglich eine Auswahl von Knoten in eine Matrix umzuwandeln. Außerdem lassen sich zwei Matrizen zu einer fusionieren oder es kann auch eine Matrix in zwei Matrizen aufgeteilt werden. Dies erlaubt es dem Benutzer eigene Gruppierungen von Entitäten zu erstellen und zu analysieren.

4 Konzept

In diesem Kapitel wird das Konzept vorgestellt um Figurennetzwerke darzustellen und miteinander vergleichen zu können.

4.1 Allgemeines Vorgehen

Die Transformation der Rohdaten zu einer visuellen Repräsentation in der Anwendung soll nach den Schritten des Referenzmodell für Visualisierung, siehe Abschnitt 2.4, vonstatten gehen. Der spezifische Prozess für die zu implementierende Anwendung ist in Abbildung 4.1 zu sehen. Zu Beginn sollen Rohdaten in Form von XML-Dateien in das System importiert werden können. Dieser XML-Datensatz wird anschließend in eine interne Datenstruktur umgewandelt. Anhand dieser Datenstruktur können die Datensätze auf verschiedene visuelle Strukturen, wie die einer Adjazenzmatrix oder eines Node-Link Diagramm umgewandelt werden. Zuletzt werden die Darstellungsarten dem Benutzer auf der Benutzeroberfläche angezeigt. Wie auch von Card et al. vorgestellt, soll der Benutzer die Möglichkeit haben in die drei Prozessschritte des Referenzmodells mit eingreifen zu können. So soll es möglich sein neue Datensätze zu importieren, zwischen der visuellen Struktur einer Adjazenzmatrix und einem Node-Link Diagramm zu wechseln und dargestellte Graphen durch Zoom oder Filterung genauer zu untersuchen.

Ebenfalls orientiert sich das System an dem Visual Information-Seeking Mantra von Shneiderman et al., welches in Abschnitt 2.3 erklärt wird. So soll der Benutzer in einer Ansicht einen kompletten Überblick über alle importierten Datensätze bekommen. Anschließend soll es möglich sein, durch eine implementierte Zoomfunktion die Daten genauer zu untersuchen. Darüber hinaus sollen auch Filterfunktionen zur Verfügung stehen, um einzelne Datenabschnitte genauer untersuchen zu können. „Details on demand“, der letzte Schritt des Mantras soll sofern umgesetzt werden, dass konkrete Zahlenwerte durch Benutzerinteraktion zu einzelnen Datenelementen angezeigt werden. Insgesamt beinhalten die Datensätze, welche erstmals von der Anwendung unterstützt werden sollen, wenig zusätzliche Informationen zu einzelnen Datenelementen. Somit ist es vorerst auch nicht möglich dem Benutzer nach einer Interaktion viele Detailinformationen anzuzeigen. Es ist jedoch geplant im Falle einer Weiterentwicklung weitere Datenmengen importieren zu können, welche mehr Nebeninformationen besitzen.

4.2 Importieren der Datenmenge

In dieser Arbeit sollen Datensätze im GEXF-Format¹ importiert werden können. Das GEXF-Format ist ein XML-Datenformat welches Netzwerkstrukturen beschreibt. Eingeführt wurde

¹<https://gephi.org/gexf/format/>

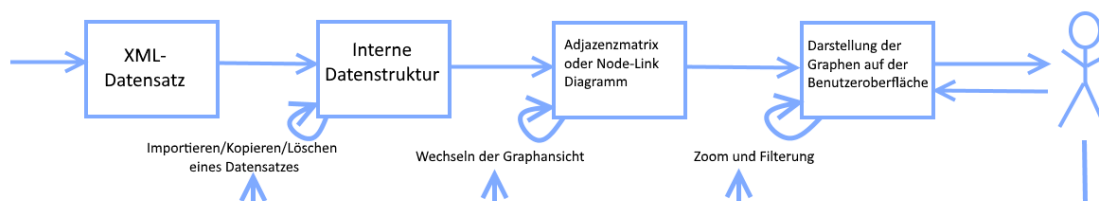


Abbildung 4.1: Der für die Anwendung konzipierte Prozess zu einer visuellen Repräsentation, nach dem Referenzmodell von Card et al. [CMS99].

das Format 2007 für das Gephi Projekt². Gephi ist ein kostenloses Open-source Programm um Graphen und Netzwerke darzustellen und zu bearbeiten. Für diese Arbeit wurden vier Dateien, welche Figurennetzwerke zum dritten und vierten Buch Parzivals modellieren bereitgestellt. Diese sollen von der späteren Anwendung unterstützt werden. Parzival ist ein Roman von Wolfram von Eschenbach welcher zwischen 1200 und 1210 nach Christus entstand und in 16 Bücher gegliedert wird. Die vier Dateien wurden von Blessing et al. [BEJR17] erstellt und analysiert. Während alle Dateien den selben Textabschnitt von Parzival beschreiben, unterscheidet sich die verwendete Textsegmentierung. Parzival ist in Strophen geschrieben, welche jeweils aus 30 Zeilen Text bestehen. Für die Segmentierung des Textes wurde in zwei der vier Dateien jede Strophe im Roman als ein Segment verwendet. In den anderen zwei Dateien wurde ein Satz als ein Segment gesetzt. Da sich die Segmentgröße zwischen den zwei Segmentierungsmethoden unterscheidet, wurde die Satzsegmentierung auf die Größe einer 30 Zeilen Segmentierung aggregiert. Außerdem berücksichtigen zwei der vier Dateien Appellative im Text. Appellative sind Substantive welche eine Entität oder eine Gruppe von Entitäten über einen Gattungsnamen beschreibt. Beispiele für Appellative sind Königin, Mutter oder Ritter.

Die vier Dateien enthalten Informationen zu Knoten und Kanten des dargestellten Figurennetzwerks. Wie in Abschnitt 2.2 bereits erwähnt, steht ein Knoten für eine Entität in Parzival und eine Kante für eine Relation zwischen zwei Entitäten. Alle der gespeicherten Attribute einer Datei sind in Tabelle 4.1 zu sehen. Durchschnittlich enthält jeder Datensatz 32 Knoten und 87 Kanten. Die Positionen aller Knoten für ein Node-Link Diagramm wurden bereits durch ein Force-directed Layout in Gephi berechnet. Alle Kanten der Figurennetzwerke sind ungerichtet. Um die Datensätze importieren zu können, muss eine systeminterne Datenstruktur bereitgestellt werden. Diese muss die Visualisierung wie auch die Filterung der Figurennetzwerke unterstützen.

4.3 Graphdarstellung

Nachdem eine Datei importiert wurde, sollen die importierten Daten dem Benutzer dargestellt werden. Zwei Darstellungsmöglichkeiten für einen Graphen sind, wie in Abschnitt 2.2 vorgestellt,

²<https://gephi.org/>

Attribut	Erklärung
Node id	Eindeutige Erkennungsnummer eines Knoten
Node label	Der Name der Entität für welchen der Knoten steht
Node spells	Alle Vorkommen der Entität im Text in Segmentangabe
Node position	Von Gephi berechnete Position des Knotens
Edge id	Eindeutige Erkennungsnummer einer Kante
Edge source	Id des Knoten von welchem aus die Kante beginnt
Edge target	Id des Knoten zu welchem die Kante führt
Edge type	Gibt an ob die Kante gerichtet oder ungerichtet ist
Edge weight	Gewicht der Kante
Edge spells	Alle Vorkommen der Relation im Text

Tabelle 4.1: Alle Attribute zu den vier gegebenen GEXF-Dateien über das Buch Parzival

das Node-Link Diagramm und die Adjazenzmatrix. Das Node-Link Diagramm ist die am häufigsten gewählte Darstellungsart um ein Figurennetzwerk darzustellen, da es einen guten Gesamtüberblick über die verschiedenen Entitäten und Beziehung innerhalb des Netzwerks vermittelt. So können interessante Relationen und Gruppierungen von Entitäten einfacher erkannt werden. Laut Ghoniem et al. [GFC04] sind jedoch Adjazenzmatrizen deutlich effizienter um explizite Fragen über das Figurennetzwerk zu beantworten. Darüber hinaus skaliert die Adjazenzmatrix deutlich besser mit der Größe und Komplexität der Datenmenge. Aus diesem Grund werden in dieser Arbeit beide Darstellungsarten unterstützt. Es soll dabei möglich sein zwischen beiden Darstellungsarten jederzeit wechseln zu können. Dies ermöglicht die Stärken des Node-Link Diagramms und der Adjazenzmatrix zu vereinen, indem ,je nach Aufgabe und Präferenz des Benutzers, die bessere visuelle Struktur gewählt werden kann.

Eine weitere gute Möglichkeit um ein Figurennetzwerk darzustellen ist die in Abschnitt 3.2 vorgestellte visuelle Struktur NodeTrix [HFM07] von Henry et al. Jedoch fehlt hierfür in den verwendeten Datensätzen eine Gruppierung der vorkommenden Entitäten. Eine Gruppierung von Entitäten ist aber für NodeTrix notwendig, da es sonst nicht möglich ist zu bestimmen welche Entitäten zusammen in einer Matrix dargestellt werden. Stattdessen müsste der Benutzer jede dargestellte Adjazenzmatrix aufwendig manuell erstellen. Sollten jedoch spätere Datensätze Informationen zur Gruppierung von Entitäten enthalten, lohnt es sich darüber nachzudenken, diese visuelle Struktur nachzuimplementieren.

Um später verschiedene Figurennetzwerke miteinander vergleichen zu können, soll es dem Benutzer möglich sein mehrere Netzwerkgraphen zu importieren und darzustellen. Damit zwischen den verschiedenen importierten Figurennetzwerke unterschieden werden kann, wird jedem Graphen eine einzigartige Farbe zugeordnet.

4.3.1 Darstellung als Node-Link Diagramm

Um das importierte Figurennetzwerk anzuzeigen, soll eine Darstellung als Node-Link Diagramm gewählt werden können. Hierfür sollen Entitäten als Kreise und Knoten als Linien dargestellt werden. Damit das Figurennetzwerk als Node-Link Diagramm dem Benutzer übersichtlich dargestellt wird, müssen sinnvolle Knotenposition berechnet werden. Eine Option für die Knotenberechnung

ist das Figurennetzwerk als Force-directed Graphen anzuzeigen. Dadurch könnten die Positionen gleicher Entitäten sich jedoch auf unterschiedlichen Graphen unterscheiden, da ein Force-Layout die Positionen der Knoten im Graphen durchgehend anhand eines physikalischen Modells neu berechnet. Außerdem wären die sich ändernden Positionen der Knoten unvorteilhaft für einen Vergleich, da bei einem Vergleich durch Juxtaposition die Positionen von Knoten gemerkt werden müssen. Deswegen ist es am sinnvollsten die Position der Knoten zu fixieren und nicht veränderbar zu machen. Wie bereits erwähnt, beinhalten die gegebenen Datensätze durch ein Force-directed Layout vorberechnete Positionen. Dabei sind die gespeicherten Positionen der Knoten, welche eine identische Entität beschreiben äquivalent, da diese durch ein Figurennetzwerk über das dritte bis vierte Buch von Parzival berechnet wurden. Da somit die gespeicherten Knotenpositionen ein sinnvolles Layout darstellen, sollen diese in der späteren Anwendung verwendet werden. Während in den gegebenen Datensätzen die Koordinaten von minimal -400 bis zu maximal 565 variieren, reicht das in der Anwendung benutzte Koordinatensystem von 0 bis zur aktuellen Größe des Containers in Pixel, in welchem der Graphen gezeichnet werden soll. Aus diesem Grund müssen die Position der Knoten auf die Größe des von der Anwendung zugesicherten Platzes skaliert werden. Da die Größe und Komplexität des gezeichneten Node-Link Diagramms nicht zu groß ist, soll das ursprüngliche Seitenverhältnis des Figurennetzwerks in der Datenmengen beibehalten werden. Dies hat den Vorteil, dass der durch das Force-directed Layout berechnete relative Abstand zwischen den Knoten erhalten bleibt.

4.3.2 Darstellungen als Adjazenzmatrix

Neben der Darstellung als Node-Link Diagramm, soll das Figurennetzwerk auch als Adjazenzmatrix visualisiert werden können. Das im folgenden vorgestellte Konzept orientiert sich an der von M. Bostock vorgestellten Matrixdarstellung, um das Figurennetzwerk des Romanes „Les Misérables“ darzustellen³.

In der Matrixdarstellung sollen alle Namen der Entitäten als Beschriftung der vertikalen und horizontalen Matrix-Achse verwendet werden. Ein erster Prototyp der Matrix ist in Abbildung 4.2 zu sehen. In jeder Zelle der Matrix wird dargestellt, wie oft die Entitäten der spezifischen Spalte und Zeile miteinander in Relation stehen. Um leichter einen Überblick über alle Relationen der verschiedenen Entitäten zu bekommen, wird die relative Quantität der Relationen farblich dargestellt. Je weniger häufig beide Entitäten in Relation stehen umso transparenter wird die Farbe. Während durch die Farbdarstellung zwar leicht Unterschiede erkannt werden können, bleibt jedoch die exakte Anzahl der Relationen unbekannt. Aus diesem Grund soll dem Benutzer durch eine Interaktion mit einer Zelle die Anzahl der Relationen von Entitäten als Zahl in der jeweiligen Zelle dargestellt werden. Wie bereits im Node-Link Diagramm soll eine Zoomfunktion die Analyse der Matrix vereinfachen. Hierfür soll ein simpler geometrischer Zoom verwendet werden, welcher alle Elemente der Adjazenzmatrix vergrößert beziehungsweise verkleinert darstellt.

³<https://bost.ocks.org/mike/miserables/>

	Entität 1	Entität 2	Entität 3	Entität 4
Entität 1			4 ↗	
Entität 2				
Entität 3				
Entität 4				

Abbildung 4.2: Entwurf einer Adjazenzmatrix um ein Figurennetzwerk darzustellen.

4.3.3 Filterung eines Figurennetzwerkes nach einem Textbereich

Entitäten sowie Relationen zwischen Entitäten eines importierten Figurennetzwerks sind an jene Textsegmente gebunden, in denen sie genannt werden. Dies hat zur Folge, dass Figurennetzwerke sich über den Verlauf eines literarischen Text ändern. Ein Figurennetzwerk gleicht deshalb dem in Abschnitt 2.6 vorgestellten dynamischen Netzwerkgraph. Oftmals sind Figurennetzwerke über Ausschnitte des Textes, wie zum Beispiel Kapitel, für Analysten sehr interessant. Um dies zu ermöglichen, soll der zu berücksichtigende Textbereich von dem Benutzer ausgewählt werden können. Ein Textbereich geht dabei von einem bestimmten Startsegment bis zu einem ausgewählten Endsegment. Wurde der Textbereich geändert, wird die Datenmenge des Figurennetzwerks auf Datenelemente, welche innerhalb dieses Bereiches im Text auftreten gefiltert. Anschließend muss die visuelle Darstellung als NodeLink-Diagramm beziehungsweise als Adjazenzmatrix anhand der gefilterten Datenmenge erneut berechnet und dargestellt werden. Damit die Knoten des Node-Link Diagramms die gleichen Positionen für jeden ausgewählten Textabschnitt behalten, werden die bestimmten relativen Positionen der Knoten des Netzwerkgraphen über den gesamten Textabschnitt beibehalten. Auch die Adjazenzmatrix soll nach Textbereichen gefiltert werden können. Wird eine Entität in der gefilterten Datenmenge nicht mehr genannt, so soll deren Spalte und Zeile aus der Adjazenzmatrix entfernt werden. Neben der Filterung kann durch die Änderung des Textbereiches wie im Time-to-time Mapping, siehe Abschnitt 2.6, der Verlauf eines Figurennetzwerkes über den Text als eine Art Animation verfolgt werden.

4.3.4 Platzierung mehrerer Graphen auf der Benutzeroberfläche

In der Anwendung soll es möglich sein, mehrere Figurennetzwerke gleichzeitig anzuzeigen. Die Figurennetzwerke können dabei auf verschiedenen Datensätzen basieren, es soll aber auch möglich sein eine Kopie eines bereits importierten Figurennetzwerks zu erstellen. Das Kopieren eines Netzwerkgraphen bringt mehrere Vorteile mit sich. So ist es möglich verschiedene Textbereiche

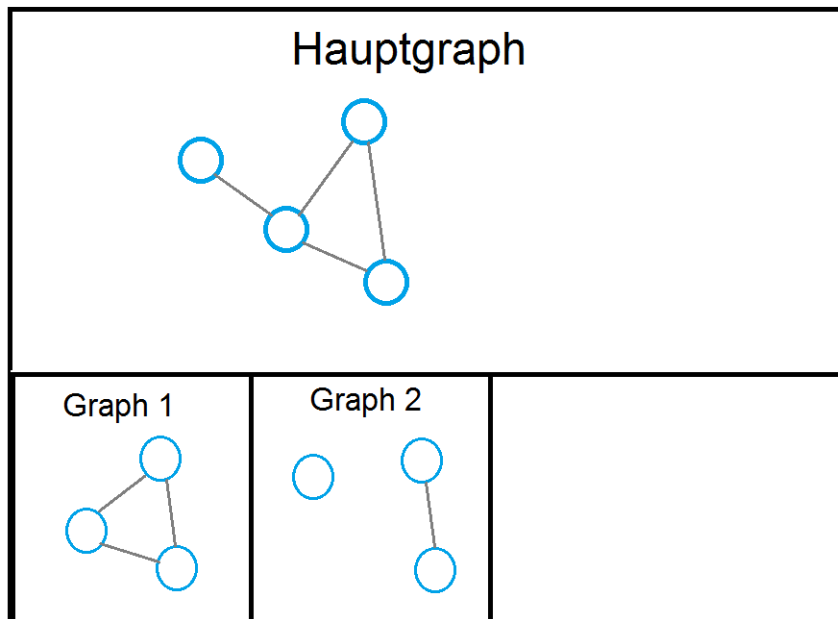


Abbildung 4.3: Die Anordnung der Graphdarstellung um einen Graphen genauer untersuchen zu können und mehrere Graphen in Juxtaposition zu vergleichen.

eines dynamischen Graphen nach dem Prinzip des Time-to-Space Mapping in Juxtaposition gleichzeitig anzuzeigen. Dabei kann der Benutzer jedoch selbst auswählen welche Textbereiche ein Netzwerkgraph darstellen soll. Ebenfalls sollen neue Figurennetzwerke einfach durch das Importieren einer neuen Datenmenge hinzugefügt werden können. Da mehrere Graphen gleichzeitig dargestellt werden, ist es möglich mehrere Figurennetzwerke in Juxtaposition miteinander zu vergleichen. Die gleichzeitig Darstellung der verschiedenen Graphen soll jedoch mehr einen allgemeinen Überblick über alle importierten Datenmengen geben, als dass sie einen detaillierten Vergleich ermöglicht. Genauere Vergleich sollen durch einen vergleichende Graphen, welcher im kommenden Kapitel vorgestellt wird, möglich sein. Damit der Benutzer jeweils eines der importierten Figurennetzwerke genauer untersuchen kann, soll für einen Graphen mehr Platz einberechnet werden. Dieser Platz ist auch für den später vorgestellten vergleichenden Graphen von Nöten. Der größer gezeichnete Graph wird in dieser Arbeit auch Hauptgraph genannt. Alle anderen importierten Netzwerkgraphen sollen unter dem Hauptgraph platziert werden. Die Positionierung der Graphen ist in Abbildung 4.3 zu sehen, wird ein neuer Graph hinzugefügt, soll der aktuelle Hauptgraph rechts neben dem letzten kleinen Graph platziert werden. Das neu importierte Figurennetzwerk wird anschließend als großer Graph gezeichnet. Sollte der zugewiesene Platz für die kleinen Graphen nicht ausreichen, so sollen die übertretenden Graphen durch eine Scrollbar in das Sichtfeld gezogen werden können. Damit der Benutzer auswählen kann, welche Graphen sichtbar sind und welcher Graph als Hauptgraph dargestellt wird, sollen die Positionen aller Graphen vom Benutzer austauschbar sein.

4.4 Vergleich zweier Graphen

In diesem Abschnitt wird konzipiert wie zwei Figurennetzwerke in der Anwendung miteinander verglichen werden können. Wie in Abschnitt 4.3.4 vorgestellt, sollen Graphen in der Anwendung durch Juxtaposition gegenübergestellt werden können. Dieser Vergleich soll dabei jedoch nur einen Überblick geben und ist nicht für eine detaillierte Analyse gedacht. Stattdessen soll es dem Benutzer möglich sein, zwei dargestellte Graphen zu selektieren und diese zu einem vergleichenden Graphen, in dieser Arbeit auch Vergleichsgraph genannt, zu fusionieren. Bei einem Vergleichsgraphen handelt es sich um einen Graphen, welcher zwei Graphen durch Superimposition und Explicit Encoding vergleicht. Für Figurennetzwerke hat der Vergleich durch Superimposition und Explicit Encoding den Vorteil, dass der Benutzer direkt den Unterschied zwischen zwei Knoten beziehungsweise Relationen sieht. So ist es nicht mehr notwendig sich die Position des Knoten beziehungsweise der Zelle eines Node-Link Diagramm oder einer Adjazenzmatrix zu merken. Darüber hinaus können Muster in den Unterschieden beider Graphen leichter erkannt werden.

Dabei soll es dem Benutzer jederzeit möglich sein die Textbereiche beider zu Grunde liegenden Graphen festzulegen, um damit die zwei Netzwerkgraphen über einen jeweils bestimmten Textbereich vergleichen zu können. Wurde der Textbereich angepasst, so muss, ähnlich wie bei der Filterung eines Figurennetzwerkes, die Datenmengen der zwei grundlegenden Netzwerkgraphen neu berechnet werden. Anschließend kann der Vergleichsgraph neu gezeichnet werden. Ein Nachteil des Vergleichsgraphen ist, dass die Komplexität und der verbrauchte Platz des Graphen steigt. Besonders bei zwei groß gewählten Textabschnitten wird es meist schwierig sein, den Vergleichsgraphen auf dem gegebenen Platz übersichtlich darstellen zu können. Denn Knoten und Kanten müssen groß gezeichnet werden, um den dargestellten Vergleich sichtbar zu machen. Eine Darstellung anhand eines Node-Link Diagrammes birgt darüber hinaus die Schwierigkeiten Knotenüberschneidung zu vermeiden. Aus diesem Grund sollen weitere Filtermöglichkeiten für einen Vergleichsgraph bereitgestellt werden. Oftmals sind für den Benutzer nur Entitäten eines Vergleichsgraphen von Bedeutung, welche in beiden Graphen vorkommen. Auch kann es interessant sein, welche Entitäten oder Relationen innerhalb beider Figurennetzwerke sich zu einem gewissen Prozentsatz unterscheiden. Aus diesem Grund soll es möglich sein, einen Vergleichsgraph anhand der Unterschiede beider Graphen filtern zu können. Dabei soll der Benutzer einen minimalen und maximalen Prozentsatz auswählen können, in welchem sich die Unterschiede bewegen dürfen. Ein Unterschied der Knoten in Höhe von 100% bedeutet zum Beispiel, dass die Entität für welche der Knoten steht, nur in einem der zwei Datensätze vorkommt. Im Gegensatz dazu bedeutet ein Unterschied der Knoten von 0%, dass die Entität in beiden Datenmengen gleich häufig erscheint. Neben den Knoten sollen auf diese Weise auch die Kanten des Vergleichsgraphen gefiltert werden können.

Wie auch für einen Netzwerkgraphen werden zwei visuelle Strukturen genauer untersucht um einen Vergleichsgraphen anzeigen zu können. So kann der Vergleichsgraph wie in Abschnitt 3.1 vorgestellt als Node-Link Diagramm dargestellt werden. Es ist jedoch auch möglich eine Darstellung als Adjazenzmatrix zu wählen. Meiner Meinung nach bringen beide Darstellungsarten Vor- und Nachteile mit sich. Ein Node-Link Diagramm kann leicht Unterschiede zwischen Knoten vermitteln. Der Vergleich zweier Relationen ist dagegen schwerer darzustellen, da je breiter eine Kante ist desto dichter wird das letztendlich dargestellte Diagramm. Darüber hinaus überschneiden sich Kanten in einem Node-Link Diagramm häufig, wodurch nicht immer die gesamte Kante

eines Node-Link Diagramm zu sehen ist. Im Gegensatz dazu ist es leicht möglich anhand einer Adjazenzmatrix entweder Entitäten oder Relation eines Figurennetzwerk in den verschiedenen Zellen zu vergleichen. Beide Unterschiede gleichzeitig übersichtlich darzustellen ist dagegen schwerer umsetzbar. In dieser Arbeit soll es deswegen möglich sein, einen Vergleichsgraphen als Adjazenzmatrix sowie auch als Node-Link Diagramm darstellen zu können. Die Darstellung als Adjazenzmatrix soll sich dabei auf den Unterschied zwischen Relationen und die Darstellung als Node-Link Diagramm auf die Unterschiede von Knoten konzentrieren. Dabei soll der Benutzer jederzeit zwischen beiden visuellen Strukturen wechseln können. In den folgenden zwei Abschnitten wird untersucht wie beide Darstellungsarten umgesetzt werden können.

4.4.1 Vergleich anhand eines Node-Link Diagramms

Um einen Vergleichsgraphen als Node-Link Diagramm darstellen zu können, muss durchdacht werden, wie Relationen anhand einer Kante und Entitäten anhand eines Knotens verglichen werden können. Damit die Darstellung übersichtlich bleibt, muss darauf geachtet werden, dass die Knoten und Kanten des Diagrammes nicht häufig kollidieren. Wie bereits erwähnt soll sich das vergleichende Node-Link Diagramm auf den Vergleich zwischen den Entitäten zweier Figurennetzwerke konzentrieren. Dennoch werden auch Kantendarstellung konzipiert, um durch die Darstellung einen besseren Überblick zu erlangen. Für die Bestimmung der Knotenpositionen des vergleichenden Node-Link Diagramms werden, wie bei einem einfachen Node-Link Diagramm, die in der Datenmenge gespeicherten Positionen verwendet und anschließend auf die Größe der Zeichenfläche skaliert. Die Darstellung eines Vergleichsgraphen verbraucht allgemein mehr Platz als die eines Netzwerkgraphen, da die Knoten- und Kantendarstellung deutlich mehr Platz verbraucht. Aus diesem Grund soll bei der Skalierung der Knotenpositionen das ursprüngliche Seitenverhältnis des Node-Link Diagramms nicht mehr berücksichtigt werden. Stattdessen soll der gesamte zugesicherte Platz ausgenutzt werden. Da je nach Knotendarstellungsart der Durchmesser der Knoten deutlich ansteigen kann, müssen darüber hinaus Knoten, die mit einem anderen Knoten kollidieren, verschoben werden. In Abbildung 4.4 werden verschiedene Mock-Ups zur Visualisierung eines Vergleichsgraphen durch ein Node-Link Diagramm vorgestellt. Im Folgenden werden die verschiedenen Kanten und Knotendarstellung untersucht und analysiert.

Kantendarstellungen des vergleichenden Node-Link Diagramm

Ziel der Kantendarstellung ist es, einen relativen Vergleich der Relationen beider Graphen zu schaffen. In Abbildung 4.4 a) und b) werden die Kanten beider Graphen parallel nebeneinander dargestellt und somit zu einer Kante zusammengefasst. Der zugehörige Graph einer Teilkante wird dabei per Farbe codiert. Je breiter eine Teilkante ist desto häufiger kommt die Relation im zugehörigen Graphen vor. Diese Kantendarstellung ermöglicht es leicht den ungefähren Anteil der Relation beider Graphen zu erkennen, auch wenn einzelne Kantenabschnitte verdeckt sind. Insgesamt ist es jedoch schwerer einen genauen prozentualen Unterschied herauszulesen, da die gesamte Kantenbreite nicht sehr groß ist. Dadurch unterscheidet sich die Breite beider Teilkanten häufig nur gering. Jedoch kann ein Unterschied von wenigen Millimeter bereits einen hohen prozentualen Unterschied bedeuten.

Eine weitere Möglichkeit um zwei Kanten zu vergleichen wird in Abbildung 4.4 c) gezeigt. Hier wird der Anteil eines Graphen durch den Längensanteil der Kante dargestellt. Die ursprünglichen

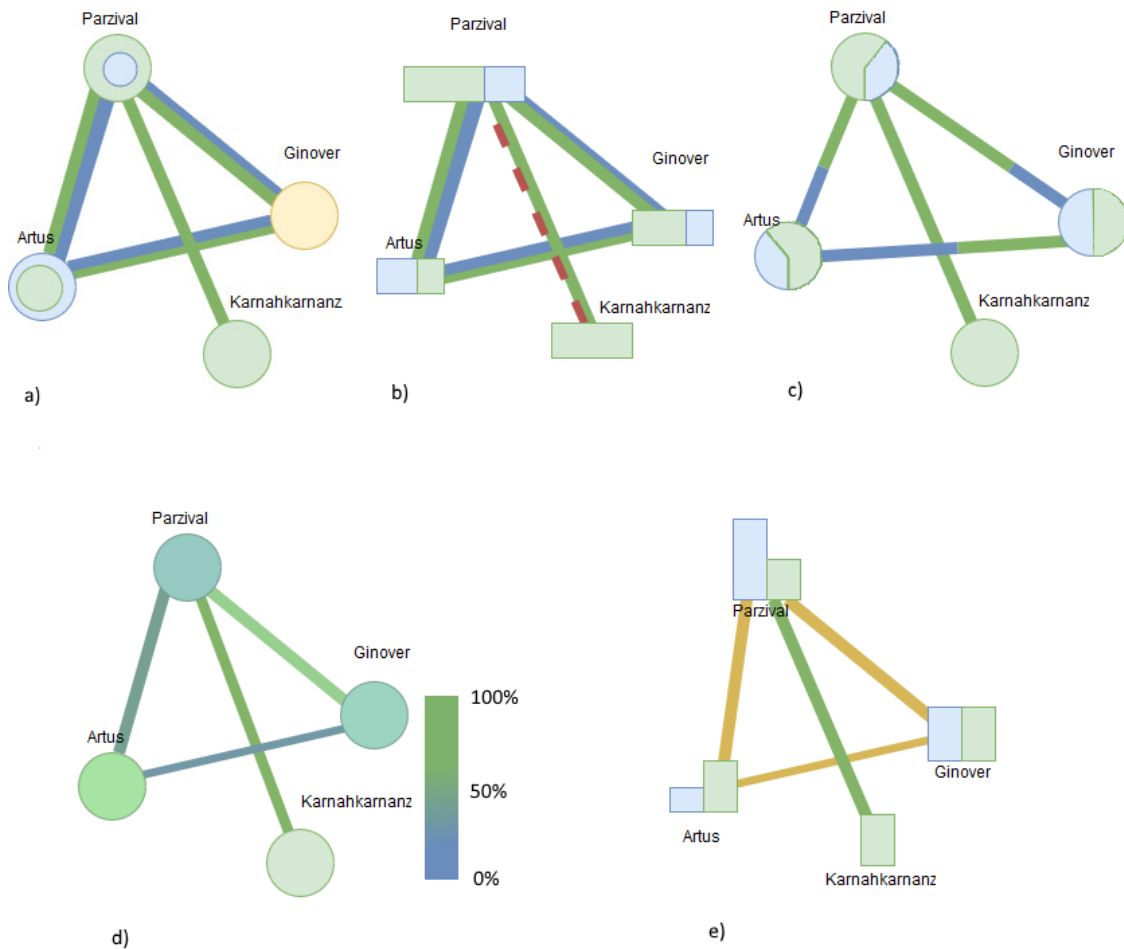


Abbildung 4.4: Verschiedene Entwürfe von Knoten- und Kantendarstellungen eines verglichenen Node-Link Diagramms.

Graphen werden dabei wieder über ihre jeweilige Farbe gekennzeichnet. Besitzt zum Beispiel Graph A vier Relationen zwischen Entität E1 und E2 und Graph B besitzt zwei, so wird zwei Drittel der Länge der Kante in der Farbe von Graph A und ein Drittel in der Farbe von Graph B angezeigt. Vorteil der Kantendarstellung ist, dass die Kantenbreite relativ dünn bleiben kann, da sie keinen Einfluss auf den Vergleich hat. Darüber hinaus kann der Unterschied beider Graphen leichter als beim Vergleich über die Kantenbreite erkannt werden, da die Kantenlänge deutlich größer als die Kantenbreite ist. Ein Nachteil ist, dass die gesamte Kantenlänge keine Aussage besitzt, da diese immer durch den Abstand zum jeweiligen Knoten entsteht. Dies kann zu Verwirrung führen, da eine Relation welche durch eine kurze Kante dargestellt wird häufiger in beiden Graphen vorkommen kann, als die Relation einer langen Kante. Darüber hinaus kann eine Überlagerung von Kanten den Vergleich für den Benutzer erschweren.

Zuletzt können Unterschiede zwischen den zu vergleichenden Graphen farblich durch Explicit Encoding dargestellt werden (siehe Abschnitt 2.5). In Abbildung 4.4 d) wird ein Farbverlauf zwischen den Farben beider Graphen erstellt. Der Unterschied wird anschließend durch die Wahl

der Farbe aus dem Farbverlauf ähnlich wie in [BBW16] dargestellt. Ist man mit den Farbverläufen vertraut, so ist dies ein guter Ansatz um die Kanten zweier Graphen schnell miteinander zu vergleichen. Jedoch muss man ansonsten häufig die Farbe der Kante auf der mit dargestellten Farbskala suchen um auf den prozentualen Unterschied zu kommen. Statt eines Farbverlaufes können auch drei verschiedene Kantenfarben, wie in Abbildung 4.4 e) verwendet werden. Die Farbe eines Graphen bedeutet, dass die dargestellte Relation häufiger in diesem Graphen vorkommt. Ist die Kantenfarbe gelb, so kommt die Relation in beiden Graphen gleich oft vor. Der Nachteil hierbei ist jedoch, dass Informationen über die Anteile beider Graphen verloren gehen. Zuletzt kann wie in Abbildung 4.4 b) das Fehlen einer Kante in einem Graphen farblich hervorgehoben werden, um den Analysten auf Unterschiede aufmerksam zu machen.

Knotendarstellung des vergleichenden Node-Link Diagramms

Anhand der Knotendarstellung sollen die Vorkommnisse einer Entität in zwei Graphen verglichen werden können. In Abbildung 4.4 werden verschiedene Ansätze vorgestellt um diesen Vergleich umzusetzen. Allgemein kann hier zwischen einem relativen und einem absoluten Vergleich unterschieden werden. Knoten, die absolut verglichen werden, haben je nach insgesamten Vorkommen der Entität eine unterschiedliche Gesamtgröße. Dagegen ist für relativ vergleichende Knoten, wie auch schon bei der Kantendarstellung, die Gesamtgröße immer gleich.

Die Abbildung 4.4 a) vergleicht dabei die Vorkommnisse relativ anhand des Flächeninhaltes zweier Kreise. Kommt die Entität in beiden Graphen gleich oft vor, so ändert sich die Farbe des Knotens. Diese Darstellung birgt jedoch den Nachteil, dass der Vergleich zweier Flächeninhalte eines Graphen oft schwierig ist, da der Radius sich nicht linear zum Flächeninhalt ändert. Zum Beispiel bedeutet die Hälfte des Radius eines Kreises nicht die Hälfte sondern ein Viertel des Flächeninhaltes. Der Benutzer kann dagegen schnell erkennen in welchem Graphen eine Entität häufiger vorkommt, da dessen Farbe den im Hintergrund liegenden größeren Kreis ausfüllt.

In Abbildung 4.4 b) werden die Knoten anhand des Flächeninhaltes eines Viereckes verglichen. Damit kann der Unterschied beider Graphen leicht erkannt werden. Da es sich um einen absoluten Vergleich handelt, können auch die verschiedenen Knoten des Node-Link Diagramm miteinander verglichen werden. Da die Knotendarstellung sehr breit ist, hat dies jedoch zur Folge, dass die Knoten des Diagramms mehr Kanten überschneiden. Auch erhöht sich durch die Darstellung die Wahrscheinlichkeit einer Knotenkollision.

Die Vorkommen der Entitäten anhand einer Farbskalierung, wie in Abbildung 4.4 d) vorgestellt zu vergleichen, birgt die selben Vor und Nachteile wie der Vergleich von Kanten anhand einer Farbskalierung.

Der Vergleich durch ein Kreisdiagramm in Abbildung 4.4 c) oder durch ein Balkendiagramm in Abbildung 4.4 e) verwendet bereits bekannte Darstellungsformen. Der Vorteil eines Balkendiagramms ist, dass die Knoten absolut verglichen werden. Auch können die Unterschiede durch die Höhe der Balken leicht verglichen werden. Jedoch können hohe Extremwerte dazu führen, dass nur ein Balken groß dargestellt wird, während der Rest sich nur durch wenige Millimeter unterscheidet. Das Kreisdiagramm vergleicht die Vorkommnisse einer Entität relativ. Jedoch könnte auch je nach gesamten Vorkommen der Entität der Radius des Kreises angepasst werden. Der Unterschied beider grundlegender Graphen kann über den Anteil am dargestellten Kreis erkannt werden.



Abbildung 4.5: Matrixbasierte Darstellungsansätze um zwei Graphen miteinander zu vergleichen.

4.4.2 Vergleich anhand einer Adjazenzmatrix

Mit Hilfe einer Adjazenzmatrix ist es schwierig sowohl die Relationen als auch die Entitäten zweier Figurennetzwerke gleichzeitig zu vergleichen. Es könnte ein ähnlicher Ansatz wie von Beck et al. [BBD09] gewählt werden um die Vorkommen einer Entität zu vergleichen, jedoch sollen durch die Adjazenzmatrix keine Zeitpunkte miteinander verglichen werden. Dies bedeutet, dass die Reihenfolge der Spalten beziehungsweise Zeilen der Matrix zunächst beliebig gewählt werden kann und dass das Balkendiagramm auf den Achsen der Matrix keinerlei Bedeutung hat. Um die Komplexität der Adjazenzmatrix zu senken und da Knoten bereits einfach mit dem Node-Link Diagramm verglichen werden können, sollen lediglich Relationen anhand der Adjazenzmatrix gegeneinander abgewogen werden können. In Abbildung 4.5 werden drei verschiedene Möglichkeiten vorgestellt um diese Aufgabe umzusetzen, Abbildung b) und c) wurden bereits in [ABH+13] vorgestellt.

Abbildung 4.5 a) und b) vergleicht Relationen zweier Graphen relativ anhand des Flächeninhaltes zweier Quadrate oder Kreise. Der größere Anteil füllt dabei immer den Hintergrund aus. Werden zwei Relationen gleich oft genannt, so wird die visuelle Struktur gelb gezeichnet. Abbildung 4.5 c) vergleicht zwei Graphen absolut, mithilfe von Balkendiagrammen, welche in den Zellen der Matrix dargestellt werden. Die Vor- und Nachteile gleichen den im vorherigen Kapitel vorgestellten Knotendarstellungen.

4.5 Entitätenliste

Nicht immer sind alle Entitäten für die Analyse eines Figurennetzwerkes von Bedeutung. Häufig wollen Analysten lediglich bestimmte Entitäten und deren Relationen vergleichen. Aus diesem Grund soll es möglich sein, die im Graphen dargestellten Entitäten auswählen zu können. Wird eine Entität entfernt oder hinzugefügt, sollen alle importierten Datenmengen gefiltert werden. Alle nicht selektierten Entitäten sowie deren Relationen werden dabei aus allen Datenmengen entfernt.

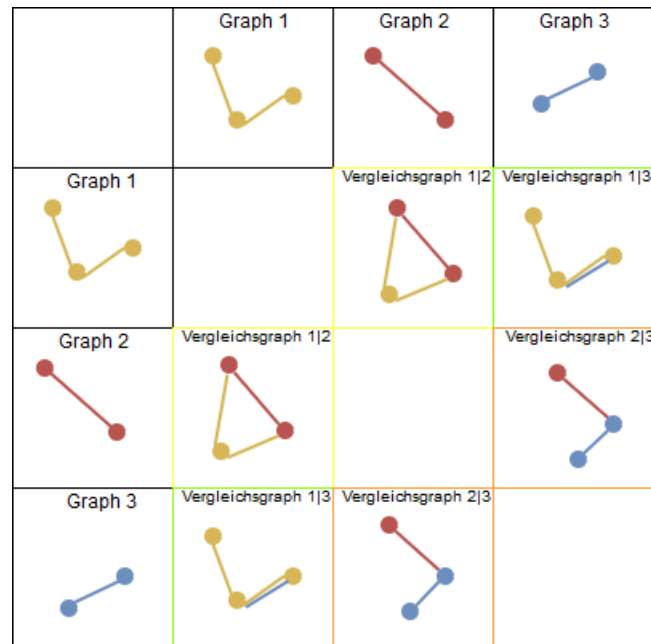


Abbildung 4.6: Die Rasteransicht gibt einen Überblick über alle erstellbaren Vergleichsgraphen.

Anschließend müssen alle Graphen anhand der aktualisierten Datenmengen gezeichnet werden. Wie auch schon für die Filterung nach Textbereichen, sollen die Knotenpositionen beibehalten und nicht neu berechnet werden.

4.6 Rasteransicht

Neben einer Übersicht über alle importierten Figurenetzwerke, soll es für den Benutzer möglich sein eine Übersicht über alle erstellbaren Vergleichsgraphen zu bekommen. Dies hat den Vorteil, dass der Benutzer schneller erkennen kann welche Vergleichsgraphen interessant sind, um sie anschließend genauer zu untersuchen. Für den Überblick sollen die verschiedenen Graphen in einer Rasteransicht, wie in Abbildung 4.6 konzipiert, dargestellt werden. Die Darstellung ähnelt dabei der von [Bur16] vorgestellten Matrix zum Vergleichen zweier dynamischen Graphen. Dafür werden alle dargestellten Graphen, welche keine Vergleichsgraphen sind, auf der vertikalen und horizontalen Achse der Matrix dargestellt. Anschließend wird in den Zellen ein Vergleichsgraph aus den Graphen der jeweiligen Spalte und Zeile erstellt. Wurden zuvor Graphen zu verschiedenen Textsegmenten zweier Datenmenge erstellt, ist es somit möglich zwei dynamische Graphen miteinander zu vergleichen. Darüber hinaus kann aber auch jeder Graph einer beliebigen Datenmenge miteinbezogen werden. Damit der Benutzer einen schnelleren Überblick über alle dargestellten Vergleiche bekommt, wird ein Unterschiedsparameter anhand der durchschnittlichen Knoten- und Kantenunterschiede für jeden Vergleichsgraphen berechnet. Der Unterschiedsparameter soll anschließend per Rahmenfarbe angezeigt werden. Alles in allem soll die Rasteransicht lediglich einen Überblick über alle Vergleichsgraphen geben. Um einen einzelnen Vergleichsgraphen genauer zu untersuchen, soll es möglich sein einen ausgewählten Graphen auch groß darzustellen.

5 Implementierung

Dieses Kapitel widmet sich der praktischen Umsetzung des zuvor vorgestellten Konzepts. Hierfür wird die implementierte Webanwendung vorgestellt und erklärt.

5.1 Verwendete Technologien

Die Webanwendung wurde mit Hilfe der Hypertext Markup Language (HTML), den Cascading Style Sheets (CSS) und Javascript entwickelt. Zusätzlich wurden folgende drei Bibliotheken verwendet:

- D3.js ¹
- JQuery ²
- JQuery UI ³

D3.js ist eine Javascript Bibliothek, welche die visuelle Darstellung von Daten in einer Webanwendung unterstützt. Mit Hilfe dieser Bibliothek wurden alle Node-Link Diagramme, Adjazenzmatrizen und die Rasteransicht implementiert. Die JQuery Bibliothek wurde genutzt, um die verschiedenen Fensteransichten zu verwalten, sowie einzelne dargestellte Element zu manipulieren. Letztlich wurde der Schieberegler der JQueryUI verwendet.

5.2 Benutzeroberfläche

Die Benutzeroberfläche der Webanwendung ist in Abbildung 5.1 zu sehen. Über die Navigationsbar kann zwischen der Graphansicht (1), der Großansicht (2) oder der Rasteransicht (3) gewählt werden. Wurde eine GEXF-Datei über den Dateibrowser (7) importiert, so wird eine Liste aller Entitäten (6) der bereits importierten Datenmengen rechts dargestellt. Die Graphübersichtsliste (5) zeigt jeden dargestellten Graphen, sowie dessen zugewiesene Versionsnummer, Dateiname und Farbe an.

¹<https://d3js.org/>

²<https://jquery.com/>

³<https://jqueryui.com/>

5 Implementierung

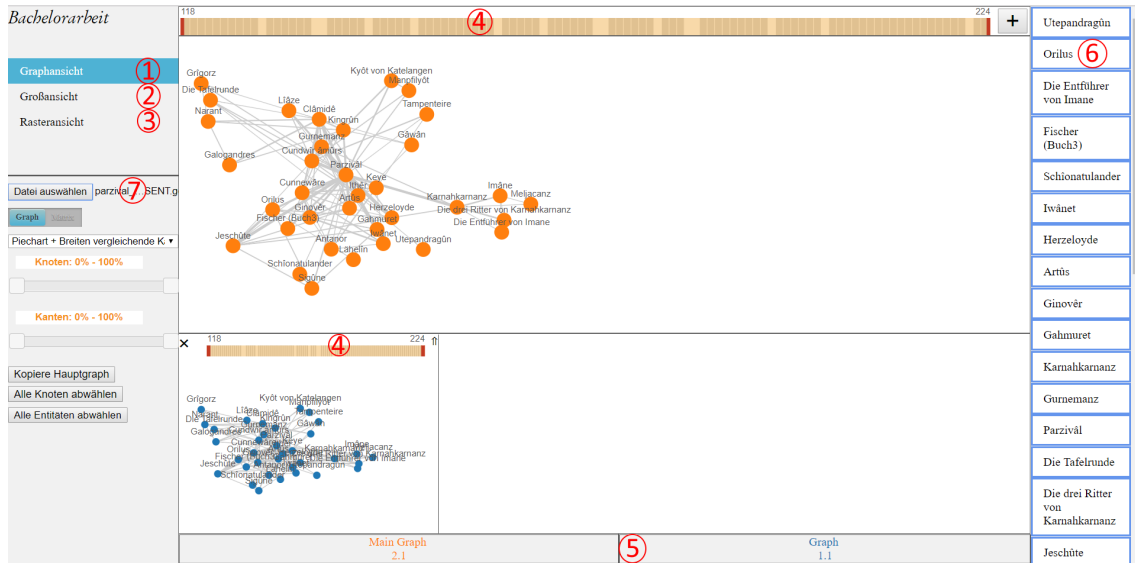


Abbildung 5.1: Benutzeroberfläche mit geöffneter Graphansicht(1).

5.2.1 Graphansicht

Alle importierten Graphen sind in der Graphenansicht, zu sehen in Abbildung 5.1, dargestellt. Dabei sind die Graphen nach der in Kapitel 4.3.4 vorgestellten Anordnung positioniert. Sollten so viele Graphen importiert werden, dass der zugesicherte Platz für die in klein dargestellten Graphen nicht mehr ausreicht, können die ausgeblendeten Graphen mit Hilfe einer Scrollbar in das Sichtfeld geschoben werden. Ebenfalls können die Plätze zweier Graphen mithilfe einer Drag und Drop Funktion getauscht werden. Damit ist es möglich, die Reihenfolge der Graphen zu ändern sowie auch einen anderen Graphen als großen Hauptgraph darzustellen. Durch einen Klick auf die „Kopiere Hauptgraph“-Schaltfläche wird der Hauptgraph kopiert und zusätzlich als kleiner Graph in einer anderen Farbe dargestellt. Über jedem Graphen wird ein zugehöriger Schieberegler (4), in dieser Arbeit auch Fingerprint genannt, angezeigt. Mit Hilfe des Schiebereglers kann der Benutzer den zugehörigen Graphen auf einen Textbereich filtern sowie einen Überblick über Nennungen von Entitäten in den Textsegmenten bekommen. Kleine Graphen können über das X-Symbol links über dem Graphen geschlossen werden. Mit Hilfe des Pfeil-Symbols rechts über dem Graphen kann ein Vergleichsgraph aus dem zugehörigen Graphen und dem Hauptgraphen erstellt werden.

5.2.2 Großansicht

Damit einzelne Figurennetzwerke und Vergleichsgraphen genauer untersucht werden können, ermöglicht es die Großansicht, zu sehen in Abbildung 5.2, alle klein dargestellten Graphen auszublenden. Durch den gewonnen Platz kann dadurch der Hauptgraph größer dargestellt werden. Alle Interaktions- und Filtermöglichkeiten bleiben dabei bestehen.

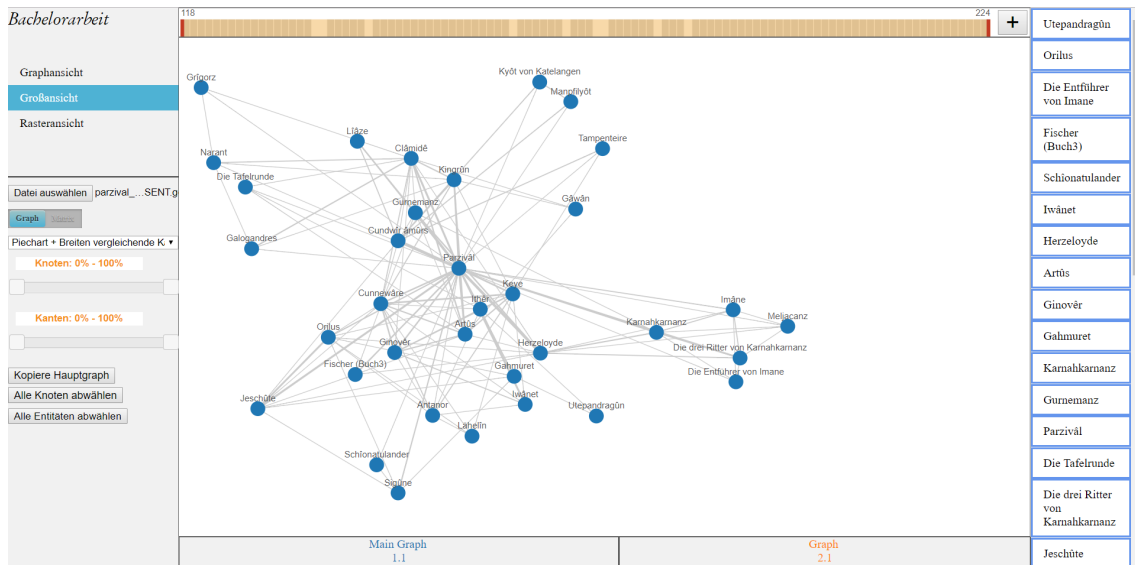


Abbildung 5.2: Ein einzelnes Node-Link Diagramm in der Großansicht dargestellt.

5.2.3 Auswahl des darzustellenden Textbereiches

Der Benutzer kann über den in Abbildung 5.1 (4) dargestellten Schieberegler einen Textbereich auswählen, über welchen ein Figurennetzwerk oder ein Vergleichsgraph gezeichnet werden soll. Über den linken roten Regler kann das minimale und über den rechten Regler kann das maximale Segment eines Textbereiches ausgewählt werden. Anhand der Beschriftung über den Reglern, kann der Benutzer nachlesen, welche Segmente zur Zeit ausgewählt sind.

Die dargestellten Vierecke im Hintergrund stehen für Textsegmente, in welchen eine Entität genannt wird. Damit soll dem Benutzer ein Überblick über die Nennungen von Entitäten im Text gegeben werden. Die aktuelle Auswahl des Textbereiches wird durch ein orangefarbenes Viereck markiert. Da Vergleichsgraphen zwei Figurennetzwerke miteinander vergleichen, können auch die Textbereiche der zwei zugrunde liegenden Graphen ausgewählt werden. Mit einem Klick auf „+“-Schaltfläche kann der Benutzer beide Schieberegler, welche jeweils für eines der Figurennetzwerke stehen, gleichzeitig anzeigen lassen.

5.2.4 Rasteransicht

Wie in Abschnitt 4.6 vorgestellt wird in der Rasteransicht eine Übersicht über alle erstellbaren Vergleichsgraphen gegeben. Die implementierte Ansicht ist in Abbildung 5.3 zu sehen. Unterschiede werden durch einen Farbverlauf von grün zu rot an den jeweiligen Rahmen der Graphen dargestellt. Grün steht dabei für sehr ähnlich und rot für sehr unterschiedlich. Die Ähnlichkeit zweier Figurennetzwerke wird durch den durchschnittlichen Unterschied zwischen den Nennungen von Relationen und dem Vorkommen von Entitäten berechnet. Jeder Graph kann durch eine Zoom-Funktion genauer untersucht werden. Da der Platz eines einzelnen Vergleichsgraphen in der Rasteransicht sehr gering ist, sind alle Knotenbeschriftungen ausgeblendet, da es

5 Implementierung

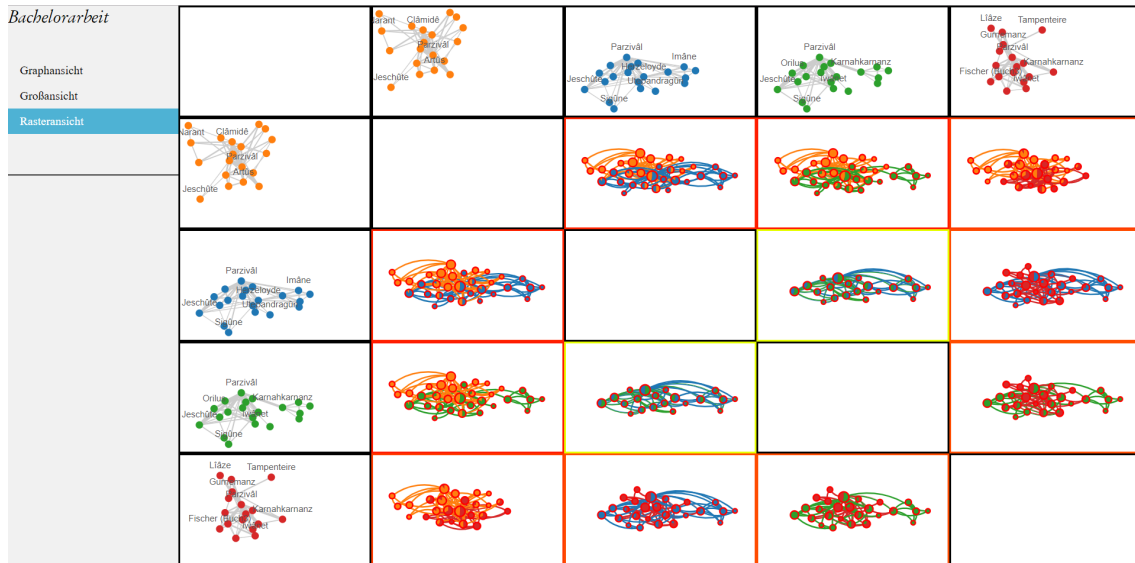


Abbildung 5.3: Die implementierte Rasteransicht, welche einen Überblick über alle erstellbaren Vergleichsgraphen gibt.

sonst zu sehr vielen Überschneidung mit anderen Beschriftungen oder Knoten kommt. Ein Doppelklick auf einen Vergleichsgraphen wechselt in die Großansicht und stellt den ausgewählten Vergleichsgraphen als Hauptgraphen dar.

5.2.5 Entitätenliste

Mit der Entitätenliste können zu berücksichtigende Entitäten aller dargestellten Figurennetzwerke, wie in Kapitel 4.5 konzipiert, ausgewählt werden. Entitäten, welche in allen Graphen dargestellt sind, sofern sie darin vorkommen, werden durch einen blauen Rahmen markiert. Nicht berücksichtigte Entitäten besitzen keinen Rahmen. Durch ein Klick auf eine Entität kann der jeweilige Zustand gewechselt werden. Wird eine Entität abgewählt wird diese aus allen dargestellten Graphen entfernt. Außerdem werden alle Relationen mit dieser Entität nicht mehr berücksichtigt. Andererseits wird durch das Selektieren einer Entität, diese wieder in allen importierten Datenmengen dargestellt. Damit es in einem vergleichendem Node-Link Diagramm zu keinen Knotenüberschneidungen mit den neu eingblendeten Knoten kommt, werden Knotenkollisionen ausfindig gemacht. Anschließend werden einzelne Knotenpositionen, falls notwendig, angepasst, damit es zu keinen Überschneidungen mehr kommt. Dies hat zur Folge, dass Knotenpositionen in verschiedenen vergleichenden Node-Link Diagrammen sich leicht unterscheiden können. Dieser Unterschied ist jedoch so gering, dass Knoten dennoch auch in anderen Graphen schnell wiedergefunden werden können.

5.2.6 Graphübersichtsliste

Da nicht immer alle Graphen gleichzeitig angezeigt werden können, gibt die Graphübersichtsliste, zu sehen in Abbildung 5.4, einen Überblick über alle importierten Graphen. Jedes neu importierte Figurennetzwerk oder jeder neu erstellte Vergleichsgraph bekommt eine neue Versionsnummer

parzival_3_4_noE_noDR_AP_noEK_SENT.gexf			
Main Graph 4.1	Graph 1.1	Graph 2.1	Graph 3.1

Abbildung 5.4: Die Graphübersichtsliste mit eingeblendeter Kurzinformation.

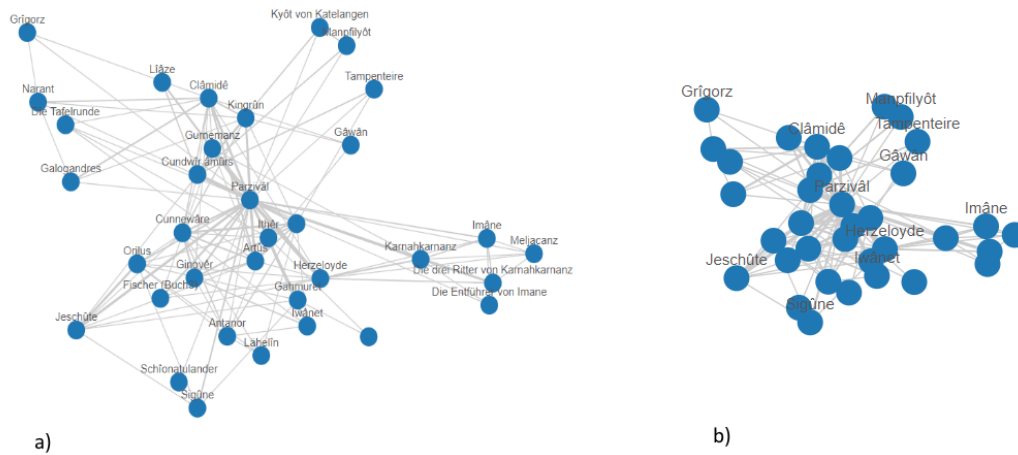


Abbildung 5.5: Ein groß dargestelltes Node-Link Diagramm, in welchem jede Knotenbeschriftung eingeblendet ist (a) und ein Node-Link Diagramm, aus welchem herausgezoomt wurde und in welchem einzelne Beschriftungen aufgrund von Überschneidungen ausgeblendet wurden (b).

zugewiesen. Kопierte Graphen unterscheiden sich durch eine unterschiedliche Subversionsnummer. Wenn die Maus über eine Graphversion gehalten wird, wird eine Kurzinformation mit dem Dateinamen der importierten Datenmenge des Graphen angezeigt. Bei Vergleichsgraphen werden die Dateinamen beider zugrunde liegender Figurennetzwerke angegeben.

5.3 Graph Visualisierung

Wie in Abschnitt 4.3 festgelegt, können die importierten Netzwerkgraphen, sowie die erstellten Vergleichsgraphen als Node-Link Diagramm und als Adjazenzmatrix dargestellt werden. Der Benutzer kann dabei die visuelle Struktur durch das Betätigen eines Schiebereglers wechseln. Jede Graphdarstellung kann durch eine Zoom-Funktion genauer untersucht werden, auch kann der gesamte Graph auf der Zeichenfläche verschoben werden. Darüber hinaus wird jedem dargestellten Graphen eine unikale Farbe zugewiesen. Die Farbauswahl eines Graphen basiert auf einer von D3.js bereitgestellten Farbpalette aus 20 Farben. Werden mehr als 20 verschiedene Farben benötigt, wird jede weitere Farbe zufällig berechnet.

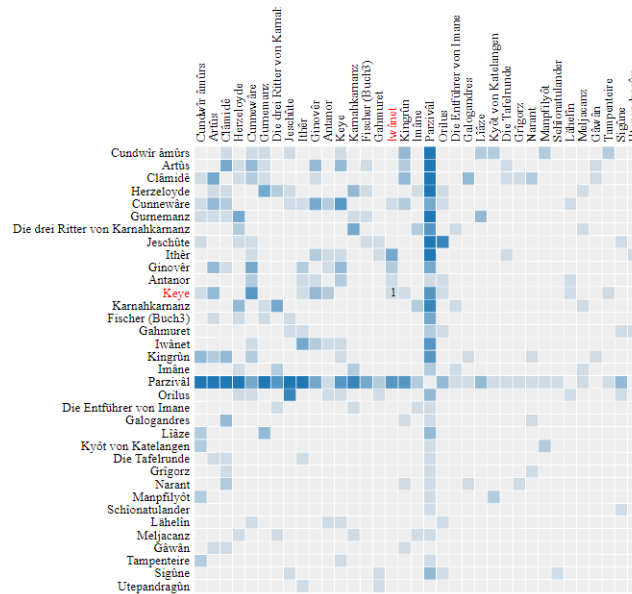


Abbildung 5.6: Die implementierte Adjazenzmatrix.

5.3.1 Node-Link Diagramm

Das implementierte Node-Link Diagramm wird in Abbildung 5.5 dargestellt. Wie in Abschnitt 4.3.1 beschlossen, bleibt das Seitenverhältnis des ursprünglichen Force-directed Graphen erhalten. Auch sind Positionen von Knoten festgelegt und einzelne Knoten können nicht verschoben werden. Um das Node-Link Diagramm genauer zu analysieren, wurde eine semantische Zoom-Funktion implementiert. Diese behält bei der Verwendung des Zooms die Kanten und Knotengröße des Node-Link Diagramms bei. Die Zoom-Funktion ermöglicht es dadurch den Graphen mehr zu entzerren, ohne dass sich die Kanten und Knotengröße verändert. Eine Herausforderung der Darstellung der Knotenbeschriftungen war, dass diese sich häufig mit anderen Beschriftungen überschneiden. Um dies zu verhindern wurde eine Kollisionserkennung der Beschriftungen implementiert. Anhand dieser werden Kollisionen mit anderen Beschriftungen erkannt, anschließend wird jene Beschriftung ausgeblendet, welche weniger häufig im Text vorkommt. Werden die Knoten und deren Beschriftungen durch die Zoom-Funktion so weit entzerrt, dass zwei Beschriftungen nicht mehr miteinander kollidieren, so werden beide im Node-Link Diagramm wieder dargestellt.

5.3.2 Adjazenzmatrix

Die Adjazenzmatrix wurde nach dem Vorbild des in Abschnitt 4.3.2 vorgestellten Prototypen implementiert. Wie in Abbildung 5.6 zu sehen ist, sind die Namen der Entitäten auf der horizontalen und vertikalen Achse dargestellt. Zellen zeigen die Häufigkeit der jeweiligen Relation anhand der Farbstärke an. Zellen von Relationen, welche nicht im Text vorkommen werden nicht ausgefüllt. Fährt man mit der Maus über eine Zelle, so wird die Anzahl der Vorkommen der Relation im Text als Zahl in der Zelle angezeigt. Außerdem werden die zwei Entitäten auf der Matrixachse, deren Relation die Zelle darstellt rot gefärbt.

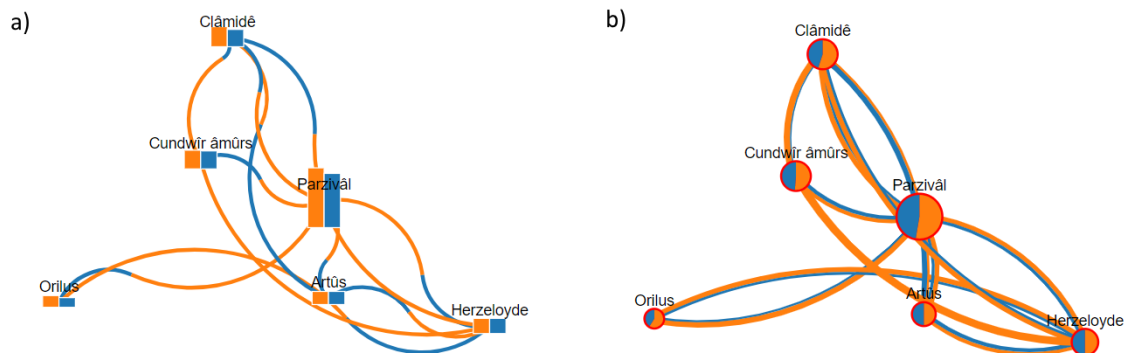


Abbildung 5.7: Vergleichsgraph zu Datensätzen von Parzival Buch drei und vier. Einmal mit Satzsegmentierung und einmal mit 30 Zeilensegmentierung. In a) werden Längenvergleichende Kanten und Knoten in Form von Balkendiagrammen dargestellt. Breitenvergleichende Kanten und Kreisdiagramme als Knoten sind in b) zu sehen.

5.3.3 Vergleichendes Node-Link Diagramm

In Abschnitt 4.4.1 wurden verschiedene Kanten und Knotendarstellungen vorgeschlagen, um ein vergleichendes Node-Link Diagramm zu erstellen. Letztendlich wurden zwei visuelle Strukturen von Knoten und Kanten in die Implementierung mit übernommen. Knoten werden als Kreis- oder Balkendiagramm visualisiert. Die visuelle Repräsentation wurde gewählt, da es sich um eine für den Anwender bekannte Darstellung handelt, welche einen einfachen Vergleich ermöglicht. Für den Kantenvergleich wurden breiten- und längenvergleichende Kanten implementiert. Alle Darstellungsarten sind in Abbildung 5.7 zu sehen.

Da der Vergleichsgraph häufig komplex und unübersichtlich ist, werden neben der Auswahl von Entitäten und der Wahl eines Textbereiches weitere Filtermethoden bereitgestellt. Oftmals sind Kanten oder Knoten von Interesse, welche sich sehr stark oder sehr wenig zwischen zwei Figurennetzwerken unterscheiden. Aus diesem Grund können durch die Bedienung zweier Schieberegler Knoten und Kanten ausgegraut werden, welche sich um einen bestimmten Prozentsatz unterscheiden. Der ausgewählte Prozentsatz bezieht sich auf den Vergleich der Vorkommen einer Relation oder einer Entität zweier Graphen in beiden Texten. Kommt eine Entität nur in einem der Graphen vor, so ist der Unterschied der Entität 100%. Kommt eine Entität oder eine Relation in beiden Graphen gleich oft vor, unterscheiden sich diese um 0%.

Um eine einzelne Entität und deren Relationen innerhalb des vergleichenden Node-Link Diagramms genauer zu erkennen, kann mit der Maus über einen Knoten im Vergleichsgraph gefahren werden. Daraufhin werden alle Knoten und Kanten, welche nicht in Relation mit dem ausgewählten Knoten stehen ausgegraut, wie in Abbildung 5.8 zu sehen ist. Für alle nicht ausgegrauten Knoten wird darüber hinaus die Anzahl der Vorkommen in den zwei Graphen, welche verglichen werden, als Zahl im Knoten angezeigt. Außerdem ist es möglich, einzelne Knoten und deren Relationen per Mausklick auszugrauen oder wieder anzuzeigen. Somit kann der Benutzer auch selbst auswählen welche Entitäten und Relationen dargestellt werden sollen. Mit der „Alle Knoten abwählen“-Schaltfläche kann der Benutzer alle Knoten und Kanten des Vergleichsgraphen ausgrauen. Anschließend kann er durch das Klicken auf einzelne Knoten selbst auswählen, welche Entitäten im Vergleichsgraphen dargestellt werden sollen. Ebenfalls unterstützt die Darstellung

5 Implementierung

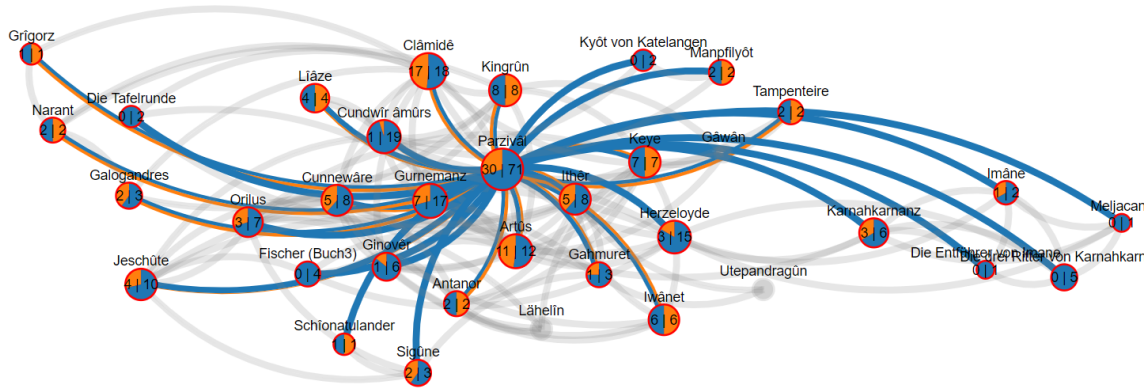


Abbildung 5.8: Durch das Halten des Mauszeigers über einen Knoten im vergleichenden Node-Link Diagramm, werden alle in Relation stehenden Knoten und Kanten hervorgehoben

eine Zoom-Funktion. Im Gegensatz zu einem Node-Link Diagramm handelt es sich dabei aber um einen einfachen geometrischen Zoom. Dieser ermöglicht es Knoten und Kanten des Vergleichsgraphen zu vergrößern, um den dargestellten Unterschied genauer untersuchen zu können.

5.3.4 Vergleichende Adjazenzmatrix

Im Unterschied zum vergleichenden Node-Link Diagramm, wurde für die Adjazenzmatrix genau einer der in Abschnitt 4.4.2 vorgestellten Vorschläge implementiert. Dies liegt daran, dass der Vergleich durch Balkendiagramme in den Zellen meiner Meinung nach den anderen zwei Darstellungsarten deutlich überlegen ist. Balkendiagramme ermöglichen es, die dargestellten Relationen in den Zellen der Matrix einfach miteinander zu vergleichen. Positioniert der Benutzer die Maus über eine Zelle der Matrix, werden die Vorkommen der Relation in den einzelnen Graphen als Zahl dargestellt. Die implementierte vergleichende Matrix ist in Abbildung 5.9 zu sehen.

Um Entitäten in der Adjazenzmatrix zu finden oder um einen besseren Überblick zu bekommen, ist es möglich die Entitäten in den Spalten und Zeilen der Matrix zu sortieren. In der Anwendung ist eine Sortierung nach Namen oder nach Vorkommen in beiden Graphen möglich. Genau wie das vergleichende Node-Link Diagramm, kann auch die Adjazenzmatrix auf Unterschiede gefiltert werden. Hierbei werden Knoten, welche nicht in den ausgewählten Unterschiedsbereich passen, nicht in der Adjazenzmatrix dargestellt.

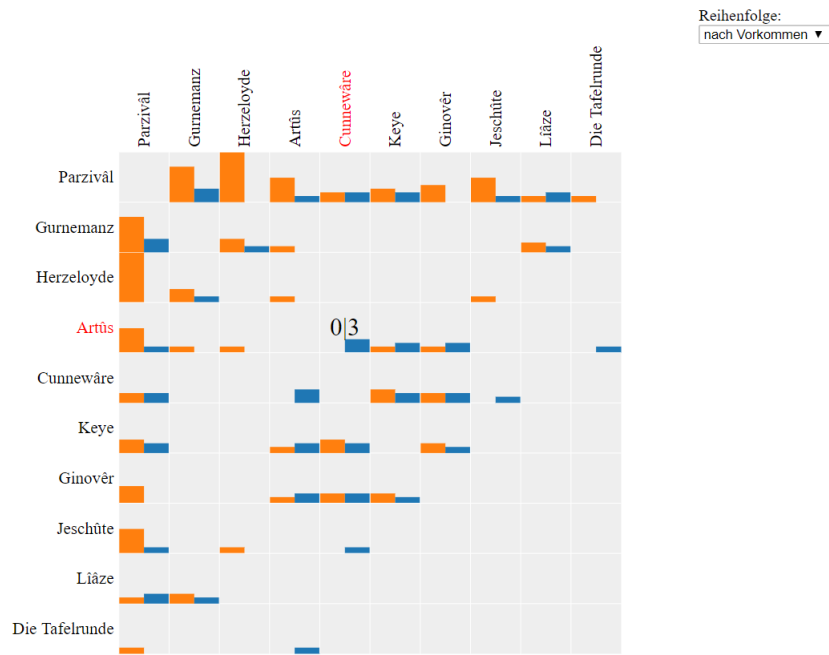


Abbildung 5.9: Die implementierte vergleichende Adjazenzmatrix nach Vorkommen der Entitäten sortiert.

6 Evaluation

In diesem Kapitel wird das implementierte System anhand eines Anwendungsfalles genauer evaluiert. Ziel des Anwendungsfalles ist es, die Handhabung und Nützlichkeit der Anwendung zu analysieren. Die vorgestellten Ergebnisse und Abläufe sollen in keiner Weise Teile einer literarischen Analyse darstellen. Anschließend wird Expertenfeedback vorgestellt, welches im Rahmen einer CRETA Werkstatt gesammelt wurde.

6.1 Anwendungsfall

Für den Anwendungsfall werden die in Abschnitt 4.2 vorgestellten Datensätze zu Buch drei und vier von Parzival verwendet. Parzival ist ein mittelhochdeutscher Versroman von Wolfram von Eschenbach. Der Roman wurde zwischen 1200 und 1210 nach Christus verfasst und handelt von den zwei Rittern Parzivâl und Gâwân, sowie deren Interaktion mit anderen Charakteren [BEJR17]. In diesem Anwendungsfall werden die Datenmengen von einer fiktionalen Literaturwissenschaftlerin analysiert. Sie ist bereits mit dem Inhalt und den Charakteren von Buch drei und vier von Parzival vertraut. Mit Hilfe der Anwendung will sie Gemeinsamkeiten und Unterschiede zwischen beiden Büchern finden. Außerdem will sie die vier bereitgestellten Datensätze miteinander und deren verschiedene Annotationsergebnisse vergleichen.

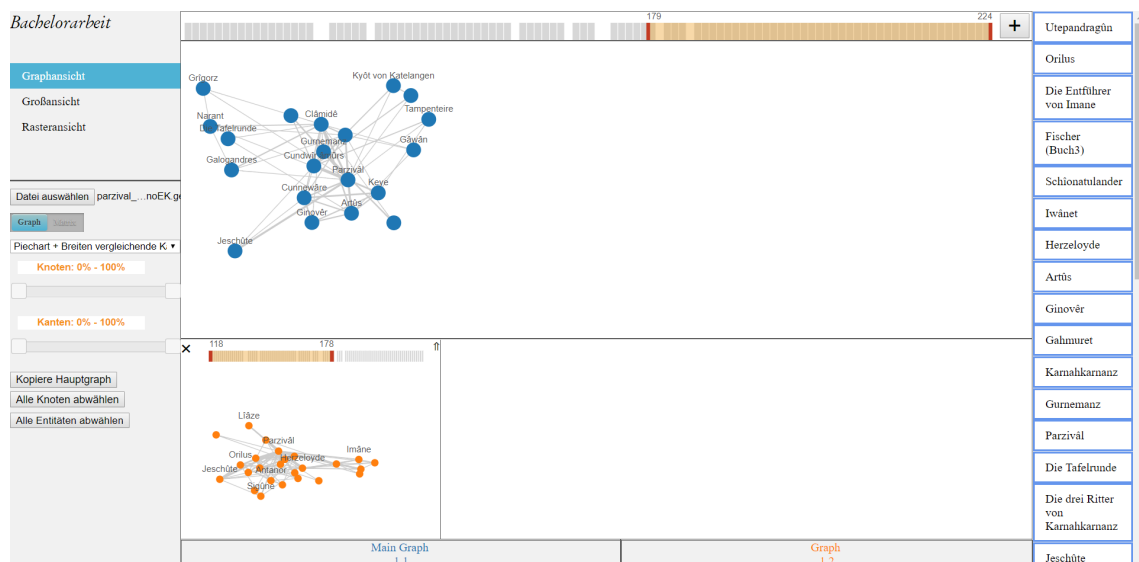


Abbildung 6.1: Benutzeroberfläche mit den Figurennetzwerken zu Buch drei und vier von Parzival.

6 Evaluation

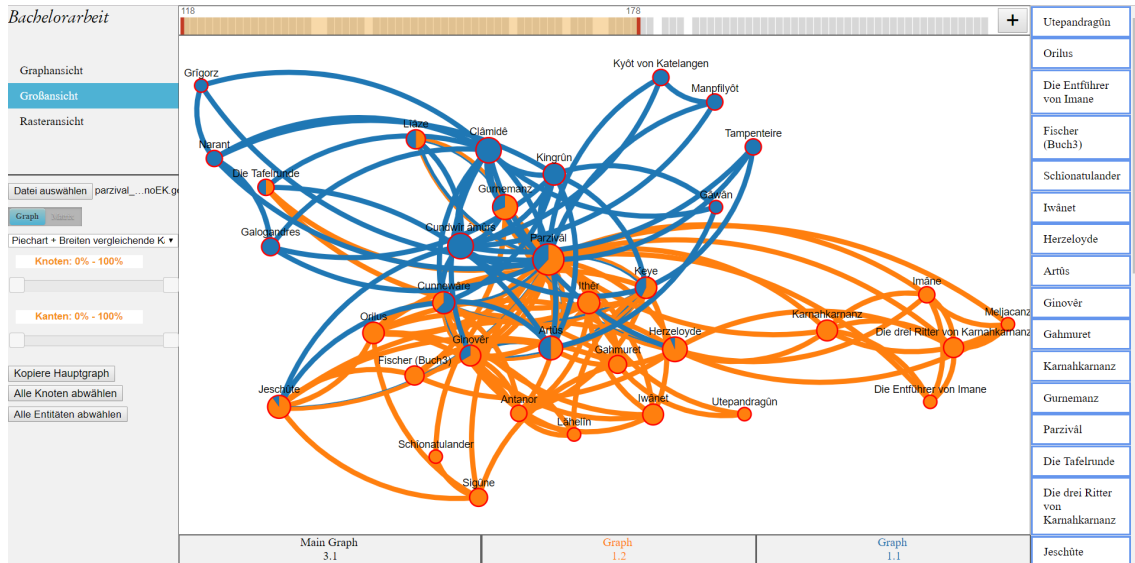


Abbildung 6.2: Der Vergleichsgraph zwischen Buch drei und vier von Parzival als Node-Link Diagramm in der Großansicht.

Zu Beginn importiert die Literaturwissenschaftlerin eine GEXF-Datei, welche durch eine 30 Zeilen Segmentierung annotiert wurde. Außerdem wurden Appellative bei der Annotation berücksichtigt. Nachdem das Figurennetzwerk importiert wurde, wird es in dem Hauptgraphen der Anwendung angezeigt. Als Erstes will die Literaturwissenschaftlerin Buch drei und vier von Parzival miteinander vergleichen. Dafür wählt sie den Textbereich von Buch drei aus. Anschließend kopiert sie den Hauptgraphen, welcher nun ebenfalls als kleiner Graph in Orange, wie in Abbildung 6.1 zu sehen ist, dargestellt wird. Anschließend wechselt sie, wie ebenfalls in der Darstellung zu sehen ist, den Textbereich des Hauptgraphen auf das Start- und Endsegment des vierten Buches. Bereits jetzt sieht sie einige Unterschiede in den Figurennetzwerken beider Bücher. So erkennt sie, dass eine Gruppierung von Entitäten welche in dem Figurennetzwerk zu Buch drei rechts zu sehen ist, nicht in Buch vier vorkommt. Durch das Verwenden der Zoomfunktion erkennt sie, dass die Namen der Entitäten „Imâne“, „Karnahkarnz“, „Meljacanz“, „Die drei Ritter von Karnahkarnz“ und „Die Entführer von Imane“ sind.

Um leichter weitere Unterschiede zwischen den beiden Büchern zu erkennen, erstellt sie durch einen Klick auf das Pfeilsymbol des kleinen Graphen einen Vergleichsgraphen. Dieser wird ihr daraufhin als Node-Link Diagramm dargestellt. Da die Literaturwissenschaftlerin nun den Graphen genauer untersuchen will, wechselt sie in die Großansicht, um ihn größer darzustellen. Die Großansicht mit dem dargestellten Vergleichsgraphen ist in Abbildung 6.2 zu sehen. Anhand des vergleichenden Node-Link Diagramm erkennt die Literaturwissenschaftlerin, dass sehr viele Entitäten des Romanes nur in einem der beiden Bücher vorkommen.

Bevor sie den Vergleichsgraphen weiter filtert, schaut sie sich den Vergleichsgraphen als Adjazenzmatrix, zu sehen in Abbildung 6.3 an, um Unterschiede in den Relationen leichter untersuchen zu können. Mit Hilfe der Adjazenzmatrix erkennt sie schnell, dass Parzival in Buch drei am häufigsten mit Herzeloyde, seiner Mutter, in Relation steht. Nachdem er jedoch seine Mutter verlassen hat steht er in Buch vier am häufigsten mit Cundwîramûrs, seiner zukünftigen Frau in Relation. Die Literaturwissenschaftlerin erinnert sich, dass Herzeloyde nach dem Weggang von Parzival stirbt

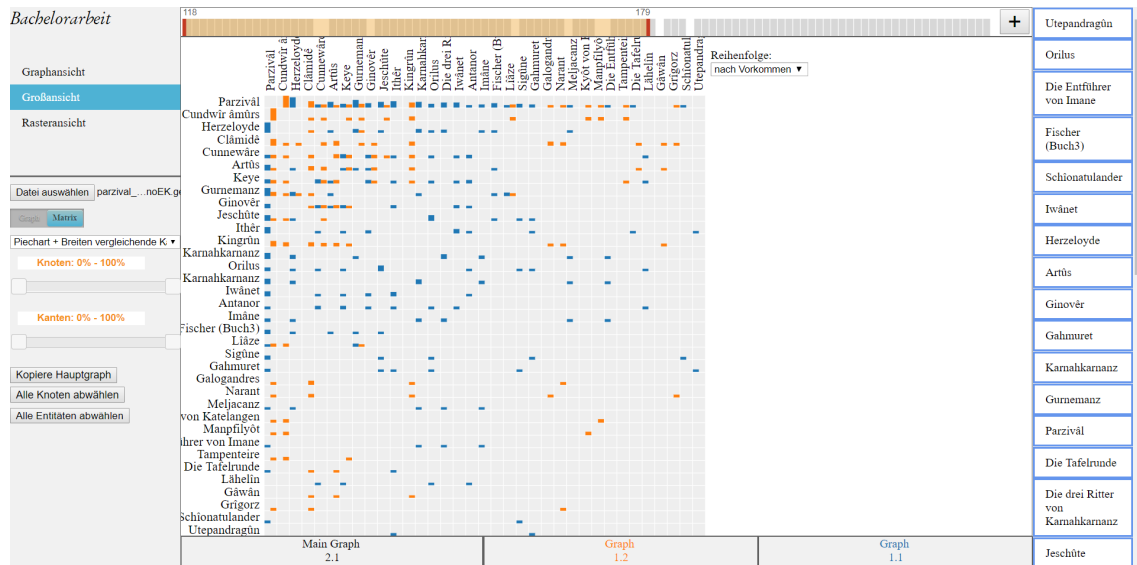


Abbildung 6.3: Der Vergleichsgraph zwischen Buch drei und vier von Parzival als Adjazenzmatrix in der Großansicht.

und er Cundwîramûrs erst in Buch vier kennenlernt. Aus diesen Gründen kommen die jeweiligen Relationen nur in einem der beiden Bücher vor.

Für den Vergleich beider Bücher will sie nun alle Entitäten dargestellt haben, welche in beiden Büchern vorkommen. Um zunächst einen besseren Überblick über alle Relationen und Entitäten zu bekommen, wechselt sie wieder in die Darstellung als Node-Link Diagramm. Anschließend wählt sie mit Hilfe des oberen Schiebereglers aus, dass nur Knoten welche sich um einen Prozentsatz zwischen 0% und 99% unterscheiden, farblich angezeigt werden sollen. Jedoch werden weiterhin Relationen im Node-Link Diagramm angezeigt, welche nur in einem der Bücher vorkommen. Deshalb wählt sie im Schieberegler darunter aus, dass nur Kanten angezeigt werden sollen, welche sich ebenfalls um einen Prozentsatz von 0% bis 99% unterscheiden. Damit werden nun alle Knoten und Kanten, welche nur in einem Figurennetzwerk vorkommen, ausgegraut. Daraufhin werden ihr nur noch die zehn Entitäten farblich angezeigt, welche in beiden Büchern genannt werden. Da „Die Tafelrunde“ zwar in beiden Büchern einmal vorkommt, aber nur in Buch vier mit Artus in Relation steht, graut die Literaturwissenschaftlerin die Entität durch einen Klick auf dessen Knoten ebenfalls aus. Der dadurch entstandene Vergleichsgraph ist in Abbildung 6.4 zu sehen. Dabei macht sie die Entdeckung, dass Herzelojde einmal in Buch vier genannt wird, obwohl sie bereits in Buch drei gestorben ist. Ob es sich dabei um eine richtige Information oder um einen Fehler der maschinellen Sprachverarbeitung handelt, muss sie selbst anhand des Textes prüfen.

Des Weiteren will die Literaturwissenschaftlerin untersuchen, wie eine andere Segmentierung und das Miteinbeziehen oder Nichtberücksichtigen von Appellativen das Figurennetzwerk über das vierte Buch ändert. Dadurch will sie unter anderem überprüfen, ob in den anderen Datensätzen Herzelojde ebenfalls in Buch vier auftritt. Da sie für die weitere Analyse mit Hilfe weiterer importierten Figurennetzwerke nur Entitäten beachten will, welche in beiden Büchern vorkommen, wählt sie die jeweiligen Entitäten in der Entitätenliste aus. Dafür wählt sie zuerst alle Entitäten mit Hilfe der „Alle Entitäten abwählen“-Schaltfläche ab. Anschließend selektiert sie die neun Entitäten in der Entitätenliste, welche in beiden Büchern vorkommen. Außerdem wechselt sie

6 Evaluation

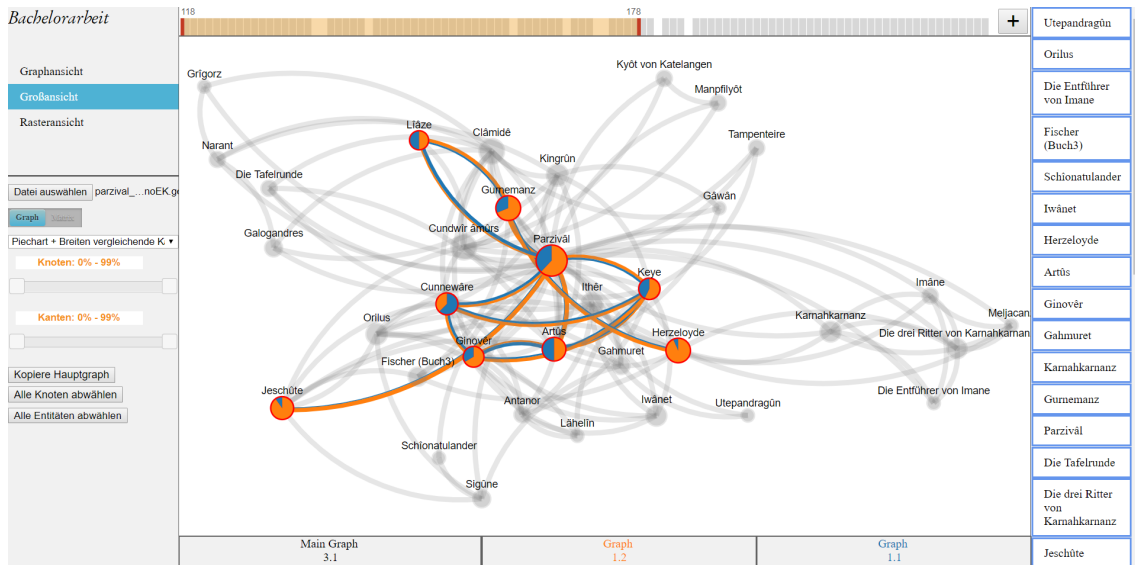


Abbildung 6.4: Der Vergleichsgraph zwischen Buch drei und vier von Parzival in dem alle Knoten und Kanten angezeigt werden, welche in beiden Büchern vorkommen.

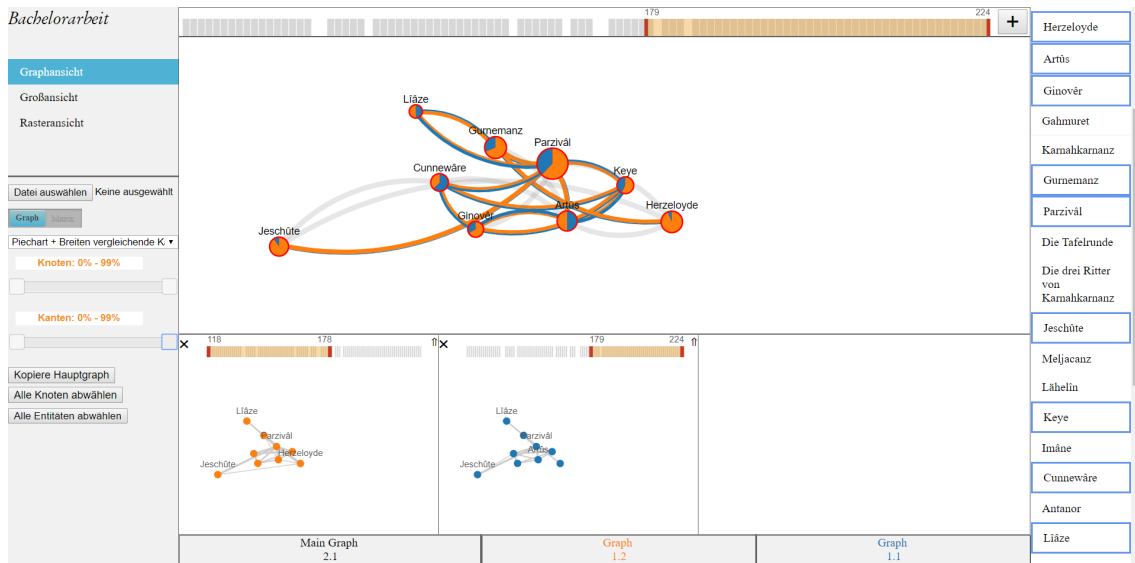


Abbildung 6.5: Der Vergleichsgraph zwischen Buch drei und vier von Parzival, sowie die Figurennetzwerke zu den einzelnen Büchern nach Entitäten gefiltert, welche in beiden Büchern vorkommen.

wieder in die Graphansicht zurück. Die dadurch gefilterten Figurennetzwerke in der Graphansicht sind in Abbildung 6.5 zu sehen.

Da die Literaturwissenschaftlerin nun lediglich Buch vier von Parzival vergleichen will, zieht sie den bereits dargestellten Netzwerkgraphen über Buch vier per Drag und Drop in den Hauptgraphen. Anschließend schließt sie beide kleinen Graphen, indem sie jeweils auf die X-Schaltfläche drückt. Danach importiert sie die drei anderen zur Verfügung stehenden Datensätze. Jedes Figu-

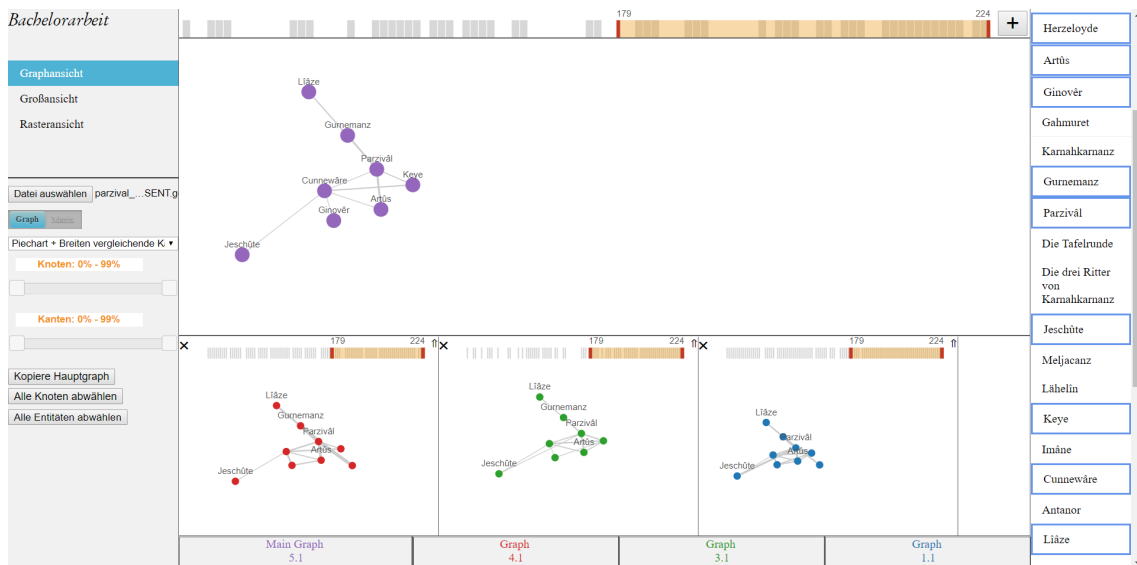


Abbildung 6.6: Alle vier Datensätze über Buch vier von Parzival in der Graphansicht.

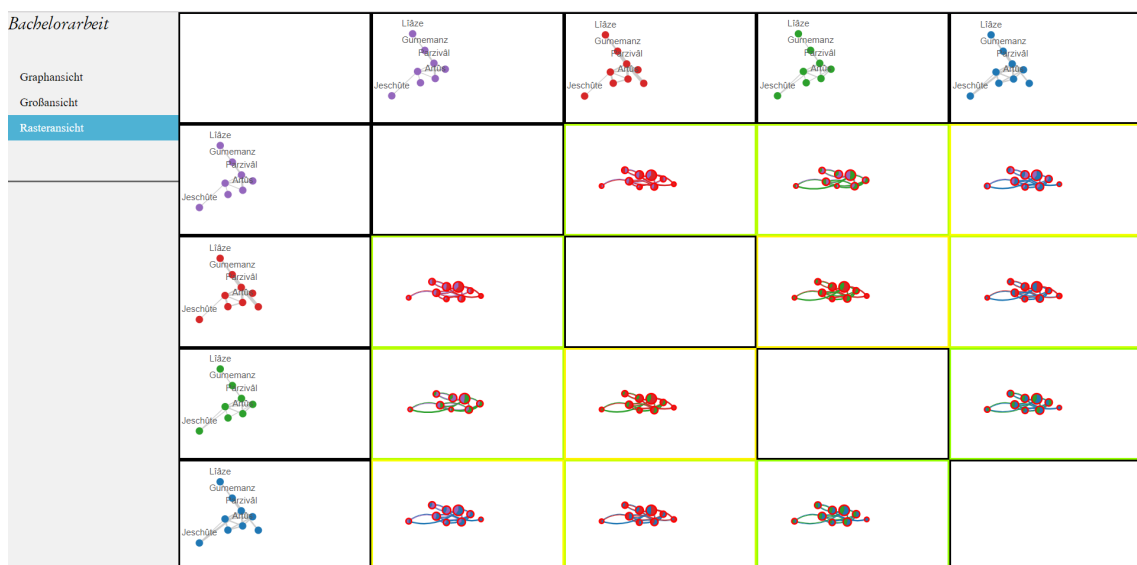


Abbildung 6.7: Die vier importierten Graphen, sowie alle Vergleichsgraphen in der Rasteransicht.

rennetzwerk filtert sie dabei nach dem Textbereich des vierten Buches. Die dadurch dargestellten Graphen sind in Abbildung 6.6 zu sehen. Anhand der Graphansicht kann sie erkennen, dass alle Figurennetzwerke ähnlich zueinander sind. Jedoch taucht Herzelojde nur in jenen zwei Datensätzen auf, welche Appellative berücksichtigen.

Um leichter einen Überblick über alle Unterschiede der importierten Figurennetzwerke zu bekommen, wechselt sie in die Rasteransicht, zu sehen in Abbildung 6.7. Auf den ersten Blick erkennt sie anhand der Rahmenfarben, dass alle Figurennetzwerke sehr ähnlich zueinander sind und sich kein Figurennetzwerk stark von den anderen unterscheidet. Den stärksten Unterschied

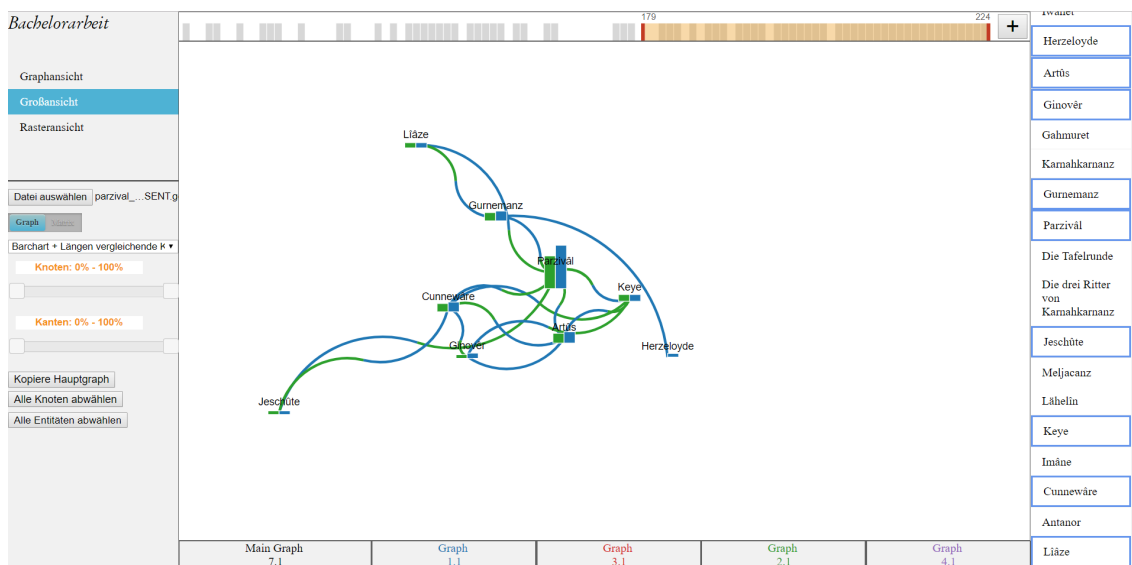


Abbildung 6.8: Der Vergleichsgraph mit geänderten Kanten und Knotendarstellungen in der Großansicht.

haben die einzelnen Figurennetzwerke jeweils zu den Netzwerken, welche sich durch eine andere Textsegmentierung und die Verwendung von Appellativen unterscheiden.

Weiter will die Literaturwissenschaftlerin untersuchen, inwiefern sich das Figurennetzwerk unterscheidet, wenn Appellative berücksichtigt oder nicht berücksichtigt werden. Deshalb doppelklickt sie auf den grünblauen Vergleichsgraphen. Daraufhin öffnet sich die Großansicht, in welcher der Vergleichsgraph als Node-Link Diagramm dargestellt wird. Da sie für kleinere Vergleichsgraphen die Darstellung der Knoten als Balkendiagramm und die andere Kantendarstellung bevorzugt, wechselt sie die visuelle Struktur über die Dropdown-Liste. Anschließend wird der Vergleichsgraph, wie in Abbildung 6.8, dargestellt. Hier sieht sie, dass Herzeloyde lediglich nur in einem der beiden Figurennetzwerke vorkommt. Jedoch werden auch einige anderen Entitäten, wie zum Beispiel Parzival oder Artüs durch Appellative häufiger erkannt.

Im Anschluss wechselt sie in die Matrixansicht, um besser zu erkennen, inwiefern sich die Relationen in beiden Figurennetzwerken unterscheiden. Anhand des Schiebereglers stellt sie nachfolgend ein, dass nur Relationen welche sich um 50% oder mehr unterscheiden, angezeigt werden sollen. Durch die Adjazenzmatrix, welche in Abbildung 6.9 zu sehen ist, erkennt sie, dass drei Relationen sich stark unterscheiden. So werden Relationen zwischen Artüs und Ginovêr, sowie zwischen Herzeloyde und Gurnemanz nur durch die Verwendung von Appellativen gefunden. Die Relationen zwischen Parzival und Gurnemanz unterscheiden sich ebenfalls stark, da durch Appellative drei Relationen im Text gefunden werden und ohne Appellative lediglich eine.

6.2 Expertenfeedback

Die implementierte Anwendung wurde während einer CRETA Werkstatt verschiedenen Professoren und wissenschaftlichen Mitarbeitern der Literatur- und Sozialwissenschaften vorgestellt. Ziel der Werkstatt war es den verschiedenen Projektteilnehmern aktuelle Forschungsthemen der

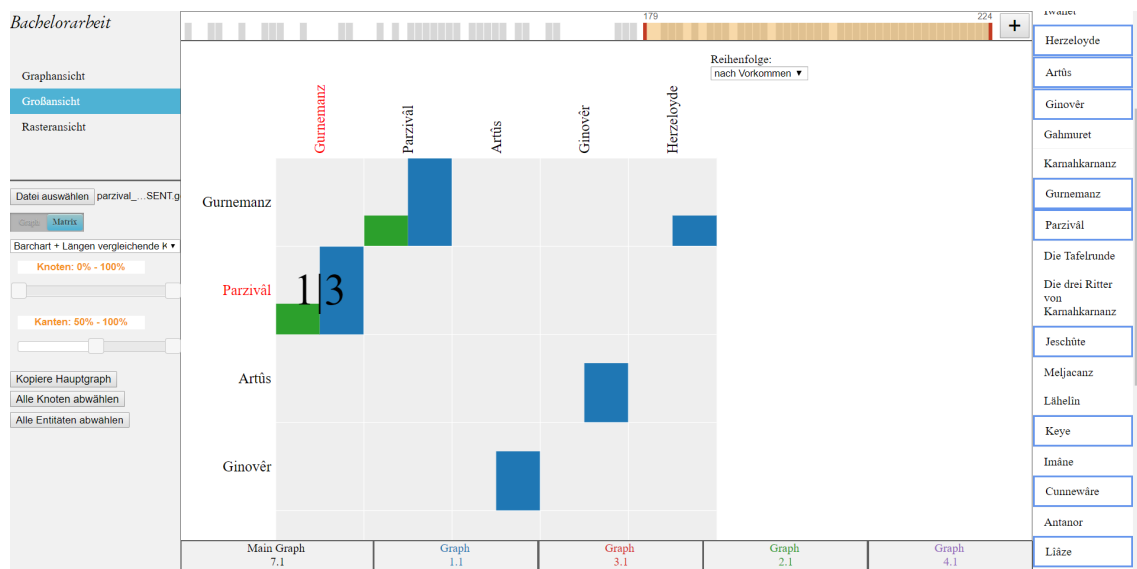


Abbildung 6.9: Unterschiede der Relationen in Buch vier von Parzival wenn Appellative berücksichtigt werden (blau) oder nicht berücksichtigt werden (grün).

verschiedenen Fachgruppen vorzustellen und über diese zu diskutieren. Nach der Vorstellung der Anwendung war die Resonanz der Beteiligten durchweg positiv. Viele Teilnehmer konnten in der Anwendung Verwendung für eine ihrer aktuellen Forschungsthemen finden. Positiv hervorgehoben wurde unter anderem, dass die Anwendung relativ einfach gehalten ist, da sie sich nur auf die Darstellung und den Vergleich von Graphen fokussiert. Ebenfalls wurde die Darstellung der Graphen als Node-Link Diagramm oder als Adjazenzmatrix und der mögliche Wechsel zwischen den visuellen Strukturen geschätzt.

Nach der Präsentation und in einer anschließenden Diskussionsrunde in kleinen Gruppen wurden mögliche Verbesserungsmöglichkeiten vorgeschlagen und diskutiert, welche im Folgenden aufgeführt werden. Es wurde erörtert, in welchem Umfang die Anwendung erweitert werden sollte. So könnten Texte statt Figurennetzwerke in die Anwendung importiert werden, welche danach in der Anwendung annotiert werden. Anschließend könnten Figurennetzwerke aus den annotierten Texten erstellt werden. Dies hätte den Vorteil, dass nicht mehrere Anwendungen benutzt werden müssten, bevor die Figurennetzwerke in der Anwendung verglichen werden können. Auch hätte man zusätzlich alle Detailinformationen, wie den literarischen Text und andere Daten, die während des Annotieren des Textes oder dem Erstellen des Figurennetzwerkes entstehen. Jedoch würde dies den Umfang der Anwendung deutlich erhöhen und die zuvor positiv bewertete Einfachheit des Programmes beeinflussen. Auch existieren bereits bewährte und umfangreiche Programme, wie zum Beispiel Gephi um einzelne Figurennetzwerke zu erstellen und zu analysieren.

Bedarf an Verbesserung wurde in der Auswahl von Entitäten gesehen. Ein Vorschlag war die Entitätenliste nach verschiedenen Kategorien, wie dem Alphabet oder dem Vorkommen sortieren zu können, da häufig nur die wichtigsten Entitäten in einem Figurennetzwerk von Interesse sind. Ebenfalls wurde eine effizientere Auswahl mehrerer Entitäten vorgeschlagen, in dem mit gehaltener Maustaste über mehrere Entitäten gezogen werden kann, welche anschließend selektiert werden.

Eine weitere Idee war die dargestellten Informationen des Fingerprints zu erweitern. So könnte das Vorkommen von einzelnen ausgewählten Entitäten in Segmenten farbig im Fingerprint markiert werden. Damit hätte der Benutzer einen schnellen Überblick wann Entitäten auftreten und könnte auch die Nennungen verschiedener Entitäten über den Text vergleichen. Ein weiterer Aspekt für die Literaturwissenschaftler war der Raumverlauf über den Text. So könnte mit Hilfe der Anwendung Textbereiche verglichen werden, welche an verschiedenen Orten stattfinden. Um diese Textbereiche leichter zu erkennen, könnten im Fingerprint die verschiedenen Orte farbig markiert werden. Auch könnten weitere Visualisierungsmethoden für die Anwendung implementiert werden, welche den Raumverlauf verschiedener Texte miteinander vergleichen.

Für die Literaturwissenschaftler war ebenfalls wichtig, Wendepunkte leichter in der Handlung eines Figurennetzwerkes zu finden. So war ein Vorschlag für die Auswahl des Textbereiches, nicht erst nachdem die Maus losgelassen wird das neue berechnete Figurennetzwerk anzuzeigen, sondern noch während der Regler bedient wird und über ein weiteres Textsegment gezogen wird. Damit entsteht, wie auch im Time-to-time Mapping, eine Animation und der Benutzer kann schneller Textbereiche erkennen, in welchen große Änderungen vorkommen. Auch wurde vorgeschlagen nicht nur das minimale oder das maximale Textsegmente ändern zu können, sondern einen eingestellten Textbereich verschieben zu können. Somit könnten Figurennetzwerke, welche zum Beispiel über zwanzig Textsegmente gehen, animiert über den Verlauf des Textes dargestellt werden.

Ein weiterer Vorschlag für eine bessere Verknüpfung zwischen Node-Link Diagramm und Adjazenzmatrix war, beide visuelle Strukturen gleichzeitig nebeneinander anzuzeigen. Fährt der Benutzer mit der Maus über einen Knoten im Node-Link Diagramm so könnte die Spalte und Zeile des Knotens in der Adjazenzmatrix hervorgehoben werden. Umgekehrt, wenn mit der Maus über eine Zelle in der Adjazenzmatrix gefahren wird, könnte die Kante im Node-Link Diagramm markiert werden. Besonders für Vergleichsgraphen könnte der Benutzer so Knoten und Kanten leichter vergleichen ohne zwischen den beiden visuellen Strukturen wechseln zu müssen. Auch könnte der Benutzer Knoten und Kanten in der anderen Darstellungsart so leicht wiederfinden.

7 Zusammenfassung und Ausblick

In dieser Arbeit wurde ein Konzept zur Darstellung und zum Vergleich von Figurennetzwerken vorgestellt. Das Konzept wurde dabei mit Hilfe von zuvor präsentierten Grundlagen und verwandten Arbeiten entwickelt. Anschließend wurde anhand des vorgestellten Konzeptes eine Webanwendung implementiert. Mit Hilfe von GEXF-Dateien können Figurennetzwerke in die Anwendung importiert und dem Benutzer dargestellt werden. Dabei können die Figurennetzwerke dem Benutzer als Node-Link Diagramm oder als Adjazenzmatrix visualisiert werden, wobei jederzeit die visuelle Struktur vom Benutzer gewechselt werden kann. Ebenfalls ist es möglich, mehrere Figurennetzwerke gleichzeitig darzustellen, um einen Überblick über alle importierten Daten zu bekommen, als auch einen Vergleich in Juxtaposition zu ermöglichen. Darüber hinaus wurde mit dem Vergleichsgraphen eine visuelle Struktur vorgestellt, um zwei Figurennetzwerke einfach miteinander vergleichen zu können. Der Vergleichsgraph kann als vergleichendes Node-Link Diagramm oder als vergleichende Adjazenzmatrix dem Benutzer angezeigt werden. Da beide visuelle Strukturen unterschiedliche Vorteile und Nachteile besitzen liegt bei dem vergleichenden Node-Link Diagramm der Vergleich von Entitäten zweier Figurennetzwerke im Fokus. Dagegen können mit der vergleichenden Adjazenzmatrix leicht Unterschiede zwischen den Relationen zweier Figurennetzwerke festgestellt werden. Beide visuelle Strukturen vergleichen dabei zwei Datenmengen durch eine hybride Darstellung aus Superimposition und Explicit Encoding. Da die importierten Datenmengen auch komplex und groß sein können, werden dem Benutzer verschiedene Filtermöglichkeiten zur Verfügung gestellt, um die dargestellt Datenmenge zu reduzieren. So können alle in der Anwendung dargestellten Graphen nach Textbereichen gefiltert werden. Auch können Entitäten ausgewählt werden, welche in den Graphen dargestellt oder nicht berücksichtigt werden sollen. Da Vergleichsgraphen mehr Platz für die Darstellung der Knoten und Kanten benötigen, kann der Benutzer auswählen, dass nur Entitäten oder Relationen dargestellt werden sollen, welche sich in den zwei zugrunde liegenden Netzwerkgraphen um einen bestimmten Prozentbereich unterscheiden. Damit der Benutzer einzelne Graphen genauer untersuchen kann, ist es möglich einen Graphen in der Großansicht alleinig darzustellen. Darüber hinaus kann der Benutzer in der Rasteransicht eine Übersicht über alle erstellbaren Vergleichsgraphen bekommen. Abschließend wurde die Bedienung und Praktikabilität der Anwendung anhand eines Anwendungsfalles evaluiert. Auch wurde Expertenfeedback vorgestellt, welches im Rahmen einer CRETA Werkstatt gesammelt wurde.

Ausblick

Neben den im Expertenfeedback (siehe Abschnitt 6.2) genannten Verbesserungsmöglichkeiten wird im folgenden ein Ausblick auf weitere Erweiterungen und Verbesserung der vorgestellten Forschungsarbeit gegeben.

Bisher werden die die Positionen von Knoten eines Node-Link Diagramms aus dem importierten Datensatz übernommen. Jedoch sind die Positionen für den Platz, in welchem das Figurennetzwerk dargestellt werden soll, nicht immer optimal. Auch könnten in anderen Datensätzen die Knotenpositionen gleicher Entitäten nicht immer äquivalent sein. Aus diesen Gründen könnte die Layoutberechnung eines Node-Link Diagrammes mit in die Anwendung implementiert werden.

Wie auch schon in Abschnitt 4.1 erwähnt, kann in der bisherigen Arbeit der literarische Text des importierten Figurennetzwerks in der Anwendung nicht nachgelesen werden. Wie auch schon von Jänicke et al. [JFCS15] festgestellt ist es nicht nur wichtig Daten in Form einer Visualisation dargestellt zu haben, auch „Distant Reading“ genannt. Sondern gerade die Kombination mit „Close Reading“, also dem direkten Zugriff auf den Quelltext, ist für Geisteswissenschaftler von Vorteil, wenn sie mit Hilfe von Visualisierungen Texte analysieren. So kann der Benutzer in der Anwendung durch das direkte Springen auf Textstellen, in welchen ausgewählte Relationen oder Entitäten vorkommen, sowohl die präsentierten Ergebnisse überprüfen, als auch weitere Informationen für die Analyse erlangen.

Zuletzt könnte eine Adjazenzmatrix anhand einer Fokus und Kontext Technik, wie Table Lens von Rao et al. [RC94], genauer untersucht werden. Diese würde dem Benutzer ermöglichen einzelne Zellen einer Adjazenzmatrix zu vergrößern. Dadurch könnten mehr Informationen in einer einzelnen Zelle einer Adjazenzmatrix dargestellt werden, wenn diese vom Benutzer genauer untersucht wird.

Weiterhin ist geplant eng mit Geisteswissenschaftler zusammenzuarbeiten, um die Anwendung zu erweitern und zu verbessern.

Literaturverzeichnis

- [ABH+13] B. Alper, B. Bach, N. Henry Riche, T. Isenberg, J.-D. Fekete. „Weighted Graph Comparison Techniques for Brain Connectivity Analysis“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '13. Paris, France: ACM, 2013, S. 483–492. ISBN: 978-1-4503-1899-0. DOI: [10.1145/2470654.2470724](https://doi.org/10.1145/2470654.2470724). URL: <http://doi.acm.org/10.1145/2470654.2470724> (zitiert auf S. 24–26, 37).
- [AWW09] K. Andrews, M. Wohlfahrt, G. Wurzinger. „Visual Graph Comparison“. In: *2009 13th International Conference Information Visualisation*. Juli 2009, S. 62–67. DOI: [10.1109/IV.2009.108](https://doi.org/10.1109/IV.2009.108) (zitiert auf S. 21).
- [BBD09] F. Beck, M. Burch, S. Diehl. „Towards an Aesthetic Dimensions Framework for Dynamic Graph Visualisations“. In: *2009 13th International Conference Information Visualisation*. Juli 2009, S. 592–597. DOI: [10.1109/IV.2009.42](https://doi.org/10.1109/IV.2009.42) (zitiert auf S. 18, 37).
- [BBD13] F. Beck, M. Burch, S. Diehl. „Matching Application Requirements with Dynamic Graph Visualization Profiles“. In: *2013 17th International Conference on Information Visualisation*. Juli 2013, S. 11–18. DOI: [10.1109/IV.2013.2](https://doi.org/10.1109/IV.2013.2) (zitiert auf S. 19).
- [BBDW14] F. Beck, M. Burch, S. Diehl, D. Weiskopf. „The State of the Art in Visualizing Dynamic Graphs“. In: *EuroVis - STARS*. Eurographics Association, 2014, S. 83–103. DOI: [10.2312/eurovisstar.20141174](https://doi.org/10.2312/eurovisstar.20141174). URL: <http://dx.doi.org/10.2312/eurovisstar.20141174> (zitiert auf S. 19).
- [BBW16] F. Beck, M. Burch, D. Weiskopf. „A Matrix-based Visual Comparison of Time Series Sports Data“. In: *Proceedings of the Conference on Vision, Modeling and Visualization*. VMV '16. Bayreuth, Germany: Eurographics Association, 2016, S. 53–60. ISBN: 978-3-03868-025-3. DOI: [10.2312/vmv.20161342](https://doi.org/10.2312/vmv.20161342). URL: <https://doi.org/10.2312/vmv.20161342> (zitiert auf S. 23, 36).
- [BEJR17] A. Blessing, N. Echelmeyer, M. John, N. Reiter. „An End-to-end Environment for Research Question-Driven Entity Extraction and Network Analysis“. In: *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*. Vancouver, Canada: Association for Computational Linguistics, Aug. 2017, S. 57–67. DOI: [10.18653/v1/W17-2208](https://doi.org/10.18653/v1/W17-2208). URL: <http://www.aclweb.org/anthology/W/W17/W17-2208> (zitiert auf S. 28, 49).
- [Bur16] M. Burch. „The Dynamic Call Graph Matrix“. In: *Proceedings of the 9th International Symposium on Visual Information Communication and Interaction*. VINCI '16. Dallas, TX, USA: ACM, 2016, S. 1–8. ISBN: 978-1-4503-4149-3. DOI: [10.1145/2968220.2968226](https://doi.org/10.1145/2968220.2968226). URL: <http://doi.acm.org/10.1145/2968220.2968226> (zitiert auf S. 24, 38).

- [CMS99] S. K. Card, J. D. Mackinlay, B. Shneiderman. „Readings in Information Visualization“. In: Hrsg. von S. K. Card, J. D. Mackinlay, B. Shneiderman. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999. Kap. Information Visualization, S. 1–34. ISBN: 1-55860-533-9. URL: <http://dl.acm.org/citation.cfm?id=300679.300680> (zitiert auf S. 15, 16, 28).
- [GAW+11] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, J. C. Roberts. „Visual Comparison for Information Visualization“. In: *Information Visualization* 10.4 (Okt. 2011), S. 289–309. ISSN: 1473-8716. DOI: [10.1177/1473871611416549](https://doi.org/10.1177/1473871611416549). URL: <http://dx.doi.org/10.1177/1473871611416549> (zitiert auf S. 16, 17).
- [GFC04] M. Ghoniem, J.-D. Fekete, P. Castagliola. „A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations“. In: *Proceedings of the IEEE Symposium on Information Visualization*. INFOVIS '04. Washington, DC, USA: IEEE Computer Society, 2004, S. 17–24. DOI: [10.1109/INFOVIS.2004.1](https://doi.org/10.1109/INFOVIS.2004.1). URL: <http://dx.doi.org/10.1109/INFOVIS.2004.1> (zitiert auf S. 14, 29).
- [HFM07] N. Henry, J.-D. Fekete, M. J. McGuffin. „NodeTrix: A Hybrid Visualization of Social Networks“. In: *IEEE Transactions on Visualization and Computer Graphics* 13.6 (Nov. 2007), S. 1302–1309. ISSN: 1077-2626. DOI: [10.1109/TVCG.2007.70582](https://doi.org/10.1109/TVCG.2007.70582). URL: <http://dx.doi.org/10.1109/TVCG.2007.70582> (zitiert auf S. 25, 29).
- [JFCS15] S. Jänicke, G. Franzini, M. F. Cheema, G. Scheuermann. „On Close and Distant Reading in Digital Humanities: A Survey and Future Challenges“. In: *Eurographics Conference on Visualization (EuroVis) - STARs*. Hrsg. von R. Borgo, F. Ganovelli, I. Viola. The Eurographics Association, 2015. DOI: [10.2312/eurovisstar.20151113](https://doi.org/10.2312/eurovisstar.20151113) (zitiert auf S. 58).
- [MGK11] A. G. Melville, M. Graham, J. B. Kennedy. „Combined vs. Separate Views in Matrix-based Graph Analysis and Comparison“. In: *2011 15th International Conference on Information Visualisation*. Juli 2011, S. 53–58. DOI: [10.1109/IV.2011.49](https://doi.org/10.1109/IV.2011.49) (zitiert auf S. 24, 25).
- [RC94] R. Rao, S. K. Card. „The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus + Context Visualization for Tabular Information“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '94. Boston, Massachusetts, USA: ACM, 1994, S. 318–322. ISBN: 0-89791-650-6. DOI: [10.1145/191666.191776](https://doi.org/10.1145/191666.191776). URL: <http://doi.acm.org/10.1145/191666.191776> (zitiert auf S. 58).
- [Shn96] B. Shneiderman. „The eyes have it: a task by data type taxonomy for information visualizations“. In: *Proceedings 1996 IEEE Symposium on Visual Languages*. Sep. 1996, S. 336–343. DOI: [10.1109/VL.1996.545307](https://doi.org/10.1109/VL.1996.545307) (zitiert auf S. 15).
- [SLN05] P. Saraiya, P. Lee, C. North. „Visualization of graphs with associated timeseries data“. In: *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005*. Okt. 2005, S. 225–232. DOI: [10.1109/INFVIS.2005.1532151](https://doi.org/10.1109/INFVIS.2005.1532151) (zitiert auf S. 22).
- [SSU08] S. Schreibman, R. Siemens, J. Unsworth. *A Companion to Digital Humanities*. Wiley Publishing, 2008. ISBN: 9781405168069 (zitiert auf S. 13).

- [Tha17] M. Thaller. „Digital Humanities als Wissenschaft“. In: *Digital Humanities: Eine Einführung*. Hrsg. von F. Jannidis, H. Kohle, M. Rehbein. Stuttgart: J.B. Metzler, 2017, S. 13–18. ISBN: 978-3-476-05446-3. DOI: [10.1007/978-3-476-05446-3_2](https://doi.org/10.1007/978-3-476-05446-3_2). URL: https://doi.org/10.1007/978-3-476-05446-3_2 (zitiert auf S. 13).
- [The12] The Humanities and Technology Camp. *Glossary of Digital Humanities*. 2012. URL: https://docs.google.com/document/d/129_260mbp6C6JaqmylmhMTXA1sv7YYQ0OLsAOAOoCJE/edit (zitiert auf S. 13).

Alle URLs wurden zuletzt am 27. 10. 2017 geprüft.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift