

Institut für Visualisierung und  
Interaktive Systeme  
Universitätsstraße 38  
D-70569 Stuttgart

Bachelorarbeit

# **Interaktive Annotation und Visualisierung von korrelierten Textsegmenten**

Jena Satkunarajan

<b>Studiengang:</b>	Informatik
<b>Prüfer/in:</b>	Prof. Dr. Thomas Ertl
<b>Betreuer/in:</b>	Dipl.-Math. Martin Baumann, Dr. Steffen Koch, M. Sc. Markus John
<b>Beginn am:</b>	9. Mai 2018
<b>Beendet am:</b>	9. November 2018



## Kurzfassung

Die Textanalyse kommt im Rahmen literaturwissenschaftlicher und linguistischer Aufgaben nicht ohne die Annotation von Texten aus. Die Auswahl an Werkzeugen zur Annotation kurzer Textabschnitte ist vielfältig, schwindet jedoch rasch, wenn zusätzlich längere Abschnitte und Relationen zwischen annotierten Textsegmenten bearbeitet und dargestellt werden sollen.

Der Fokus dieser Bachelorarbeit liegt auf der interaktiven Annotation und Visualisierung von Textsegmenten und den Beziehungen zwischen diesen. Ein besonderer Schwerpunkt stellt die Auszeichnung und Darstellung von Relationen zwischen Annotationen in weit entfernten Textabschnitten dar, welcher unter Einsatz unterschiedlicher Textansichten und einer geeigneten Visualisierung der Distanz zwischen Textabschnitten realisiert werden soll. Sich auf diesen Schwerpunkt konzentrierend, wird im Rahmen dieser Arbeit ein Ansatz entwickelt, der die interaktive Annotation von Textsegmenten verschiedener Längen und Relationen zwischen den Annotationen umsetzt. Zur Visualisierung der Distanzen zwischen mehreren Abschnitten, ermöglicht der Ansatz die Aufteilung des Textes auf mehrere, zusammenhängende Ansichten, welche interaktiv mittels einer visuellen Metapher auf die Veränderungen der Distanz zwischen den Abschnitten reagieren. Die Funktionen zur Auszeichnung von Annotationen und Relationen werden in allen Ansichten integriert, um die abschnittsweise Bearbeitung des Textes zu erlauben. Unter Abwägung der verschiedenen Möglichkeiten zur Darstellung von Relationen im Entwurfsprozess, wird eine Darstellung gewählt, welche die Herausforderungen bezüglich der Überlagerung und Überschneidung von Relationen sinnvoll löst. Das interaktive Zusammenspiel der verschiedenen Komponenten wird hinsichtlich dem Informationsverlust durch temporär nicht sichtbarer Annotationen beleuchtet und um eine gezielte, visuelle Repräsentation der Informationen ergänzt. Die Komponenten des Ansatzes werden in einem Anwendungsszenario präsentiert. Im Rahmen einer Expertenbefragung und einer Diskussion werden die Ergebnisse des Ansatzes abschließend vorgestellt.

## **Abstract**

The annotation of text is an essential task in the fields of (computational) linguistics and literary studies. Several tools with a wide range of functionalities exist for the purpose of annotating short segments of text. The same does not hold if large segments of text have to be annotated and if correlations between such segments need to be visualized.

Subject of this work is the interactive annotation and visualization of segments of text and correlations between annotated segments. The work is mainly concerned with the annotation and visualization of distant relations between text segments of variable length and therefore investigates the possibilities of separating and displaying the text in several views. The views are then modified to visualize the textual distance between the texts of different views. Proceeding the modification an annotating scheme is implemented into the views that allows creating and combining different classes of annotations and relations between annotations. Several presentations of correlations between text segments are then compared against each other. The most promising representation regarding the overview of correlations and reducing clutter will then be used to visualize relations between annotations. To keep track of temporarily non-visible annotations an approach is presented that visualizes the information lost due to hidden annotations. An expert survey and a discussion will lastly present the results of this work.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Zielsetzung . . . . .	1
<b>2</b>	<b>Grundlagen</b>	<b>3</b>
2.1	Grundlagen der Informationsvisualisierung . . . . .	3
2.2	Grundlagen der Textvisualisierung . . . . .	5
2.3	Grundlagen der (relationierten) Annotation . . . . .	5
<b>3</b>	<b>Verwandte Arbeiten</b>	<b>11</b>
3.1	ANALEC . . . . .	11
3.2	Glozz . . . . .	13
3.3	BRAT . . . . .	15
<b>4</b>	<b>Konzepte</b>	<b>17</b>
4.1	Anforderungen . . . . .	17
4.1.1	Darstellung des Textes . . . . .	17
4.1.2	Visualisierung von Distanzen . . . . .	17
4.1.3	Erstellung von Annotationen und Relationen . . . . .	17
4.1.4	Visualisierung überlagernder Relationen . . . . .	18
4.1.5	Dynamische Visualisierung von Relationen und Interaktivität . . . . .	18
4.1.6	Benutzeroberfläche . . . . .	18
4.2	Entwurf . . . . .	18
4.2.1	Positionierung von Textsegmenten . . . . .	19
4.2.2	Initiale Ansicht und Oberfläche . . . . .	19
4.2.3	Funktion der Bänder . . . . .	21
	Streckung bei sinkender Distanz . . . . .	21
	Streckung bei steigender Distanz . . . . .	24
	Mehrere Trennfelder . . . . .	25
	Distanzkodierung . . . . .	28
4.2.4	Repräsentation und Erstellung von Annotationen und Relationen . . . . .	29
4.2.5	Visualisierung der Annotationen und Relationen . . . . .	30
	Grundlegende Visualisierung . . . . .	31
	Relationen in verschiedenen Ansichten und Umgang mit Überlagerung . . . . .	32
	Relationen als Bögen . . . . .	33
	Relationen in der Schwimmliniendarstellung . . . . .	35
	Relationen im kombinierten Ansatz . . . . .	37
	Filterung von Relationen . . . . .	39
	Verhalten der Annotationen und Relationen bei Scroll Operationen . . . . .	40

<b>5 Implementierung</b>	<b>45</b>
5.1 Technologie . . . . .	45
5.1.1 System . . . . .	45
5.1.2 Datensatz . . . . .	46
5.2 Benutzeroberfläche . . . . .	47
<b>6 Ergebnisse</b>	<b>51</b>
6.1 Anwendungsfälle . . . . .	51
6.1.1 Annotation des Textes in einer Splitansicht . . . . .	51
6.1.2 Auszeichnung von Relationen . . . . .	53
6.1.3 Lange Annotationen . . . . .	55
6.1.4 Relationen zwischen entfernten Annotationen . . . . .	55
6.1.5 Filterung von Relationen . . . . .	59
6.1.6 Nicht sichtbare Annotationen . . . . .	61
6.2 Expertenbefragung . . . . .	63
6.2.1 Annotation des Textes . . . . .	63
6.2.2 Split des Textes und Distanzvisualisierung . . . . .	63
6.2.3 Auszeichnung und Darstellung von Relationen . . . . .	64
6.2.4 Darstellung nicht sichtbarer Annotationen und Relationen . . . . .	65
6.2.5 Fazit . . . . .	66
6.3 Diskussion und Bewertung . . . . .	66
<b>7 Zusammenfassung und Ausblick</b>	<b>69</b>
7.1 Zusammenfassung . . . . .	69
7.2 Ausblick . . . . .	70
<b>Literaturverzeichnis</b>	<b>73</b>

# Abbildungsverzeichnis

2.1	Die Information Visualization Pipelien und ihre 5 Schritte zur Umwandlung der Daten in die visuelle Repräsentation nach Liu et al. [LCWL14] . . . . .	4
2.2	Die Identifikation der Teilnehmer beschreiben Breitenfeld und Müller-Birn in dieser Abbildung [BM17] als <i>named-entity-recognition</i> , dargestellt in den Aktionen T.1 und T.2. Im letzten Schritt wird die Richtung der Relation bestimmt, falls diese gerichtet ist, beispielsweise in einer Ursache-Folge-Beziehung. Alternativ kann die Relation ungerichtet sein, sodass Schritt T.4 ausgelassen werden kann. . . . .	7
2.3	Den Rahmen für die verschiedenen Darstellungen nach Burghardt [Bur18] bildet die Koreferenz-Annotation, bei der mehrere annotierte Pronomen einen ebenfalls annotierten Referenten, den sie innerhalb des Textes vertreten, referenzieren. . . . .	8
3.1	Links die Darstellung der Annotationen als Verkettung, Rechts die geometrische Ansicht von ANALEC [LPV12] . . . . .	11
3.2	Die Benutzeroberfläche von Glozz und ihre verschiedenen Ansichten [WM12] . . . . .	13
3.3	Die Benutzeroberfläche von BRAT [SPT+12] . . . . .	15
4.1	Der Entwurf der Benutzeroberfläche: (a) Die initiale Textansicht, (b) Das Trennfeld, (c) Freie Fläche zur Visualisierung der Relationen, (d) Menüleiste mit interaktiven Schnittstellen. Die Höhen- und Breitenmaße der 4 Hauptbereiche sind adaptiv, aber fix gewählt. Der zu bearbeitende Text befindet sich bereits in seiner Gesamtheit in der designierten Ansicht und lässt sich scrollen. . . . .	20
4.2	Es wird der Split des Textes am Satzende geezigt. Die gestrichelte Linie dient als Indikator für die Mauszeigerposition beim Ablegen des Trennfelds. . . . .	22
4.3	(a) Durch das Scrollen der unteren Ansicht nach unten, wird der Abschnitt vor dem grün markiertem Textsegment aus dem Sichtbereich der Ansicht nach oben verschoben. Die Wortmenge dieses Abschnitts beschreibt die Distanz zwischen dem letzten sichtbaren Wort der oberen und dem ersten sichtbaren Wort der letzten Ansicht nach der Scroll Operation. Die Distanz wird durch die Abrundung der Bänder visualisiert. (b) In der unteren Ansicht wurde nach oben gescrollt, sodass das grün markierte Segment nach unten verschoben wird und Teile des Textbereichs vor diesem wieder sichtbar werden. Die verborgene Textmenge zwischen dem letzten sichtbaren Wort der oberen Ansicht und dem nun neuen, ersten sichtbaren Wort der unteren Ansicht sinkt im Vergleich zu (a), genauso auch die Distanz. Die Bänder werden dadurch gestrafft. . . . .	23

4.4	Die Benutzeroberfläche wird ergänzt um ein weiteres Trennfeld unterhalb der initialen Textansicht, deren Höhe dadurch verringert wird. (a) Das erste Trennfeld wird an eine ausgewählte Stelle im Text der Initialansicht gezogen und losgelassen, wo es den ersten Split initiiert. (b) Die Initialansicht wird ersetzt durch zwei neue Splitansicht, das erste Trennfeld verbindet die beiden Ansichten. (c) Durch den zweiten Split existieren nun drei Splitansichten, jede einzelne durch ein Trennfeld von den anderen Ansichten getrennt. . . . .	25
4.5	Das Verhalten von zwei Trennfeldern beim Scrollen in den Ansichten lässt sich anhand vier Hauptfällen erläutern. Initial wird angenommen, dass sowohl in der oberen, als auch in der unteren Ansicht ein wenig gescrollt wurde, sodass beide Bänder etwas abgerundet wurden. (a) Das blau markierte Segment bewegt sich vom grün markierten Segment weg, sodass die Distanz steigt und die blauen Bänder abgerundet werden. (b) Analog zum Fall a. (c) Das grün markierte Textsegment verschiebt sich nach oben und nähert sich so dem blauen Segment an und bewegt sich vom roten Segment fort. Die blauen Bänder werden dadurch gestrafft, die orangenen Bänder werden abgerundet. (d) Das grün markierte Textsegment verschiebt sich nach unten und nähert sich so dem roten Segment an und bewegt sich vom blauen Segment fort. Die orangenen Bänder werden dadurch gestrafft, die blauen Bänder werden abgerundet. . . . .	27
4.6	Eine lokale Relation, dargestellt durch eine direkte Kante zwischen den Teilnehmern. Die Menüleiste zeigt die bisher ausgezeichneten Annotationen und Relationen inklusive ihrer Tags, die per Button aktiviert werden können. . . . .	32
4.7	Zwei Varianten der Bogendarstellung und ihr Verhalten, wenn mehrere, unterschiedliche Relationen ausgezeichnet werden. (a) Die Relation wird als abgerundeter Bogen dargestellt, der die Überschneidung des Trennfelds teilweise vermeidet. (b) Die Relation wird als eckiger Bogen dargestellt, der die Überschneidung des Textes gänzlich vermeidet. . . . .	34
4.8	Die Darstellung der Relation und ihrer Klasse als Schwimmlinie. Die Relation der Klasse "Koreferenz" erhält eine Schwimmlinie in der entsprechenden Farbe, sodass jedes Annotationspaar, für das diese Relation ausgezeichnet wird, mittels einer Kante mit der Schwimmlinie verbunden wird. Die Kante nimmt die Farbe der Klasse an. . . . .	36
4.9	Statt den Raum auf beiden Seiten der Ansicht frei zu lassen, werden Textansicht und Trennfeld an den rechten Rand verschoben, sodass nun der gesamte Raum zwischen Menüleiste und Textansichten als Visualisierungsfläche für Relationen dient. . . .	37
4.10	Die Kombination der Schwimmlinien und der Bogendarstellung. Über die Buttons "expand" und "reverse" kann zwischen den Darstellungen gewechselt werden. . .	38
4.11	Es wurden die Klassen "Koreferenz" und "Analogie" und Instanzen dieser erstellt. Durch das Drücken des designierten Buttons werden sämtliche Instanzen der Klasse "Analogie" ausgeblendet, inklusive der Schwimmlinie. Die Ausblendung wird dem Nutzer zusätzlich signalisiert, indem die Opazität des Filter Buttons vermindert wird. Die Annotationen verbleiben. . . . .	40
4.12	Durch die fixe Positionierung der Randfelder wird jede der Splitansichten nach dem Split oben und unten jeweils durch eine Grenzfläche eingeschlossen. Die Breite der Randfelder wird gleich den Textansichten gewählt. . . . .	42



4.13	Es wurden die Klassen "Koreferenz" (orange) und "Analogie" (rot) und Instanzen dieser erstellt. Durch das Scrollen in beiden Splitansichten verändert sich die Position der relationierten Annotationen: In der oberen Ansicht verschwinden sowohl das Mitglied der "Koreferenz" Relation, als auch das Mitglied der "Analogie" Relation im Bereich des oberen Randfelds. Das Diagramm zeigt diesen Sachverhalt in dem Randfeld an. In der unteren Ansicht gilt Selbiges nur für das Mitglied der "Koreferenz" Relation, entsprechend zeigt Trennfeld nur einen Balken. . . . .	42
5.1	Die finale Benutzeroberfläche des entwickelten Ansatzes. . . . .	48
6.1	Die Menüleiste nach der Erstellung der Annotationsklasse "Person" . . . . .	52
6.2	Das Trennfeld wird in den Text gezogen und am Wort "Lage" losgelassen. Wegen dem "next"-Modus wird der Text dann am nächsten Satzende getrennt. Anschließend werden die drei Personenerwähnungen annotiert. Die Erwähnungen "Gregor" und "Samsa" werden dabei ausgelassen, da sie dieselben Referenzen mit "Gregor Samsa" teilen. . . . .	52
6.3	Für die Klasse "Gregor Ref." wird die Farbe grün und für "Dame Ref." die Farbe blau gewählt, wie anhand der erstellten Klassen-Buttons zu erkennen ist. In der Folge werden zuerst die Referenten von Gregor Samsa und dann die Referenzen von Dame annotiert. Entsprechend ist die Klasse "Dame Ref." zuletzt aktiv, was an der Hervorhebung des korrespondierenden Klassen-Buttons zu erkennen ist. . . .	53
6.4	Die Benutzeroberfläche nach der Erstellung der Relationenklasse "Koreferenz". In der Menüleiste wurde der Relationenmodus aktiviert, anschließend wurden "Gregor Samsa" und "Dame" jeweils mit ihren Referenten verbunden. . . . .	54
6.5	Durch das Führen der Maus über "Gregor Samsa" wird die Opazität der annotierten Referenten und der Verbindungen erhöht, damit diese klar hervorstechen. Dasselbe Ziel verfolgend, wird darüber hinaus die Dicke der Verbindungen etwas erhöht. Der schwarze Kreis dient als Indikator für die Mauszeigerposition. . . . .	54
6.6	Die letzte Erwähnung innerhalb der unteren Ansicht ist mittels einer roten Umrandung hervorgehoben. . . . .	55
6.7	(a) Der Text nach Abschluss der langen Annotation. Die gesamte Textmenge zwischen den beiden Erwähnungen wurde annotiert. (b) Nach anklicken des "Merge Views"-Buttons werden die beiden Ansichten und ihre Texte wieder zusammengeführt und das Trennfeld an seine ursprüngliche Position zurück versetzt. Die lange Annotation wird zusammenhängend angezeigt. . . . .	56
6.8	Der Text wurde aufgeteilt, anschließend wird in der unteren Ansicht nach unten gescrollt. Durch die steigende Distanz werden die Bänder abgerundet. . . . .	57
6.9	Die Relationen der Klasse "verwandt mit" wurden zwischen Erwähnungen von "Gregor", "Vater" und "Eltern" ausgezeichnet. . . . .	57
6.10	Durch das Fahren der Maus über eine Erwähnung von "Gregor", werden die Erwähnungen von "Eltern", mit denen "Gregor" verbunden ist und die Verbindungen selbst visuell hervorgehoben. Der rote Kreis dient als Indikator für die Mauszeigerposition. . . . .	58
6.11	Die Relationen der anderen beiden Klassen wurden nun auch eingezeichnet. Aktuell aktiv ist dabei die Klasse "angestellt bei", was an ihrem Klassen-Button und ihrer Schwimmlinie zu erkennen ist. . . . .	58

6.12	(a) Wechsel in die Bogendarstellung für die Klasse "verwandt mit". (b) Wechsel in die Bogendarstellung für die Klasse "arbeitet für". (c) Wechsel in die Bogendarstellung für die Klasse "angestellt bei". In jedem der Wechsel werden die Schwimmlinien an den linken Rand der Visualisierungsfläche verschoben, um den Bögen genügend Raum zu bieten. . . . .	59
6.13	Die gleichzeitige Darstellung aller drei Klassen in der Bogendarstellung. . . . .	60
6.14	Die Benutzeroberfläche nach der Ausblendung der Klasse "verwandt mit" mithilfe des Filter-Buttons neben dem Klassen-Button. Die Ausblendung der Klasse wird zusätzlich durch die Verringerung der Opazität des Filter-Buttons signalisiert. . .	60
6.15	Die Benutzeroberfläche nach der Erstellung der fünf Annotationsklassen und der vier Relationenklassen. . . . .	61
6.16	Die Anzeige der Balkendiagramme wird angepasst, da nun nicht sichtbare relationierte Annotation existieren. Entsprechend werden drei verschiedene Balken angezeigt. . . . .	62
6.17	das Highlight hebt die Verbindung und die relationierte Annotation hervor. Der rote Stern dient als Indikator für die Mauszeiger Position. . . . .	62

# 1 Einleitung

## 1.1 Motivation

Das Annotieren von Texten, ob kurze oder lange Textsegmente, stellt eine Kernaufgabe der Textanalyse zur Aufbereitung und zum Verständnis von Texten dar. In Form von Hervorhebungen, Unterstreichungen, Anmerkungen und Weiterem werden Metadaten über den Text in den Text integriert, ohne diesen selbst explizit zu verändern. Im Kontext der digitalen Annotation stehen heutzutage zahlreiche Werkzeuge zur Verfügung. Zumeist in einer scrollbaren Ansicht dargestellt, ermöglichen diese Werkzeuge das Editieren des Textes an Ort und Stelle, beispielsweise durch farbliches Hervorheben von Textbereichen und das Anbringen von Fußnoten unter oder oberhalb des Textes. Die zugrundeliegende Form der interaktiven Bearbeitung und Visualisierung ist für grundlegende Annotationsaufgaben ausreichend, doch stößt sie an ihre Grenzen, wenn Annotationen in Quantität und Größe wachsen oder Relationen zwischen Annotationen miteinbezogen werden sollen. Zu solchen Fällen gehören

- die Annotation von großen Textbereichen
- Betrachtung und Bearbeitung von Annotationen in weit entfernten Textbereichen
- die Auszeichnung von Relationen zwischen Annotationen, sei es innerhalb eines oder zwischen mehreren Textabschnitten
- oder die Erstellung und Darstellung vieler, verschiedener Typen von Annotationen und Relationen

Die genannten Anwendungsfälle bringen individuelle Herausforderungen für die Visualisierung und Interaktivität mit sich, sei es die Darstellung der Distanz zwischen relationierten Annotationen oder die Vermeidung von Überlagerung bei der Anzeige von Relationen. Zur Lösung solcher und ähnlicher Aufgaben müssen die bereits bestehenden Konzepte zur Visualisierung sinnvoll und interaktiv erweitert werden.

## 1.2 Zielsetzung

Um eine sinnvolle Lösung zur Erweiterung der bestehenden Techniken und Werkzeuge der Textdarstellung und Textannotation zu finden, müssen Ansätze entwickelt werden, welche die Werkzeuge um eine visuell verständliche Darstellung relationierter Annotationen und der Entfernung zwischen den Annotationen ergänzen. Im Vordergrund stehen dabei die Skalierbarkeit hinsichtlich zu bearbeitender Texte und auszuzeichnender Relationen, sowie dem damit verbundenen, möglichst gering zu haltenden Informationsverlust durch Überlagerung vieler, verschiedener Relationen.

Ein entsprechender Ansatz soll im Rahmen dieser Arbeit entwickelt, umgesetzt und bewertet werden, der eine geeignete Darstellung des Textes findet, in welcher die Distanz zwischen entfernten Textabschnitten abgelesen werden kann. Zur Bearbeitung des Textes sollen das Auszeichnen von Annotationen und Relationen verschiedener Typen in einer oder mehreren Textansichten ermöglicht und Interaktionen mit den Ansichten hinsichtlich Korrelation und Distanz dynamisch visualisiert werden. Die Visualisierung der Relationen soll dabei eine geeignete Lösung zur Vermeidung von Überschneidungen und Überlagerungen von Relationen und dem damit verbundenen Informationsverlust finden. Die verschiedenen Komponenten sollen kompakt in einer übersichtlichen Benutzeroberfläche präsentiert werden, welche einen interaktiven Zugriff auf die Komponenten bietet.

## 2 Grundlagen

Im Folgenden werden die notwendigen Grundlagen für den in dieser Arbeit entwickelten Ansatz betrachtet. Dabei werden zunächst die wichtigsten Grundlagen der Informationsvisualisierung angerissen, ehe eine kurze Übersicht über die Textvisualisierung erstellt wird. Abschließend wird das grundlegende Verständnis von Annotationen und Relationen zwischen Annotationen besprochen, da die Visualisierung von relationierten Annotationen zentral im Fokus dieser Arbeit steht.

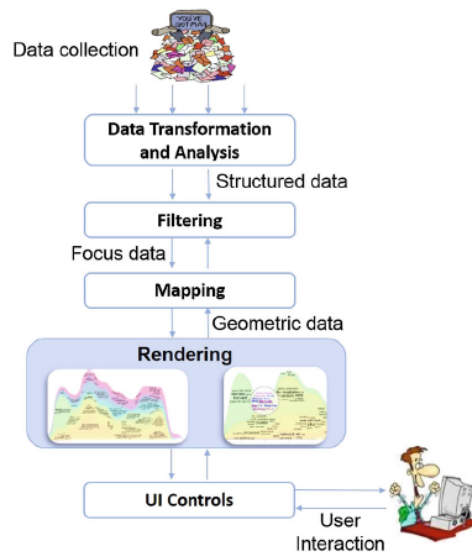
### 2.1 Grundlagen der Informationsvisualisierung

Die Visualisierung von Informationen unterschiedlichster Natur ist eine der grundlegenden Aufgaben vieler Wissenschaften. Daten werden in sehr großen Mengen gesammelt und müssen anschließend verarbeitet, analysiert und bewertet werden. Die Visualisierung ist ein essentieller Bestandteil dieses Prozesses, der dabei hilft, die zugrundeliegende Information verständlich zu repräsentieren.

Zentral ist der Begriff der Informationsvisualisierung, den Chen [Che10] als die Beschäftigung mit Computer-generierter, interaktiver und grafischer Repräsentation von Information beschreibt. Die Informationsvisualisierung arbeite mit abstrakten, nicht-räumlichen Daten, die dann in einem kreativen Prozess in eine intuitiv verständliche, grafische Repräsentation der Informationen umgewandelt werden. Sowohl Chen, als auch Munzner [MH00] grenzen die Informationsvisualisierung von der Wissenschaftlichen Visualisierung ab, indem sie der zuletzt genannten Visualisierung eine größere Rolle in der Verarbeitung von inhärent räumlichen Daten zuweisen. Aufgrund ihrer Größenausmaße überschreiten solche Daten das menschliche Verständnis, sodass sie entsprechend repräsentiert werden müssen. Die Daten der Informationsvisualisierung sehen sie abstrakter und nicht zwangsläufig quantifizierbar. Aufgabe der Informationsvisualisierung sei es dann, mithilfe von Algorithmen solche Daten in eine quantifizierbare und visuell darstellbare Form zu transformieren.

Der Prozess der Umwandlung abstrakter Daten in eine interaktive, visuelle Repräsentation lässt sich anhand der *Information Visualization Pipeline* modellieren. Liu et al. [LCWL14] machen in Abbildung 2.1 5 Kernmodule der Pipeline aus. Von den genannten Modulen spielen allen voran die Transformation der Rohdaten in ein strukturiertes Datenformat, die Zuweisung der Daten auf eine geeignete visuelle Form gefolgt von der passenden Darstellung, sowie die interaktiven Möglichkeiten des Nutzers zur Betrachtung und Manipulation der präsentierten Informationen, eine signifikante Rolle in dieser Arbeit.

Chen beschreibt den Begriff der Erkenntnis (*Insight*) als elementares Ziel der Informationsvisualisierung. Während zuvor die sofortige Auffindung relevanter Informationen in einer Visualisierung wichtig war, spiele die interaktive Suche nach relevanten Informationen und die durch die Visualisierung zur Verfügung gestellten Möglichkeiten für die Suche inzwischen eine immer größere Rolle. In diesem Rahmen stelle sich zudem oft die Frage, welche Information zu welchem Preis in



**Abbildung 2.1:** Die Information Visualization Pipeliien und ihre 5 Schritte zur Umwandlung der Daten in die visuelle Repräsentation nach Liu et al. [LCWL14]

welcher Form präsentiert wird und welche Erkenntnis durch die Visualisierung der Information erreicht werden kann. Als eine der Aufgaben mit welcher sich der Bereich der *Visual Analytics* beschäftigt, kommen solche Entscheidungsfindungen auch in dieser Arbeit zum Tragen.

Bei der Entscheidungsfindung helfen nach Shrinivasan und van Wijk [SW08] drei verschiedene Typen von visueller Repräsentation:

- In der Datensicht (*data view*) befinden sich die visuelle Darstellung der Informationen und Werkzeuge zur Interaktion mit der Darstellung.
- Die Navigationssicht (*navigation view*) ermöglicht die Exploration der visuell dargestellten Informationen, wobei die verschiedenen Zustände der sich im Rahmen der Interaktion verändernden Visualisierung erfasst werden.
- Die Wissensbasierte Sicht (*knowledge view*) präsentiert die kausalen Verbindungen zwischen den visuellen Elementen.

In der Summe kann die Visualisierung durch die interaktiven Veränderungen in verschiedenen Zuständen betrachtet und untersucht werden. Die Zustände können dann untereinander verglichen, die Gemeinsamkeiten und die Unterschiede erfasst und letztlich analysiert werden. Über das Zusammenspiel der verschiedenen Sichten lassen sich so Schlüsse ziehen, die eine einzelne Repräsentation nicht ermöglicht.

Zuletzt soll noch der Begriff des *Visual Encoding*, auf deutsch Visuelle Enkodierung, betrachtet werden, der von Munzer [MH00] als Abbildung von Informationen auf visuelle Elemente beschrieben wird. Die visuellen Elemente dienen zur Veranschaulichung der Zusammenhänge der repräsentierten Daten und ergeben nach Munzner in Kombination eine visuelle Metapher. Kodiert werden Informationen typischerweise in der Position der visuellen Elemente, ihrer Größe, ihrer Farbe, der Helligkeit dieser oder auch in ihrer Form. Insbesondere lassen sich Veränderungen an den Daten als Veränderung der zuvor genannten Attribute repräsentieren.

## 2.2 Grundlagen der Textvisualisierung

Die Textvisualisierung als eine der Anwendungsgebiete verschiedener Techniken der Informationsvisualisierung, beschäftigt sich mit der Verarbeitung von teils sehr großen Textmengen und der Darstellung der hieraus entnommenen Informationen. Cao und Cui [CC16] benennen drei verschiedene Arten von Texten, mit welchen sich die Textvisualisierung hauptsächlich, unter Betrachtung unterschiedlicher Aspekte, beschäftigt:

- Die Visualisierung von Dokumenten, beispielsweise Zeitungsartikel oder Webseiten, konzentriert sich auf die Repräsentation einzelner Dokumente und der vergleichenden Darstellung von Dokumenten, sei es im Rahmen von Verwandtschaften zwischen Dokumenten oder hinsichtlich sprachlicher oder inhaltlicher Eigenschaften.
- Ganze Kollektionen von Dokumenten werden statistisch ausgewertet und die resultierenden Ergebnisse anschließend visuell präsentiert.
- Ströme von Textnachrichten oder Texten, wie sie beispielsweise in den sozialen Netzwerken zu finden sind, werden aufgrund der kontinuierlichen Produktion dieser auf ihr Muster und auf Veränderungen über bestimmte Zeitspannen untersucht.

Besonders im Fokus stehen in dieser Arbeit einzelne Texte. Nualart et al. [NPW14] spezifizieren, dass die Texte trotz ihrer zumeist eindimensionalen, sequentiellen Natur über eine interne Struktur verfügen, die eng zusammenhängt mit der Struktur ihrer Information. Der Text und seine einzelnen Strukturen wie Überschriften oder Kapitel lassen sich im Rahmen der Textvisualisierung wiederum in kleinere Einheiten aufbrechen und visuell veranschaulichen. Ein Beispiel dafür ist die Word Cloud zur Darstellung der Schlagwörter eines Dokuments, wo unter anderem die Häufigkeit der Wörter in der Darstellung kodiert ist.

Abhängig vom Schwerpunkt des Interesses werden unterschiedliche Techniken zur Visualisierung benötigt. Mögliche Interessen sind dabei die zusammenfassende Darstellung eines Textes, die visuelle Exploration seiner Inhalte auf den verschiedenen Strukturebenen des Textes oder die Visualisierung von Metadaten und relationalen Inhalten. Häufig genutzt werden nach Cao und Cui Bäume und Graphen zur Darstellung unterschiedlicher Aspekte des Textes. Als Beispiel nennen sie die Repräsentation eines Textes und seiner inneren Strukturen als Baum. Ähnlich verfährt die Darstellung des Textes als Graph, dessen Kanten zwischen den Knoten Beziehungen oder semantische Abhängigkeiten zwischen den durch die Knoten dargestellten Elementen des Textes wiedergeben. Auch auf der Wortebene werden Bäume und Graphen zur Darstellung von Daten verwendet, wo sie zumeist Wortvorkommen, Häufigkeiten oder Satzkonstruktionen visualisieren.

## 2.3 Grundlagen der (relationierten) Annotation

Texte können auf unterschiedliche Arten, implizit oder explizit, manipuliert werden, beispielsweise um sie zusammenzufassen zu verkürzen oder um die bestehenden Informationen zu ergänzen. Die Veränderung der textuellen Daten und ihrer Eigenschaften müssen dann verständlich dargestellt werden. Im Folgenden soll die Annotation von Texten als Fallbeispiel einer solchen Manipulation

des Textes und als ein zentraler Aspekt dieser Arbeit behandelt werden. Nachdem das Verständnis von Annotationen und Relationen zwischen Annotationen angerissen wurde, sollen schließlich verschiedene Techniken zur Darstellung von Relationen besprochen werden.

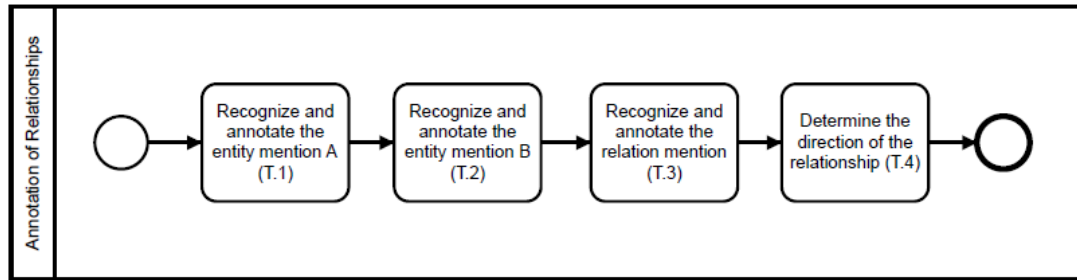
Annotationen lassen sich nach Burghardt [Bur14] als die Hinzunahme von Information zu einem bereits bestehendem Textinhalt beschreiben. Die Information kann dabei implizit im Text integriert und durch die Annotation hervorgebracht oder explizit, beispielsweise in Form von Referenzen, hinzugefügt werden. Je nach Anwendungsbereich weisen die verschiedenen Arten von Annotationen viele Gemeinsamkeiten, aber auch einige sich unterscheidende Eigenschaften auf:

- Burghardt unterteilt Annotationen im allgemeinen in zwei Kategorien: Die strukturellen Annotationen zur Beschreibung der Struktur eines Textes und die deskriptiven Annotationen welche den Inhalt des Textes, beispielsweise bestimmte Vorkommen von Wörtern, beschreiben.
- Eine einzelne Annotation lässt sich dann aufteilen in den Inhalt, den sie verkörpert, sowie den Anker, an welchem sie im Text angebracht wird. Im Bereich der digitalen Annotationen können sowohl Inhalt, als auch Anker je nach Anforderung und Nutzung unterschiedlich gewählt und visualisiert werden: Der anzubringende Inhalt kann explizit, beispielweise als Kommentar, dargestellt werden, er kann aber auch implizit in der Visualisierung realisiert sein. Mögliche Visualisierungen sind dabei die farbliche Hervorhebung des Ankers, zum Beispiel in Form eines Highlights oder durch die Unterstreichung oder Umkringelung des Ankers. Der Anker kann hierbei ein konkretes Objekt im Text sein, wie einzelne Wörter, Überschriften oder ganze Textsegmente. Alternativ kann der Inhalt der Annotation aber auch zwischen den Zeilen oder an den Rändern des Textes, zum Beispiel in der Form eines Kommentars, angebracht werden.
- Nach der Auszeichnung füllen Annotationen Informationslücken im Text, d.h. sie ergänzen diesen explizit oder sie fungieren als erweiternde Metadaten, welche bereits existierende Informationen weiter spezifizieren. Als mögliche Gründe zur Ergänzung oder Erweiterung des Textes benennt Burghardt unter anderem das Anbringen von Interpretationen des Nutzers, die Zusammenfassung oder Vereinfachung des Inhalts oder auch die Kommunikation mit anderen Annotatoren über die Annotationen.

Gerade mit digitalen Annotationen, welche mithilfe elektronischer Werkzeuge erstellt werden, beschäftigt sich diese Arbeit. Digitalen Annotationen können dabei automatisch, nach einem vordefiniertem Schema, generiert oder manuell durch einen Nutzer ausgezeichnet werden. Die zu annotierenden Textobjekte können dabei vielfältig sein und hängen mit der Annotationsaufgabe zusammen. Beispiele für solche Aufgaben sind die Annotation bestimmter Wortvorkommen oder von Worthäufigkeiten, aber auch die Annotation von textuellen oder semantischen Beziehungen zwischen Textobjekten.

Solche Beziehungen können in Texten zwischen verschiedenen Entitäten vorherrschen, beispielsweise zwischen bestimmten Wortgruppen, Textsegmenten oder ganzen Paragraphen. Die Annotation von Relationen beinhaltet somit die Identifikation der Teilnehmer einer Relation, die Identifikation der passenden Relation und die Auszeichnung der Relation. Breitenfeld und Müller-Birn [BM17] beschreiben dieses Vorgehen zur Annotation von Relationen in Abbildung 2.2.



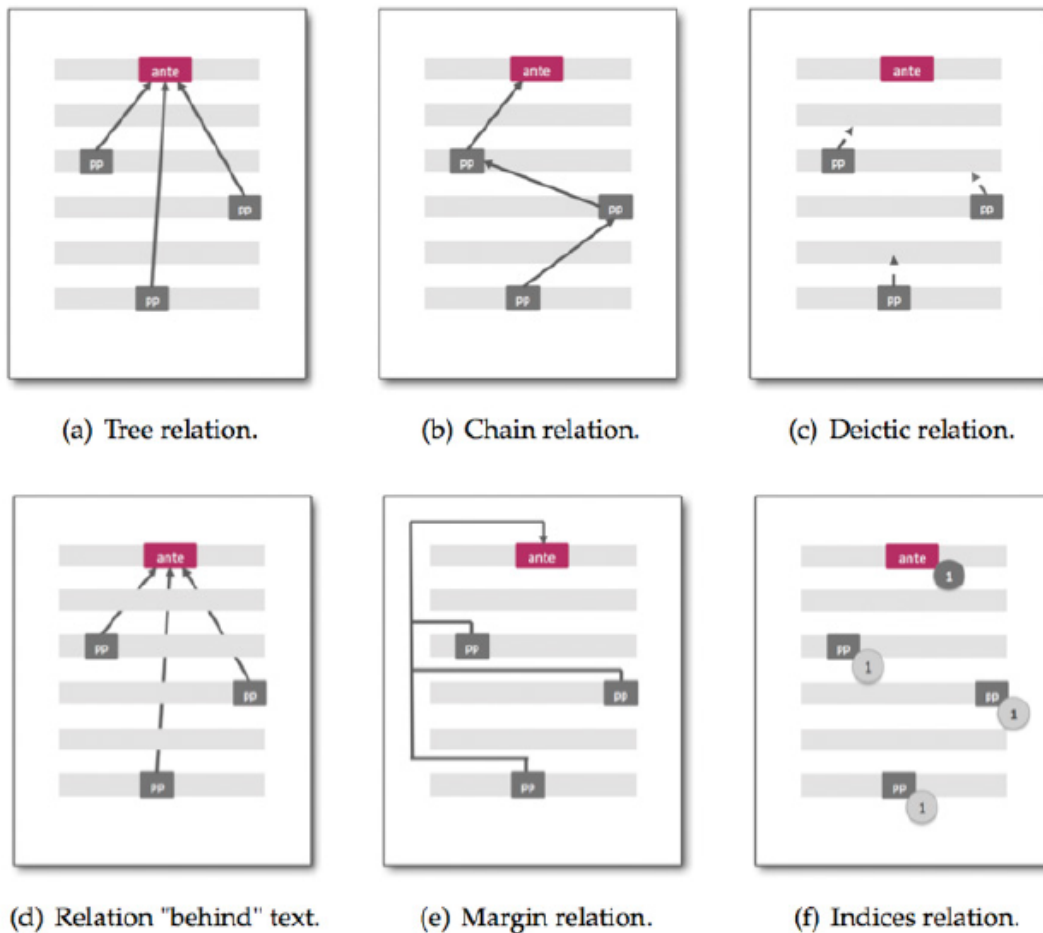


**Abbildung 2.2:** Die Identifikation der Teilnehmer beschreiben Breitenfeld und Müller-Birn in dieser Abbildung [BM17] als *named-entity-recognition*, dargestellt in den Aktionen T.1 und T.2. Im letzten Schritt wird die Richtung der Relation bestimmt, falls diese gerichtet ist, beispielsweise in einer Ursache-Folge-Beziehung. Alternativ kann die Relation ungerichtet sein, sodass Schritt T.4 ausgelassen werden kann.

Die Relation ist somit selbst eine Annotation, deren Anker die Teilnehmer der Relation sind und deren Inhalt die Darstellung und gegebenenfalls eine textuelle Beschreibung der Beziehung ist. Dabei können die Teilnehmer selbst wiederum annotierte Textobjekte sein, sodass die Relation die Beziehung zwischen Annotationen darstellt.

Ähnlich den Annotationen, existieren zur Visualisierung von Relationen verschiedene Möglichkeiten. Abbildung 2.3 zeigt einige der von Burghardt [Bur18] benannten Methoden zur Darstellung von Relationen zwischen annotierten Textsegmenten. Die unterschiedlichen Ansätze werden im Folgenden kurz beschrieben und ihre Stärken und Schwächen bezüglich verschiedener Aspekte der Informationsdarstellung untersucht:

- Die in a) gezeigte Baumstruktur verbindet die relationierten Annotationen direkt im Text mit dem Referenten über eine gerichtete oder ungerichtete Kante. Besonders im Fokus steht hier der Referent, welcher als zentrales Element agiert. Diese Darstellung bietet den Vorteil, dass die Verbindungen auf einen Blick ersichtlich sind und nachverfolgt werden können. Jedoch bergen sie den Nachteil, dass die verbindenden Kanten über den darunter liegenden Text verlaufen, sich mit diesem überschneiden und dadurch die Lesbarkeit des Textes mindern.
- Die in b) gezeigte Referenzkette ähnelt der Baumstruktur in einigen Aspekten wie der direkten Verbindung oder der Überschneidung des Textes. Hier jedoch sind die einzelnen Verbindungen kürzer, da sie zur nächstgelegenen Referenz führen, wodurch sie sich mit weniger Text überschneiden. Auch wird das Auffinden benachbarter Referenzen dadurch erleichtert. Im Austausch dafür bietet diese Darstellung eine schlechtere Übersicht über alle relationierten Annotationen. Das Auffinden des Referenten am Anfang der Kette beispielsweise erfordert die Nachverfolgung der Kette bis an die Spitze.
- Die in c) gezeigte Andeutung der Relation über Pfeile, die in Richtung des Bezugsworts zeigen, bietet keine direkte Verbindungen wie in den vorangegangenen Varianten an, minimiert dafür aber im Vergleich die Überschneidung des Textes. Zum Auffinden von Referenzen ausgehend von einer Annotationen fehlen nun jedoch die eben genannten, visuellen Verbindungen.



**Abbildung 2.3:** Den Rahmen für die verschiedenen Darstellungen nach Burghardt [Bur18] bildet die Koreferenz-Annotation, bei der mehrere annotierte Pronomen einen ebenfalls annotierten Referenten, den sie innerhalb des Textes vertreten, referenzieren.

- Ansatz d) optimiert a) hinsichtlich der Vermeidung der Überschneidung des Textes, büßt dafür aber etwas in der Verfolgbarkeit und an Übersichtlichkeit ein, da der Nutzer bei der Nachverfolgung der Kanten durch die Lücken unterbrochen wird.
- Die Nutzung des Zeilenzwischenraums in e) vermeidet ähnlich d) die Überdeckung des Textes gänzlich, bietet dabei jedoch eine visuell zusammenhängende Darstellung an. Jedoch ist hier zurückgelegte Strecke länger. In der Summe vermindert dies die Übersicht über die vorhandenen Referenzen und ihre Verbindungen zueinander.
- Ansatz f) ähnelt Ansatz c) stark, jedoch werden hier die relationierten Annotationen über Indizes referenziert. Die Überschneidung des Textes wird erneut vermieden, zudem ermöglichen die Indizes Aussagen über die Anzahl der involvierten Referenzen. Dafür fehlt der visuelle Bezug zwischen den relationierten Annotationen, sodass ausgehend von einer Annotation keine Aussagen über die Position der anderen Referenzen getroffen werden können.

Die verschiedenen Stärken und Schwächen zeigen auf, dass eine perfekte Visualisierung von Relationen nicht existiert. Je nach Struktur der Relationen und ihrer Teilnehmer, sowie der konkreten Anwendung, können unterschiedliche Schwerpunkte gesetzt werden: Die Übersicht über eine Relation und ihre Teilnehmer, die Verfolgbarkeit der einzelnen Verbindungen, Positionsaussagen ausgehend von individuellen Annotationen, Vermeidung von Überdeckung des Textes oder auch die Vermeidung von Überschneidung und Überlagerung zwischen verschiedenen Relationen.



# 3 Verwandte Arbeiten

Im Folgenden sollen einige verwandte Arbeiten betrachtet werden. Im Fokus stehen Arbeiten, die sich mit der Auszeichnung und Visualisierung von relationierten Annotationen beschäftigen. Die Arbeiten werden mit dem in dieser Arbeit vorgestellten Ansatz verglichen und von ihm abgegrenzt.

## 3.1 ANALEC

Landragin et al. präsentieren in [LPV12] mit ANALEC einen Ansatz zur dynamischen Annotation von Texten und der grafischen Visualisierung annotierter Texte. Der Ansatz erlaubt dabei das Annotieren von Texten, die Visualisierung der Annotationen, die Darstellung und Suche von Relationen zur Analyse dieser und die geometrische Repräsentation annotierter Daten. Die vorgestellten Beispiele stützen sich auf die Annotation von semantischen Rollen in Reden und die Verlinkung von Referenzen innerhalb der Reden. Abbildung 3.1 zeigt die im Folgenden besprochenen Teile der Benutzeroberfläche von ANALEC.

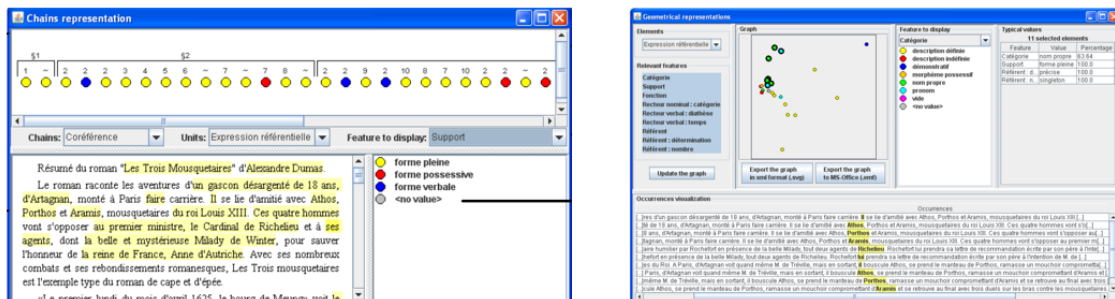


Abbildung 3.1: Links die Darstellung der Annotationen als Verkettung, Rechts die geometrische Ansicht von ANALEC [LPV12]

Zur Bearbeitung des Textes klassifiziert der Ansatz drei verschiedene Entitätstypen anhand eines Metamodells, die allesamt annotiert werden können: Einheiten (*units*) repräsentieren kleinere Texteinheiten, während Relationen (*relations*) solche Texteinheiten in Beziehung setzten. Zudem können Texteinheiten und Relationen innerhalb komplexerer Strukturen kombiniert oder verkettet werden, sodass das kombinierte Produkt Schema (*schema*) genannt wird. Der Ansatz präsentiert die Entitätstypen in unterschiedlichen Ansichten, um die Überlagerung bei der Visualisierung der verschiedenen Typen zu vermeiden. Mein Ansatz unterscheidet sich hiervon, indem Annotationen und Relationen als unterschiedliche Entitäten betrachtet und entsprechend verschiedene Ansätze zur

Darstellung verwendet werden. Komplexere Strukturen als binäre Relation und ihre Teilnehmer werden in meinem Ansatz jedoch nicht betrachtet. Zudem beschränkt sich mein Ansatz auf eine Nahansicht des Textes und seiner Annotationen und Relationen.

In einer ersten Ansicht stellt ANALEC annotierte Texteinheiten eines bestimmten Typs sowohl innerhalb des Textes, als auch separat als lineare Verkettung dar. Annotationen desselben Typs, die Teil einer Relation sind, bilden dabei eine Kette. Für Koreferenzen als Beispielrelation, wird so eine Referenzkette aus dem Referenten und den abhängigen Referenzen erstellt. Die verschiedenen Typen der Relationen lassen sich dadurch in ANALEC isoliert voneinander betrachten. Der Nutzer wählt dabei über verschiedene Drop-down-Menüs aus, Einheiten welchen Typs im Rahmen welcher Relation als Ketten angezeigt werden sollen. Während die Annotationen dann in der Textdarstellung in der vom Nutzer zugewiesenen Farbe hervorgehoben werden, werden die linearen Ketten von Knoten, welche die Annotationen repräsentieren, in einem separaten Feld dargestellt. Für das Beispiel der Koreferenzen wird über jedem Knoten zusätzlich angezeigt, wie viele abhängige Referenzen sich auf den Referenten beziehen.

Im Vergleich zu ANALEC stellt mein Ansatz Annotationen und Relationen innerhalb derselben Ansicht dar. Relationen werden dabei innerhalb der Textansichten gezeichnet, sodass der Nutzer nicht zwischen mehreren Ansichten wechseln muss. Ähnlich ANALEC hebt auch mein Ansatz annotierte Textsegmente farblich im Text hervor, jedoch greift mein Ansatz auf die direkte Visualisierung von Relationen im Text über verbindende Kanten zurück. ANALEC integriert die Filterung relevanter Annotationen und Relationen, indem diese individuell ausgewählt und in einer Tabelle dargestellt werden. In einer ähnlichen Weise bietet mein Ansatz eine interaktive Filterfunktion an. Mithilfe dieser Funktion können beliebige Typen von Relationen, in meinem Ansatz Klassen genannt, und ihre Instanzen visuell aus- und eingeblendet werden. Anders als ANALEC, verwendet mein Ansatz zwei verschiedene Darstellungen zur Visualisierung von Relationen. Relationen einer bestimmten Klasse können somit entweder explizit anhand der Filterfunktion isoliert werden oder durch den Wechsel der Darstellung nur für diese Klasse visuell hervorgehoben werden. Im Gegensatz zu ANALEC gibt es in meinem Ansatz jedoch keine Möglichkeit, Annotationen auszublenden.

Eine weitere Ansicht wird durch ANALEC verwendet, um eine geometrische Repräsentation der annotierten Daten zu erstellen. Dabei werden Annotationen, die in einem ähnlichen Kontext auftreten, innerhalb eines Graphen gruppiert, um den Kontext visuell darzustellen. Der Kontext und die Eigenschaften der Annotationen, nach denen gruppiert werden soll, können dabei vom Nutzer selbst ausgewählt und angepasst werden. Der Nutzer ist somit in der Lage, anhand der Gruppierung auf Relationen zwischen Annotationen bestimmter Typen zu schließen. Unterhalb des Graphen wird dem Nutzer zudem das Vorkommen der betroffenen Annotationen innerhalb des Textes in einem Wortregister angezeigt, welches durch den Nutzer durch das Anklicken der Gruppierungen im Graphen manipuliert werden kann.

Eine dedizierte Ansicht, welche die Annotationen ihrem durch den Nutzer veränderbaren Kontext nach gruppiert, existiert in meinem Ansatz nicht. Jedoch ermöglicht die direkte, visuelle Verbindung von relationierten Annotation in meinem Ansatz, anhand der Verteilung der visuell dargestellten Relationen in Textabschnitten Aussagen über die Zusammensetzung dieser Abschnitte zu formulieren. Möglich wäre es, meinen Ansatz um einige Funktionen der geometrischen Ansicht aus ANALEC zu ergänzen. Eine sinnvolle Erweiterung wäre hierbei, die zuvor beschriebene Filterfunktion durch eine Konfigurationskomponente zu erweitern, sodass der Nutzer anzuzeigende Relationen anhand

eines durch ihn konfigurierten Kontextes organisieren kann. Zur Umsetzung müsste mein Ansatz jedoch erlauben, weitere Metadaten zu den Annotationen und Relationen hinzuzufügen, sodass die Annotationen und Relationen anhand dieser Metadaten organisiert und gefiltert werden können.

## 3.2 Glozz

In [WM12] entwickeln Widlöcher und Mathet einen Ansatz zur Erstellung, Visualisierung und Verarbeitung von relationierten Annotationen in Referenzkorpora. Die Benutzeroberfläche des Systems ist in Abbildung 3.2 zu sehen.

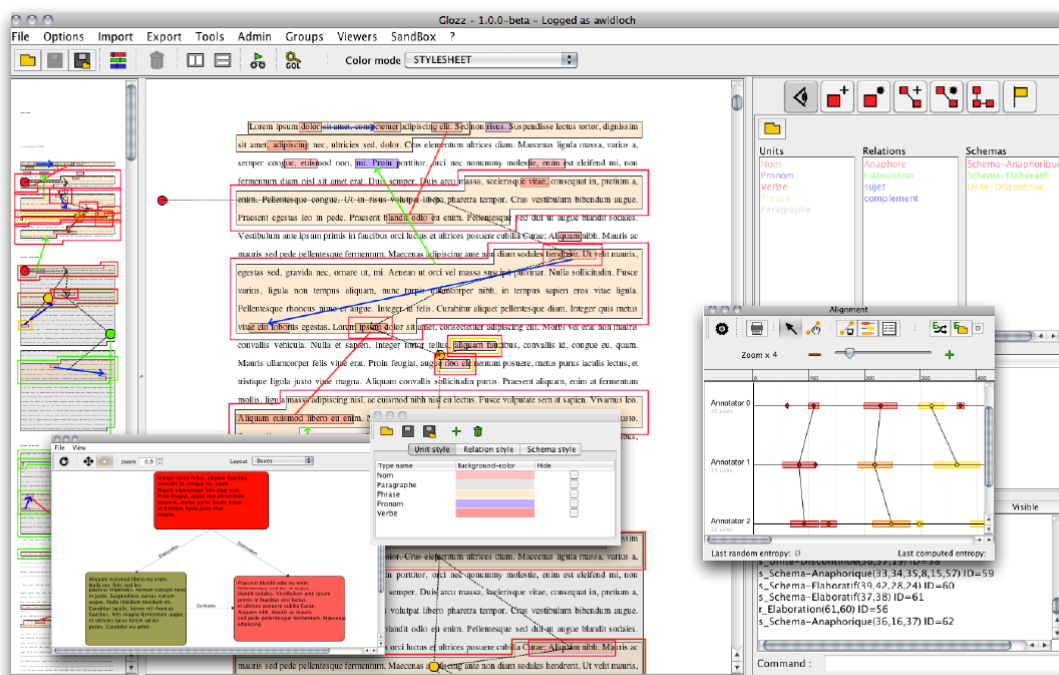


Abbildung 3.2: Die Benutzeroberfläche von Glozz und ihre verschiedenen Ansichten [WM12]

In Glozz werden die zu annotierenden Textobjekte sogenannten Elementen zugeordnet, die sich in *Unit*, *Relation* und *Schema* unterteilen lassen, analog zur Unterteilung in ANALEC [LPV12]. Jedes Element wird dabei durch den vom Nutzer zugewiesenen Typ, den Tag und die Menge an Eigenschaften charakterisiert.

Der zu bearbeitende Text wird innerhalb einer dedizierten Ansicht in einem *default*-Modus, in welchem er nur gelesen werden kann oder in einem Annotationsmodus, in welchem er annotiert werden kann, zentral dargestellt. Zur Annotation existieren drei verschiedene Modi, die sich auf die zuvor beschriebenen Elemente beziehen: *Unit Annotation*, *Relationen Annotation* und *Schema Annotation*. Die Modi erhalten innerhalb der Benutzeroberfläche jeweils ein eigenes Fenster, welches die erstellen Typen anzeigt. Innerhalb der Modi können Annotationen des jeweiligen Elements erstellt oder bestehende Annotation editiert werden.

Glozz unterscheidet sich hier in Teilen von meinem Ansatz, wo der zu bearbeitende Text in einer Ansicht angezeigt wird, welcher aufgeteilt und zur Erstellung von Annotationen und Relationen manipuliert werden kann. Zudem findet in meinem Ansatz keine Unterscheidung zwischen verschiedenen zu annotierenden Textobjekten statt. Stattdessen werden Textsegmente in meinem Ansatz als eine Folge von Tokens betrachtet, welche mit beliebig gewählten Annotationsklassen annotiert werden können. Ähnlich Glozz trennt auch mein Ansatz die Textansicht und die Übersicht über die bestehenden Annotations- und Relationenklassen innerhalb der Benutzeroberfläche räumlich auf.

Ein *Unit* beschreibt in Glozz Textsegmente beliebiger Länge, die im Rahmen einer Annotation visuell im Text in der zugewiesenen Farbe hervorgehoben werden. Dabei annotiert der Nutzer ein solches Unit, indem er zunächst den Typ der Annotation auswählt, ehe er ein einzelnes Wort per Doppelklick annotiert oder mehrere Wörter per Drag-and-drop selektiert. Die Textemnge zwischen den selektierten Wörtern wird dann automatisch Teil der Annotation. Zur Auszeichnung von Relationen wechselt der Nutzer in den Relationenmodus, wo der Nutzer zunächst ebenfalls den Typ der auszuzeichnenden Relation und anschließend zwei Annotationen auswählt, zwischen welchen dann eine binäre Relation des selektierten Typs ausgezeichnet wird. Während die Annotation über die Hervorhebung direkt im Text in der Farbe des Typs visualisiert wird, werden binäre Relationen in Glozz als direkte Verbindung der Annotationen über eine Kante in der Farbe des Typs der Relation realisiert.

In diesen Punkten weist mein Ansatz größere Parallelen zum Ansatz von Widlöcher und Mathet auf. Ähnlich wie in Glozz, werden zunächst die Klassen der Annotationen und Relationen erstellt, für welche aktivierbare Buttons in einer Menüleiste präsentiert werden. Der Nutzer kann dann konkrete Instanzen der verschiedenen Klassen erzeugen, indem er diese abwechselnd aktiviert. Wie auch in Glozz, werden die Annotationen und Relationen direkt im Text angezeigt, wobei Glozz, anders als mein Ansatz, keine Maßnahmen gegen Überschneidungen und Überlagerungen des Textes trifft. Sowohl Glozz, als auch mein Ansatz, ermöglichen die Annotation längerer Textsegmente, indem nicht das gesamte Segment selektiert werden muss. Stattdessen reicht es in beiden Ansätzen aus, den Anfang und das Ende der Annotation zu markieren. Glozz präsentiert sich überschneidende Annotationentypen desselben annotierten Textsegments indem konsekutiv ausgezeichnete Annotationen mit erhöhter Transparenz gezeichnet werden. Im Vergleich hebt mein Ansatz ein solches Textsegment grau hervor, um dem Nutzer die Überschneidung zu signalisieren.

Neben binären Relationen erlaubt Glozz auch mehrstellige Relationen, die ähnlich den binären Relationen ausgezeichnet werden. Visualisiert werden diese dabei, indem die Teilnehmer der Relation nicht direkt verbunden werden, sondern verbindende Kanten von den Teilnehmern zu einem Knoten verlaufen, der sich zentral zu den Teilnehmern befindet. Zudem lassen sich solche Relationen in Glozz auch als graphartige Struktur anzeigen. Eine ähnliche Visualisierung wäre auch für meinen Ansatz denkbar, wo n-äre Relationen nicht unterstützt werden.

Zuletzt ermöglicht Glozz die Bearbeitung eines Textes durch mehrere Annotatoren, indem Annotationen innerhalb eines Editors nach Typ gruppiert und anschließend ein- und ausgeblendet werden können. Ausgeblendete Annotationen werden dem Nutzer dabei innerhalb einer Tabelle angezeigt. Zudem bietet Glozz eine dedizierte Annotatorenansicht, die den Text mitsamt der ausgezeichneten Annotationen und ihrer Position im Text für jeden Annotator als Linie darstellt. Während eine Ein- und Ausblendfunktion für Annotationen einer bestimmten Klasse auch in meinem Ansatz denkbar wäre, lässt sich die Darstellung des Textes in einer Linie zur parallelen Annotationen durch mehrere Annotatoren schwer umsetzen. Mein Ansatz fokussiert sich auf die Bearbeitung des Textes



durch einen einzelnen Annotator, dem dabei Werkzeuge beispielsweise zur Aufteilung des Textes auf mehrere Ansichten zur Verfügung gestellt werden. Solche Werkzeuge wären bei der Darstellung des Textes als Linie schwieriger umzusetzen.

### 3.3 BRAT

Der in [SPT+12] vorgestellte Ansatz von Stenetorp et al. BRAT, befasst sich mit der Annotation und Visualisierung annotierter Daten in verschiedenen Texten im Rahmen Computerlinguistischer Tätigkeiten. Im Fokus stehen dabei komplexere Strukturen, wie größere Mengen an Annotationen von verschiedenen semantischen Typen, dicht bevölkerte Annotationen, welche sich zudem überlagern und Verbindungen zwischen Annotationen. Die Benutzeroberfläche von BRAT ist in Abbildung 3.2 zu sehen.

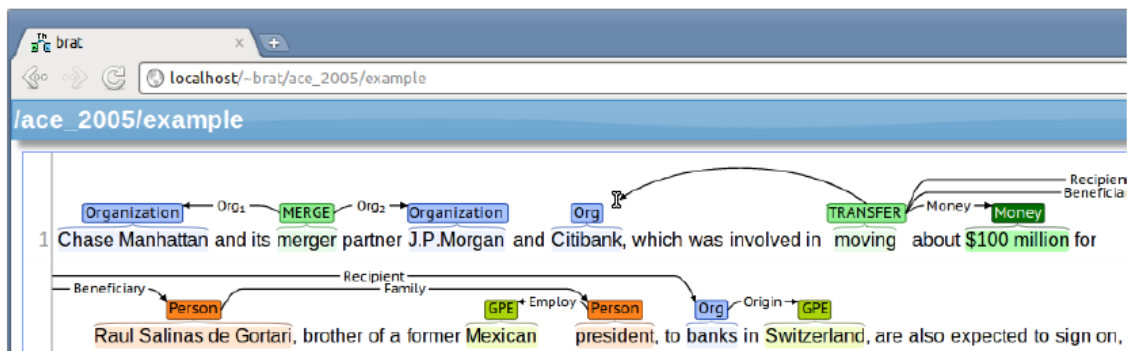


Abbildung 3.3: Die Benutzeroberfläche von BRAT [SPT+12]

Die Erstellung von Annotationen erfolgt in BRAT über das Ziehen des Mauszeigers über die Wortmenge oder über einen Doppelklick der ausgewählten Wortmenge. Jede Annotationen wird dabei einem Typ zugewiesen und kann zudem durch weitere Attribute markiert werden. Zusätzlich kann der Nutzer jeder Annotation Notizen anhängen. Erstellte Annotationen können dann verlinkt werden, indem die Verbindung mit dem Mauszeiger ausgehend von einer Annotation zur anderen gezogen wird. BRAT erlaubt hierbei sowohl binäre als auch n-äre, gerichtete oder ungerichtete Verbindungen. Für jede Relation können dabei über ein Dialogfenster Ursprung, Ziel und Typ der Relation spezifiziert werden. In den bisher genannten Punkten ähnelt BRAT meinem Ansatz teilweise, wo Annotationen durch einen einzelnen Klick einzelner Wörter oder über das Ziehen des Mauszeigers über ganze Segmente erzeugt werden. Ebenfalls wird jeder Annotation und auch jeder Relation eine Klasse sowie ein Tag als Bezeichner zugewiesen, jedoch keine weiteren Daten wie beispielsweise Notizen. Binäre, ungerichtete Relationen der aktuell aktiven Klasse werden in meinem Ansatz durch das Anklicken bereits existierender Annotationen ausgezeichnet, sofern der Nutzer mindestens eine Relationenklasse erstellt hat. Der Informationsgehalt der Annotationen und Relationen in BRAT ist damit größer als in meinem Ansatz.

BRAT integriert zudem ein konfigurierbares System zur Überprüfung semantischer Einschränkungen auf Annotationen und ihren Verbindungen, sodass beispielsweise die Einschränkung deklariert werden kann, dass eine Annoation vom Typ "Transfer" mit einem "Absender" und einem "Empfänger" verbunden sein muss. Werden solche Einschränkungen vom Nutzer aktiviert und in der Folge

nicht eingehalten, wird dies dem Nutzer über eine Fehlernachricht in einem separaten Textfenster signalisiert. Mein Ansatz unterscheidet sich hier von BRAT, da der Nutzer in meinem Ansatz beliebig die Klassen der Annotationen wählen und Instanzen dieser erstellen kann. Diese können dann wiederum durch beliebig gewählte Relationen über beliebige, binäre Kombinationen verbunden werden. Dem Nutzer wird so die komplette Freiheit, aber auch Verantwortung über die erstellten Annotationen und Relationen gelassen, ohne weitere Möglichkeiten zur Einschränkung.

Zur Visualisierung der Annotationen und Relationen strukturiert BRAT den zu bearbeitenden Text zunächst. Der Text wird dazu zeilen- und satzweise vertikal angeordnet, mit etwas mehr Raum zwischen den Zeilen. Annotierte Textsegmente werden farblich hervorgehoben, wobei die Transparenz der Hervorhebung erhöht wird. Über den Annotationen und somit zwischen den Zeilen, wird der zugewiesene Typ in Textform angezeigt und dabei in derselben Farbe mit höherer Opazität hervorgehoben. Relationen zwischen Annotationen werden dann stets innerhalb derselben Textzeile zwischen den Typanzeigen visualisiert. Eine schwarze mit dem Bezeichner der Relation betitelte Kante in der Form eines Bogens verbindet die Typanzeigen im Zeilenzwischenraum. In meinem Ansatz wird der zu bearbeitende Text innerhalb einer Ansicht strukturlos in einer Sequenz dargestellt. Eine konkrete Strukturierung ist nicht von Nöten, da, im Gegensatz zu BRAT, die Klassenbezeichner der verschiedenen Annotationsklassen als aktivierbare Buttons in einer separaten Menüleiste angezeigt werden. Hierdurch wird Raum zwischen den Zeilen im Text gespart. Diesen Raum nutzt mein Ansatz ähnlich BRAT für die Visualisierung der Relationen. Anders als in BRAT jedoch, wird jede Verbindung in der Farbe der zugehörigen Klasse gezeichnet, sodass ein Bezeichner an den Verbindungen nicht nötig ist. Während die Verbindungen in BRAT stets als bogenartige Kante dargestellt werden, welche gegebenenfalls die Textzeilen darüber überschneidet, werden relationale Kanten in meinem Ansatz, sofern möglich, strikt durch den Zeilenzwischenraum geführt, um Überschneidungen mit dem Text zu vermeiden. Mein Ansatz erlaubt zudem Relationen an beliebigen Stellen im Text und in einer oder in verschiedenen Ansichten. Eine letzte Gemeinsamkeit weisen BRAT und mein Ansatz auf, indem beide Ansätze die Überlagerung von visuell dargestellten Relationen durch eine kleine Verschiebung konsekutiv gezeichneter Relationen vermeiden. Während BRAT dies für die Relationen innerhalb des Zeilenzwischenraums verfolgt, geschieht die Verschiebung anhand eines Offsets in meinem Ansatz ausschließlich außerhalb der Textansicht, wenn Verbindungen zwischen relationierten Annotationen in unterschiedlichen Ansichten über die Seiten der Ansichten geführt werden. Denkbar wäre es, ähnlich BRAT solche Verschiebungen auch in den Zeilenzwischenräumen zu verwenden, um die Überlagerung der Verbindungen dort zu vermeiden. Jedoch ist der verfügbare Raum hier begrenzt, sodass der Raum für die effektive Verwendung der Verschiebungen erweitert werden müsste.

## 4 Konzepte

Nachfolgend soll das Konzept hinter den zu wählenden Ansätzen beschrieben werden. Hierzu werden die verschiedenen Anforderungen hinsichtlich Visualisierung und Interaktivität an das System definiert und anhand dieser anschließend der Entwurf sowie die Umsetzung des Systems abgeleitet.

### 4.1 Anforderungen

Die Anforderungen zur Darstellung relationierter Annotationen lassen sich anhand folgender Kriterien erheben

#### 4.1.1 Darstellung des Textes

Um dem Nutzer Aufschluss über die Entfernung zwischen Texten beziehungsweise relationierten Annotationen zu geben, soll von der klassischen Darstellung des Textes in einer Sequenz abgewichen werden. Stattdessen soll der gewählte Ansatz die Betrachtung und Annotation des Textes in mindestens zwei oder mehr visuell gesonderten Ansichten ermöglichen. Die Ansichten sollen dabei in Höhe und Breite so gewählt sein, dass sie den verfügbaren Raum optimal ausnutzen und den Nutzer dadurch in der Betrachtung und Bearbeitung größerer Textsegmente nicht beschränken.

#### 4.1.2 Visualisierung von Distanzen

Nach der Trennung des Textes soll der Raum um und zwischen den gesonderten Ansichten genutzt werden, um sowohl die Entfernung im Text, das heißt die Distanz als Textmenge zwischen Textbereichen, als auch zwischen relationierten Annotationen geeignet zu visualisieren. Insbesondere soll die Distanz intuitiv verständlich und erkennbar in der Visualisierung kodiert sein. Die Visualisierung selbst soll dabei die Bedeutung der Zu- oder Abnahme der Distanz geeignet wiedergeben und sich weiterhin dynamisch an die Interaktion mit dem Nutzer anpassen.

#### 4.1.3 Erstellung von Annotationen und Relationen

Innerhalb der Textansichten soll der Nutzer in der Lage sein, den Text mit eigenen Annotationen auszuzeichnen und diese miteinander verlinken zu können, sowohl innerhalb einer als auch zwischen mehreren Ansichten. Sowohl Annotationen, als auch Relationen sollen dabei typunabhängig miteinander kombinierbar sein. Die Repräsentation der Annotationen und Relationen soll so gewählt

sein, dass diese klar voneinander zu unterscheiden sind. Die Auszeichnung von Annotationen und Relationen soll dabei als interaktiver und intuitiv auszuführender Prozess realisiert werden, der die Kombination unterschiedlicher Typen ermöglicht.

### **4.1.4 Visualisierung überlagender Relationen**

Bei der Darstellung der Relationen von korrelierten Annotationen soll der gewählte Ansatz eine geeignete Lösung finden für den Kompromiss bei der Wahl der Visualisierung der Relation bezüglich dem dargestelltem Informationsgehalt, der Skalierbarkeit und der Übersichtlichkeit. Bedeutend ist dabei die konkrete Darstellung der Relationen im Text und der passende Umgang mit Überlagerung von Relationen, die bei punktueller Anhäufung dieser unweigerlich entsteht. Insbesondere soll die Lösung dabei etwaige Differenzen in der Visualisierung von Annotationen und Relationen in sowohl einer, als auch mehreren Textansichten beachten.

### **4.1.5 Dynamische Visualisierung von Relationen und Interaktivität**

Verändert sich bei der Interaktion durch den Nutzer die Positionen der korrelierten Annotationen einer Relation, so soll sich die Visualisierung dieser dynamisch sowohl an die Positionsänderung, als auch an die daran gekoppelte Zu- oder Abnahme der Entfernung zwischen den Annotationen anpassen. Weiterhin soll dabei ein geringer Informationsverlust durch die sich ebenfalls bewegenden Text und Überlagerungen angestrebt werden. In der Summe soll ein geeignetes Maß gefunden werden für die Möglichkeit und die Notwendigkeit von Interaktion zur Informationsgewinnung. Der Zugang zu den Informationen soll dabei flexibel sein und die Relevanz und die Priorisierung dieser widerspiegeln.

### **4.1.6 Benutzeroberfläche**

Für die graphische Benutzeroberfläche des Systems soll eine Anordnung ihrer Bestandteile gefunden werden, welche die Kernfunktionalitäten des Systems visuell kapselt, gleichzeitig aber das Zusammenspiel dieser unterstützt. Dem Nutzer soll anhand der Kapselung intuitiv und verständlich dargestellt werden, welche der Bereiche der Oberfläche welche Funktionalität integrieren. Werden Bereiche im Rahmen der Interaktion des Nutzers mit dem System visuell verändert, sollen die Änderungen der einzelnen Bereiche adäquat und konfliktfrei repräsentiert werden.

## **4.2 Entwurf**

Im Folgenden sollen auf Basis der zuvor erörterten Anforderungen die Entwurfsideen hinter den einzelnen Komponenten des Systems erläutert und aus den vielversprechendsten Ideen die Umsetzung der jeweiligen Komponente und ihrer Eigenschaften abgeleitet werden. Als Einstieg in das Kapitel wird der Begriff der Positionierung im Kontext des entwickelten Ansatzes beschrieben. Es folgt dann der Beginn des konzeptionellen Entwurfs mit der Erstellung einer möglichen Übersicht über die Benutzeroberfläche und ihrem initialen Zustand. Anschließend wird die Bedeutung des Trennfeldes für den Split der Textansichten und die Distanzdarstellung ergründet, ehe die Sinnhaftigkeit mehrer

Trennfelder diskutiert wird. In der Folge wird das Verständnis von Annotationen und Relationen aus der Entwurfssicht erläutert und auf Basis dessen die Umsetzung der Erstellung und der Visualisierung von Annotationen und Relationen, sowie Lösungsansätze für die Probleme, die dabei auftreten, abgeleitet. Zuletzt wird das kombinierte Verhalten der Komponenten beleuchtet und auf den möglichen Verlust von Informationen hierbei, sowie mögliche Lösungen eingegangen.

### 4.2.1 Positionierung von Textsegmenten

In diesem kurzen Unterkapitel soll das Verständnis des Begriffs der Position für den Verlauf der Arbeit erläutert werden. Im Rahmen des Vergleichs der verschiedenen Konzepte wird oft die Position annotierter Textsegmente zur Argumentation oder für Beispiele herangezogen. Generell lassen sich aber verschiedene Positionen eines Textsegments ausmachen:

- Die statische Position im Text, die sich, solange der Text selbst nicht verändert wird, ebenfalls nicht ändert.
- Die Position auf dem Bildschirm, auf dem der Text betrachtet und bearbeitet wird. Bezüglich dieser Position interessiert im Folgenden vor allem, ob ein Textsegment sichtbar oder nicht sichtbar ist. Anhand der Sichtbarkeit kann man die Position auf dem Bildschirm weiter spezifizieren:
  - Die absolute Position auf dem Bildschirm, das heißt ob sich ein Textsegment auf dem Bildschirm befindet oder nicht. Beispielsweise ist ein Textsegment in einem Textdokument, das geschlossen wurde, nicht sichtbar.
  - Die Position eines Textsegments relativ zu den anderen Objekten auf dem Bildschirm, das heißt ob das Segment durch ein anderes Objekt verdeckt wird oder nicht. Diese Position wird insbesondere für scrollbare Textansichten relevant. Der Text wird dabei nicht in seiner Vollständigkeit angezeigt, da entweder die Bildschirmgröße nicht ausreicht oder die Textansicht durch weitere Objekte auf dem Bildschirm beschränkt wird. Gerade der zuletzt beschriebene Fall tritt in dieser Arbeit wegen der Aufteilung des Textes oft auf. Wird der Text beispielsweise innerhalb seiner Ansicht gescrollt, dann verändert sich die Position des betroffenen Textsegment, indem sie aus dem Sichtbereich der Ansicht bewegt wird. Das Textsegment kann sich absolut betrachtet immer noch auf dem Bildschirm befinden, wird jedoch effektiv durch das angrenzende Objekt ober- oder unterhalb der Ansicht verdeckt.

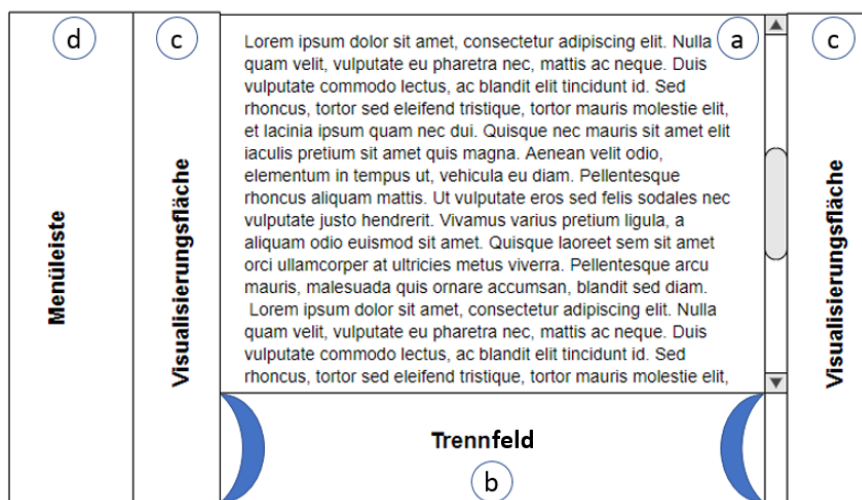
Von besonderem Interesse ist in dieser Arbeit die zuletzt genannte relative Positionierung eines Textsegments, die sich durch die Scroll Operationen des Nutzers oft verändert. Wird im Folgenden also die Position eines annotierten und gegebenenfalls relationierten Textsegments erwähnt, so ist die Rede stets von der relativen Position des Textsegments.

### 4.2.2 Initiale Ansicht und Oberfläche

Die in Abbildung 4.1 gezeigte graphische Oberfläche des Systems besteht im Wesentlichen aus 4 gesonderten Bereichen, die jeweils unterschiedliche Visualisierungs- und Interaktionsaufgaben übernehmen. In seiner Ausgangslage wird der zu bearbeitende Text strukturlos zunächst zusam-

menhangend in einer scrollbaren Textansicht presentiert, die sich zentral im Blickfeld des Nutzers befindet. Neben der Darstellung des Textes bietet die Ansicht die Moglichkeit, diesen im Rahmen der Erstellung von Annotationen und Relationen oder zur Dekomposition und Aufteilung des Textes auf zwei oder mehr separate Ansichten zu manipulieren. Sie fungiert hierbei als Nahansicht des Textes und seiner Annotationen und Relationen und bietet dadurch eine lokale Ubersicht uber solche Textabschnitte und ihre Daten, die sich zum Zeitpunkt der Betrachtung innerhalb des Sichtbereichs der Ansicht wiederfinden.

Fur die Aufteilung des Textes, die ab sofort als Split referenziert wird, wird das sogenannte Trennfeld verwendet. Das Trennfeld ist als Rechteck, mit Seitenlangen gleich den Textansichten, unterhalb der zentralen Ansicht realisiert. An den Innenseiten des Rechtecks finden sich bogenartige Strukturen, die ab sofort als Bander referenziert werden sollen. Die Funktionalitat des Trennfelds und seiner Bander sollen im Laufe des Entwurfs erlautert werden. Die freien Flachen links und rechts von der Textansicht und dem Trennfeld werden fur die Darstellung der Relationen zwischen verschiedenen Textansichten reserviert. Eine Menuleiste ganz links, welche Schnittstellen fur die Selektion von und Interaktion mit Annotationen, Relationen und den Bandern beinhaltet, rundet die Benutzeroberflache des Systems ab.



**Abbildung 4.1:** Der Entwurf der Benutzeroberflache: (a) Die initiale Textansicht, (b) Das Trennfeld, (c) Freie Flache zur Visualisierung der Relationen, (d) Menuleiste mit interaktiven Schnittstellen. Die Hohen- und Breitenmae der 4 Hauptbereiche sind adaptiv, aber fix gewahlt. Der zu bearbeitende Text befindet sich bereits in seiner Gesamtheit in der designierten Ansicht und lasst sich scrollen.

### 4.2.3 Funktion der Bänder

Die klassische, sequentielle Darstellung des zu bearbeitenden Textes erschwert den Umgang mit längeren Textsegmenten, die in ihrer Gesamtheit nur über das Scrollen des Textes betrachtet und annotiert werden können. Dazu kommen kurze oder lange Segmente, die in Relation gesetzt werden sollen. Solche relationierten Textsegmente können so nicht zur selben Zeit betrachtet werden, sodass ein Scrollen zwischen den jeweiligen Textabschnitten notwendig wird. Informationen über relevante Abschnitte, beispielsweise wie weit der Nutzer sich von diesem entfernt hat, gehen dabei verloren, sobald der betroffene Abschnitt aus dem Sichtbereich der Ansicht verschwindet.

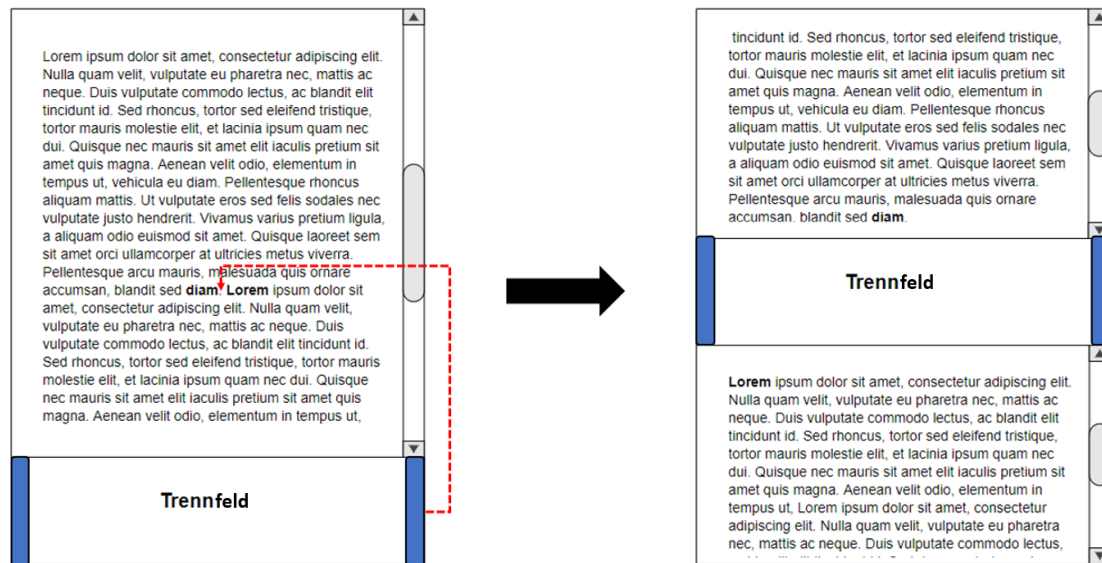
Zur Lösung dieser Problematik soll das System das Aufteilen des Textes an beliebig gewählten Stellen ermöglichen, sodass die Abschnitte in zwei oder mehr Ansichten zur selben Zeit bearbeitet werden können. Der Begriff Ansichten beschreibt dabei die Repräsentation desselben Textes abschnittsweise in mehreren, scrollbaren Textfenstern. Die Entfernung zwischen den aufgeteilten Textabschnitten soll dabei, sich dem Scroll Verhalten des Nutzers anpassend, visuell dargestellt werden. Als Werkzeug dazu integriert das System sogenannte Trennfelder.

#### Streckung bei sinkender Distanz

Wie vorangehend erwähnt, steht dem Nutzer mindestens ein Trennfeld mit sogenannten Bändern zur Verfügung, mittels dessen er den zugrunde liegenden Text trennen und auf zwei oder mehr scrollbare Ansichten, abhängig von der Anzahl verfügbarer Trennfelder, aufteilen kann. Um den Split zu initiieren, bewegt der Nutzer das Trennfeld per Drag-and-drop zur gewünschten Stelle im Text, an welcher dieser aufgeteilt werden soll. Das Trennfeld an die ausgewählte Position gebracht, an welcher der Split des vorliegenden Textes ausgeführt werden soll, bestimmt die Position des Mauszeigers das Wort im Text, das als Ausgangspunkt für den Split dient. Je nach gewählter Heuristik, geschieht der Split dann am Wort selbst, am Satzanfang ausgehend vom Wort oder, wie in Abbildung 4.2 gezeigt, am Satzende. Die Position des Splits markiert auch die neue Position des Trennfelds, während Ansicht wie Text geteilt werden, sodass die Ansicht zuvor durch zwei neue Textansichten mit den beiden Textteilen ersetzt wird. Die Gesamthöhe der verschiedenen Ansichten inklusive dem Trennfeld bleibt vor wie nach dem Split gleich. Abbildung 4.2 zeigt den Prozess.

Nach dem Split verbindet das Trennfeld die zwei neuen Textansichten in der Mitte, welche jeweils einen Teil des Textes umfassen. Jede dieser individuellen Splitansichten verhält sich wie die initiale Textansicht, bietet dieselben Optionen zur Manipulation des Textes und kann dementsprechend als Ziel für konsekutive Splits ausgewählt werden. Dabei gilt, dass für  $n$  Trennfelder stets  $n + 1$  Ansichten nach dem Split existieren: Der Split einer Splitansicht bedeutet die Bewegung des Trennfelds weg von seiner vorherigen Position. In der Folge wird das aktuelle Ziel des Splits geteilt und es existiert kurzzeitig zwei Splitansichten, die nicht durch ein Trennband geteilt werden. Diese werden dann zusammengeführt und der Inhalt der beiden Splitansichten innerhalb einer Ansicht dargestellt.

Scrollt der Nutzer innerhalb der beschriebenen Splitansichten, reagieren die Bänder der betroffenen Trennfelder visuell darauf. Initial befinden sich die Bänder dabei in einem gestrafften Zustand, in welchem sie in einer geraden Form von der oberen zur unteren Kante des Trennfelds spannen. Beide Bänder realisieren dieselbe Funktion, indem sie die Distanz zwischen den Textteilen der Ansichten repräsentieren, die durch das Trennfeld verbunden sind. Die Distanz bezeichnet dabei konkret den



**Abbildung 4.2:** Es wird der Split des Textes am Satzende geeizt. Die gestrichelte Linie dient als Indikator für die Mauszeigerposition beim Ablegen des Trennfelds.

für den Nutzer nicht sichtbaren Teil des betrachteten Textes, der zwischen den Sichtfenstern der Textansichten, genauer zwischen dem letzten sichtbaren Wort der oberen Ansicht und dem ersten sichtbaren Wort der unteren Ansicht, verborgen liegt.

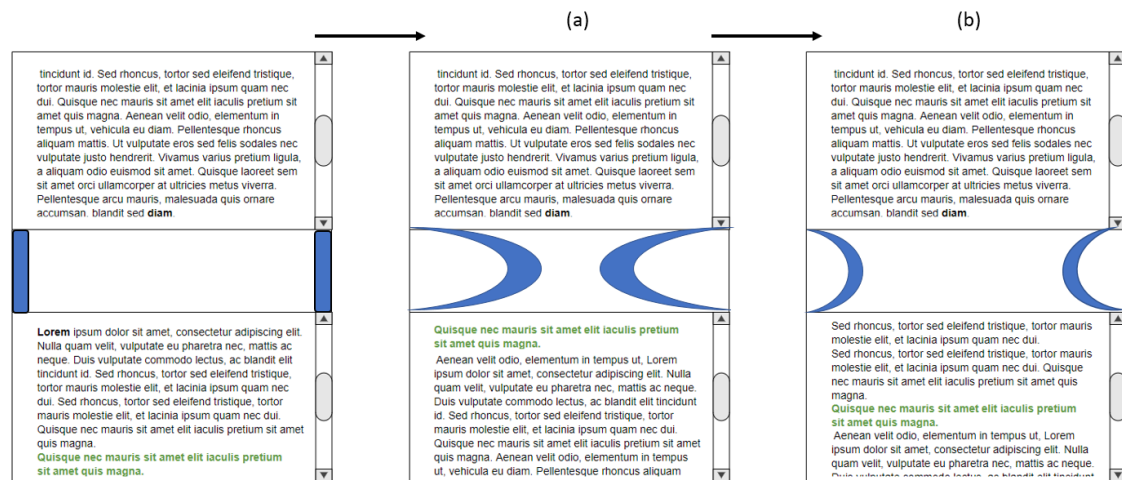
Wird nun innerhalb der oberen Ansicht nach oben gescrollt, um anschließend den Text in der unteren Ansicht nach unten zu scrollen oder unverändert zu lassen, so verschiebt sich der sichtbare Bereich einer oder beider Textteile. Zugleich wächst die Menge an verborgenem Text, da sich die beiden Sichtbereiche voneinander wegbewegen und damit die Distanz zwischen den Texten der betroffenen Ansichten zunimmt. Analog wird der Effekt durch das nach unten Scrollen in der unteren Ansicht und dem anschließenden nach oben Scrollen oder nicht Scrollen in der oberen Ansicht erreicht.

Die steigende Distanz wird dem Nutzer durch das Abrunden der beiden Bänder wie in Abbildung 4.3a visualisiert. Je größer also die Distanz wird, desto größer die Abrundung der Bänder, ehe ein Scrollen in einer oder beiden Ansichten nicht mehr möglich ist. Der zuletzt genannte Fall wird dem Nutzer signalisiert dadurch, dass sich die Bänder in der Mitte des Feldes an den Scheitelpunkten berühren.

Die Visualisierung von sinkender Distanz beim Annähern der beiden Textteile durch entsprechendes Scrollen in den jeweiligen Ansichten verhält sich symmetrisch, indem die Bänder gestrafft werden, je kleiner die Distanz wird (Abbildung 4.3b). Wurde die obere Ansicht zum letzten Wort der Textes und die untere Ansicht zum ersten Wort des Textes gescrollt, wodurch zwischen den beiden Schnittstellen kein Text mehr verborgen liegt, gleichen die Bänder ihrem initialen Zustand und spannen erneut von der oberen zur unteren Kante ihres Feldes.

Die Trennung der Texte und die Visualisierung der Distanz zwischen diesen erlaubt die effektive Annotation langer Textsegmente. Mithilfe der Splitansichten kann der Nutzer mehrere relevante Textsegmente zur selben Zeit betrachten und bearbeiten. Textsegmente die zu lang sind, um sie innerhalb einer Ansicht darzustellen, können mittels Split auf zwei Splitansichten verteilt





**Abbildung 4.3:** (a) Durch das Scrollen der unteren Ansicht nach unten, wird der Abschnitt vor dem grün markiertem Textsegment aus dem Sichtbereich der Ansicht nach oben verschoben. Die Wortmenge dieses Abschnitts beschreibt die Distanz zwischen dem letzten sichtbaren Wort der oberen und dem ersten sichtbaren Wort der letzten Ansicht nach der Scroll Operation. Die Distanz wird durch die Abrundung der Bänder visualisiert.

(b) In der unteren Ansicht wurde nach oben gescrollt, sodass das grün markierte Segment nach unten verschoben wird und Teile des Textbereichs vor diesem wieder sichtbar werden. Die verborgene Textmenge zwischen dem letzten sichtbaren Wort der oberen Ansicht und dem nun neuen, ersten sichtbaren Wort der unteren Ansicht sinkt im Vergleich zu (a), genauso auch die Distanz. Die Bänder werden dadurch gestrafft.

werden. Dies eröffnet die Möglichkeit, innerhalb der oberen Ansicht zum Anfang des relevanten Textabschnitts zu Scrollen und dort den Beginn der Annotation zu markieren. Anschließend lässt sich das Ende dieser innerhalb der unteren Ansicht am Ende des relevanten Abschnitts annotieren. Ein umständliches Scrollen über die gesamte Länge des relevanten Abschnittes bleibt dem Nutzer so erspart.

Darüber hinaus ermöglichen es die Bänder, bei der Bearbeitung eines Textes in mehreren Ansichten die Übersicht über die Position bearbeiteter Segmente bei Scroll Operationen zu behalten. Werden zum Beispiel korrelierte Textsegmente über mehrere Ansichten hinweg annotiert, verrät die in der Formveränderung der Bänder kodierte Distanz, wo sich diese Segmente relativ zueinander befinden und insbesondere, wie viel Text zwischen den Segmenten liegt. Die Information kann gerade dann wertvoll werden, wenn Relationen zwischen solchen Textsegmenten ausgezeichnet werden und die Distanz respektive Menge an Text zwischen den Segmenten Aufschluss über Eigenschaften der Relation gibt.

### **Streckung bei steigender Distanz**

Als alternative Visualisierung der steigenden und sinkenden Distanz zwischen den Textteilen bietet es sich an, das Verhalten der Bänder umzudrehen. Somit würden die Bänder bei steigender Distanz gestrafft und bei sinkender Distanz abgerundet werden.

Im direkten Vergleich stellen die beiden Ansätze die Information bezüglich der Entfernung unterschiedlich dar. Die Streckung bei sinkender Distanz bildet die Steigung der Distanz durch die Abrundung der Bänder auf die Abnahme der Fläche zwischen den Bändern ab. Insbesondere übersetzt dieser Ansatz die Zunahme der Menge der nicht sichtbaren Wörter in die Zunahme der Länge der Bänder. Der Ansatz der Streckung bei steigender Distanz präsentiert dagegen die Steigung der Distanz, indem die Fläche zwischen den Bändern, welche gestrafft werden, zunimmt. Hier wird die Zunahme der nicht sichtbaren Textmenge über die Abnahme der Länge der Bänder vermittelt.

Beide Ansätze decken damit jeweils unterschiedliche Anforderungen an die Distanzvisualisierung ab. Der Ansatz der Streckung bei sinkender Distanz erfährt seinen vollen Nutzen, wenn die Textmenge, die sich im Laufe einer Scroll Operation zwischen Start- und finalem Zustand der Textansicht nach unten oder oben bewegt, akkurat durch die Visualisierung dargestellt werden soll. Zwischen zwei Momentaufnahmen gibt dieser Ansatz einen besseren Aufschluss über den Zuwachs an Text beziehungsweise der Distanz. Interessiert sich der Nutzer dagegen mehr für die zurückgelegte Distanz ausgehend von einem fixen Startpunkt, so beschreibt der Ansatz der Streckung bei steigender Distanz die inkrementelle Distanzierung von diesem Punkt besser.

Beide Ansätze realisieren darüber hinaus unterschiedliche visuelle Metaphern, welche zur Diskussion und Bewertung verglichen werden können. Die angesrtebte Metapher hinter der Streckung bei sinkender Distanz ist die Repräsentation eines Gummibandes durch das Trennfeld. Die weiße Fläche zwischen den blauen Bögen soll dabei die Fläche des Gummibandes darstellen, die blauen Bögen selbst die Ränder des Bandes. Durch die Abrundung der blauen Bögen bei steigender Distanz soll so die Straffung des Gummibands und das daraus resultierende schmalere werden des Bandes visualisiert werden. Eine alternative Interpretation boten die Experten im Rahmen der Expertenbefragung, die im Ergeniskapitel besprochen werden soll, an. Unabhängig voneinander, kamen beide Experten zum Entschluss, dass das Trennfeld ein Blatt Papier repräsentiert und die Streckung bei sinkender Distanz das Zusammenschieben des Papiers und die daraus resultierende Wölbung darstellt.

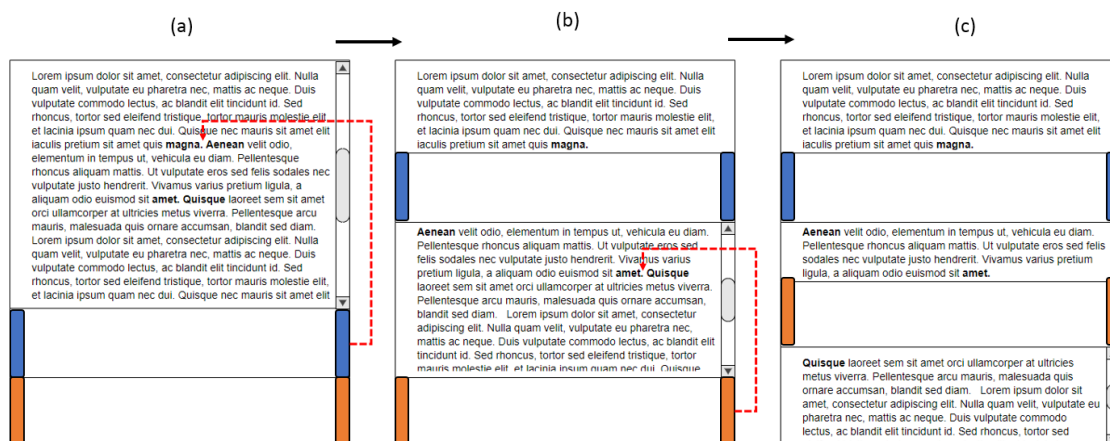
Der Ansatz der Streckung bei steigender Distanz steht hier im Kontrast, da das Trennfeld und die beiden Bänder jeweils andere Aufgaben in der Metapher übernehmen. In dieser Metapher stellen die beiden Bögen jeweils ein Gummiband dar, deren Enden an den jeweiligen Textansichten gebunden sind. Entfernen sich nun die sichtbaren Textabschnitte voneinander, was zur Zunahme der Distanz führt, so werden die Gummibänder mitgezogen und gestrafft. Andersherum erschlaffen die Bänder, wenn die Distanz sinkt, die Texte sich annähern und dadurch die Spannung, die auf den Bändern lastet, abnimmt.

Um die Interpretation frei zu gestalten, werden beide Ansätze in Kombination in das System integriert. Mittels eines aktivierbaren Buttons in der Menüleiste der Benutzeroberfläche, soll dem Nutzer dabei die Option gegeben werden, bezüglich der Ausrichtung und dem Verhalten der Bänder auf Wunsch zwischen den beschriebenen Ansätzen wechseln zu können. Insbesondere kann der Wechsel auch nach Abschluss einer oder mehrerer Scroll Operation und vor Beginn der Nächsten ausgeführt werden, um dem Nutzer so situationsabhängige Anpassungen zu ermöglichen.

## Mehrere Trennfelder

Das Trennfeld realisiert damit die Distanzvisualisierung auf Basis einer gewählten Funktion zwischen zwei Textansichten respektive ihren Textabschnitten. Daneben dient es als das Werkzeug für den Nutzer, um den Split der Textansichten auszuführen. Sollen nun mehr als zwei Textansichten zur selben Zeit bearbeitet werden, liegt es nahe, entsprechend die Anzahl an Trennfeldern, die dem Nutzer zur Verfügung gestellt werden, zu erhöhen.

Ursprünglich sah der initiale Entwurf der Benutzeroberfläche dabei zwei Trennfelder vor, die sich unter der initialen Ansicht befinden und unabhängig voneinander konsekutiv in den Text gezogenen werden, um den zu bearbeitenden Text an unterschiedlichen Position aufzutrennen. Abbildung 4.4 zeigt das Aussehen und Verhalten des Systems beziehungsweise seiner Oberfläche in diesem Fall: Zunächst wird die initiale Ansicht in 4.4a mit dem ersten Trennfeld geteilt, sodass zwei neue Splitansichten in 4.4.b entstehen. Wie diese Abbildung verdeutlicht, kann der komplette Textabschnitt innerhalb des oberen Sichtfensters dargestellt werden, da diese Splitansicht nur wenig Text enthält. Die obere Splitansicht kommt somit ohne Scrollbalken auskommt. Anschließend wird das zweite Trennfeld an eine ausgewählte Stelle im Text der unteren Splitansicht gezogen, wo es den zweiten Split initiiert. Abbildung 4.4c zeigt, wie die untere Splitansicht ersetzt wird durch zwei neue Splitansichten: Eine Ansicht ersetzt sie positionsgetreu und erhält den ersten Teil des geteilten Textes, die andere Ansicht wird unter dem zweiten Trennfeld positioniert und erhält den zweiten Teil. Ähnlich der oberen Ansicht in 4.4a, kann auf den Scrollbalken für die obere Ansicht hier und die mittlere Ansicht verzichtet werden. Die Gesamthöhe der Ansichten und der Trennfelder bleibt darüber hinaus über beide Splits hinweg gleich.



**Abbildung 4.4:** Die Benutzeroberfläche wird ergänzt um ein weiteres Trennfeld unterhalb der initialen Textansicht, deren Höhe dadurch verringert wird.

- Das erste Trennfeld wird an eine ausgewählte Stelle im Text der Initialansicht gezogen und losgelassen, wo es den ersten Split initiiert.
- Die Initialansicht wird ersetzt durch zwei neue Splitansicht, das erste Trennfeld verbindet die beiden Ansichten.
- Durch den zweiten Split existieren nun drei Splitansichten, jede einzelne durch ein Trennfeld von den anderen Ansichten getrennt.

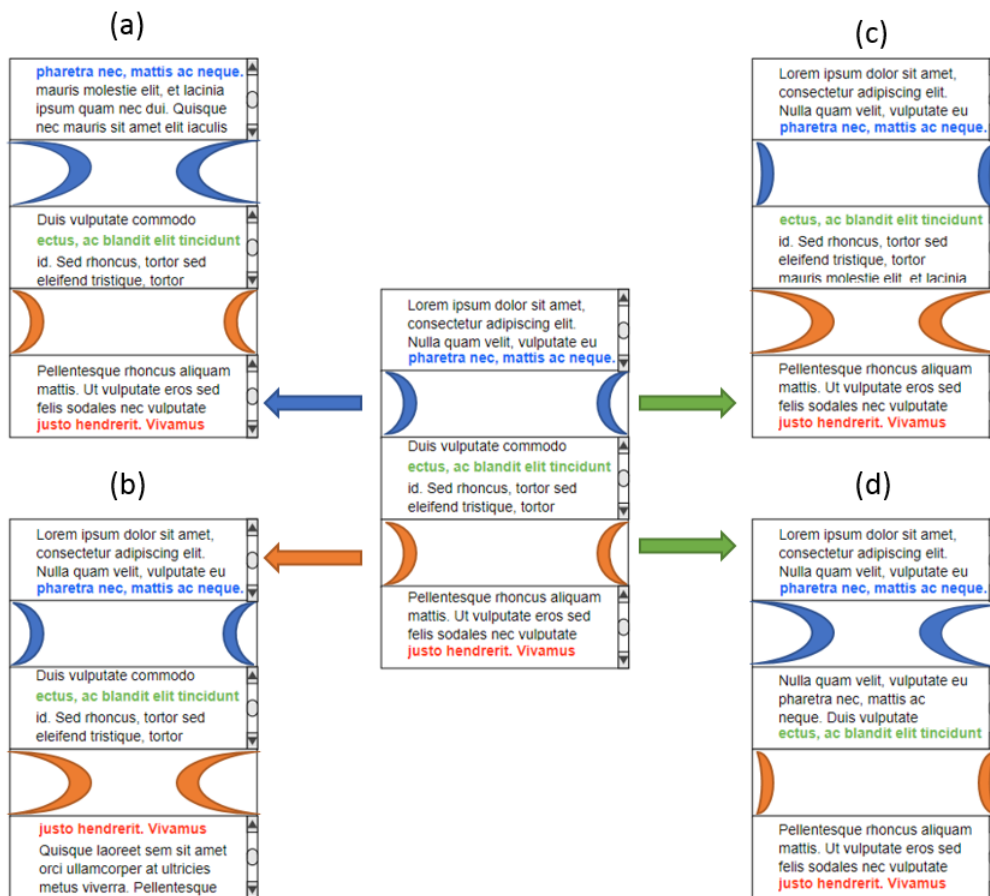
Die Trennfelder und ihre Bänder bieten jeweils dieselbe Funktionalität eines einzelnen Trennfeldes an, sodass sie auf Basis derselben Funktion die Distanz zwischen den zwei Ansichten visualisieren, welche sie miteinander verbinden.

Als Ergebnis existieren drei verschiedene Ansichten: Die obere Ansicht oberhalb eines Trennfelds, die untere Ansicht unterhalb des anderen Trennfelds, sowie die mittlere Ansicht zwischen den beiden Trennfeldern. Interaktion, das heißt Scrollen des Textes innerhalb der oberen oder der unteren Ansicht, bewirkt je nach Veränderung der Distanz die Abrundung oder Streckung der Bänder der beiden, unmittelbar benachbarten Trennfelder, ähnlich dem Verhalten eines einzelnen Trennfeldes. Scroll Bewegungen in der mittleren Ansicht führen dagegen zur simultanen Anpassung der Bänder beider Trennfelder. Die visuelle Reaktion der jeweiligen Bänder unterscheidet sich dabei abhängig davon, ob innerhalb der mittleren Ansicht in Richtung des betroffenen Trennfelds oder von diesem Weg gescrollt wird:

- Wird innerhalb der oberen Ansichten gescrollt, während die mittlere und untere Ansicht unverändert bleiben, so steigt die Distanz zwischen den Textabschnitten der oberen und der mittleren Ansicht in der Folge. Die Bänder des oberen Trennfelds werden dann abgerundet, wie Abbildung 4.5a zeigt.
- Analog verhält sich der Fall, wenn die untere Ansicht gescrollt wird, während die mittlere und die obere Ansicht unverändert bleiben. Abbildung 4.5b verdeutlicht dies.
- Wird nun in der mittleren Ansicht nach unten gescrollt, während die obere und die untere Ansicht unverändert bleiben, so sinkt die Distanz zwischen der mittleren Ansicht und einem der beiden anderen Ansichten. Gleichzeitig steigt die Distanz für die andere, verbleibende Ansicht. Abbildungen 4.5c und 4.5d zeigen dies anhand zweier Beispiele. In Abbildung 4.5c nähern sich die Texte der mittleren und oberen Ansicht an, wodurch die Bänder des Trennfelds zwischen diesen Ansichten gestrafft werden. Gleichzeitig steigt die Distanz zwischen der mittleren und der unteren Ansicht, sodass die Bänder des Trennfelds zwischen diesen Ansichten abgerundet werden.

Die Nutzung von zwei Trennfeldern ermöglicht dem Nutzer somit, den Text in bis zu drei Ansichten zur selben Zeit zu betrachten und zu bearbeiten. Die mittlere Ansicht dient dabei als Verbindungsstück zwischen den voneinander isolierten Ansichten oberhalb und unterhalb. Die mittlere Ansicht und die beiden Randansichten werden paarweise auf Basis der Visualisierung der Distanz durch die Bänder in Relation gesetzt, welche aber nicht transitiv ist: Wächst die Distanz zwischen den dargestellten Textabschnitten der mittleren und der unteren Ansicht, impliziert dies nicht den Wachstum für die obere und die untere Ansicht und vice versa. Der dargestellte Abschnitt einer Ansicht bleibt fix solange, bis der Nutzer in dieser Ansicht scrollt. Dies eröffnet die Möglichkeit, mehrere Textsegmente effizient zu bearbeiten, welche zueinander in Beziehung stehen. Bei  $n - 1$  Trennfeldern, lassen sich dadurch in den  $n$  resultierenden Splitansichten die Teilnehmer  $n$ -ärer Relationen jeweils einzeln betrachten und bearbeiten. Für zwei Trennfelder wären so Konstellationen möglich, in welchen ein als zentral erachtetes Textsegment in der mittleren Ansicht betrachtet wird, während der umgebende Text in der oberen- beziehungsweise unteren Ansicht bearbeitet wird, ohne dabei das zentrale Segment aus dem Blick zu verlieren.

Während die Einbindung mehrerer Trennfelder Vorteile bezüglich der Darstellung und Bearbeitung relationierter Textsegmente mit sich bringt, stellt sie gleichzeitig eine Herausforderung bezüglich der Visualisierung des gesamten Textes dar.



**Abbildung 4.5:** Das Verhalten von zwei Trennfeldern beim Scrollen in den Ansichten lässt sich anhand vier Hauptfällen erläutern. Initial wird angenommen, dass sowohl in der oberen, als auch in der unteren Ansicht ein wenig gescrollt wurde, sodass beide Bänder etwas abgerundet wurden.

(a) Das blau markierte Segment bewegt sich vom grün markierten Segment weg, sodass die Distanz steigt und die blauen Bänder abgerundet werden.

(b) Analog zum Fall a.

(c) Das grün markierte Textsegment verschiebt sich nach oben und nähert sich so dem blauen Segment an und bewegt sich vom roten Segment fort. Die blauen Bänder werden dadurch gestrafft, die orangenen Bänder werden abgerundet.

(d) Das grün markierte Textsegment verschiebt sich nach unten und nähert sich so dem roten Segment an und bewegt sich vom blauen Segment fort. Die orangenen Bänder werden dadurch gestrafft, die blauen Bänder werden abgerundet.

Um die Relation zwischen der sich verändernden Distanz und Formveränderung der Bänder ersichtlich und nachvollziehbar wiederzugeben, muss jedem Trennfeld eine fixe Minimalhöhe abhängig von der verfügbaren Höhe für die Benutzeroberfläche zugestanden werden. Zur Folge hat dies, dass mit jedem zusätzlichen Trennfeld die resultierenden Splitansichten in ihrer Höhe schrumpfen, sodass diese in der Darstellung größerer Textsegmente limitiert werden. Insbesondere stellt sich die Frage, ob die Visualisierung der Distanz zwischen Textabschnitten an mehr als einer Stelle im Text notwendig ist, wenn die Abschnitte selbst nicht lang genug sind, um die für die Bearbeitung relevanten Bereiche, die bei einem Trennfeld genügend Raum in einem Abschnitt finden würden, zu präsentieren. Zuletzt werden dem Nutzer dadurch mehr Scroll Operationen abverlangt, um zu lange Textsegmente betrachten und bearbeiten zu können. Dem Vorteil von mehr als zwei Splits inklusive Sets an Bändern, stehen so die Nachteile der Visualisierung hinsichtlich effektiver Darstellung des Textes direkt gegenüber.

In der Summe erfüllt der Ansatz mit einem Trennfeld bereits die Anforderung an das System, den zu bearbeitenden Text in mehreren, geeigneten Ansichten darzustellen. Die mit den Trennfeldern wachsende Anzahl an Splitansichten sowie Bänder haben ihren speziellen Nutzen, falls der zu bearbeitende Text sehr lang ist und die Aufteilung auf kleinere Ansichten die Bearbeitung vereinfacht oder auch, wenn vermehrt mehrstellige Relationen ausgezeichnet und bearbeitet werden sollen. Abseits der beschriebenen Anwendungsfälle, bietet ein einzelnes Trennfeld bereits die notwendigen Funktionen zur Annotation längerer oder entfernter Textsegmente und Visualisierung binärer Relationen, wie sie im Fokus dieser Arbeit stehen, an. Schließlich fiel somit die Entscheidung zugunsten eines einzelnen Trennfeldes.

### Distanzkodierung

Die zu visualisierende Distanz selbst soll kodiert werden als die Anzahl an nicht sichtbaren Wörtern, die sich im Scroll Bereich unterhalb des sichtbaren Textabschnitts in der oberen Ansicht und im Scroll Bereich oberhalb des sichtbaren Textbereichs der unteren Ansicht befindet. Die nichtnegative Summe wird bei jeder Scroll Operation des Nutzers neu berechnet, um die Konsistenz der Visualisierung zu wahren. In beiden Ansätzen wird die Summe abgebildet auf die Krümmung der Bänder, welche diese bei der Abrundung erfahren. Die abbildende Funktion kann dabei nach Bedarf unterschiedlich gewählt werden. Im Folgenden wird sie für den ersten Ansatz als Quotient der Summe der Wörter  $S$  und der verfügbaren halben Breite  $B$  des Trennfelds in Pixeln realisiert, ihre Ausgabe ist die nichtnegative Krümmung  $K$  eines Bandes in Pixeln, wobei  $K$  initial 0 ist und sich im Intervall  $[0, \dots, B]$  bewegt. Der Faktor  $c$  wird zur Normalisierung verwendet.

$$K := K + c \cdot \frac{S}{B} \quad (4.1)$$

Im zweiten Ansatz wächst die Krümmung antiproportional zur Summe der verborgenen Wörter. Mit  $K = B$  wird die Krümmung dazu initial als halbe Breite des Trennfelds gewählt und sinkt anschließend, je größer die Distanz wird. Der Faktor  $d$  ist eine Konstante zur Normalisierung.

$$K := K - d \cdot \frac{S}{B} \quad (4.2)$$

#### 4.2.4 Repräsentation und Erstellung von Annotationen und Relationen

Im Folgenden soll eine geeignete Repräsentation der Annotationen und Relationen im Rahmen des Ansatzes gefunden werden, auf derer basierend die interaktive Erstellung und Visualisierung von Annotationen umgesetzt werden kann. Die konkrete Umsetzung soll dabei den Entstehungsprozess möglichst frei gestalten, sodass der Nutzer Freiheiten bei der Kombination verschiedener Annotations- und Relationentypen genießt. Der Fokus liegt im folgenden auf der Repräsentation und der Darstellung von Annotationen, so dass der Inhalt der Annotationen beschränkt werden soll.

Eine Annotation ist im Wesentlichen ein Textsegment variabler Länge, welches durch den Nutzer durch das Anklicken oder durch das Ziehen des Mauszeigers selektiert und in der Folge entsprechend im Text gekennzeichnet wird. Jede Annotation stellt eine in sich geschlossene Entität dar, die eine feste Menge an Attributen mit sich bringt:

- Einen Bezeichner zur eindeutigen Identifikation der Annotation
- Die assoziierte Textmenge, welche die Annotation auf Textebene repräsentiert. Als tokenbasierter Ansatz werden hiermit konkret die durch den Nutzer selektierten Token des Textes, das heißt alle möglichen Symbole abseits der Leerzeichen, referenziert.
- Eine zugeordnete Klasse, die Aussagen über den Typ der Annotation tätigt. Anhand der Klasse lassen sich Annotation desselben Typs gruppieren. Will der Nutzer beispielsweise jedes Vorkommen eines Informatikers in einem wissenschaftlichen Dokument annotieren, so kann er jedes dieser Vorkommen mit einer Annotation der Klasse "Informatiker" auszeichnen.
- Die Bezeichner der Klassen werden eindeutig gewählt, um Duplikate zu vermeiden.
- Dasselbe gilt auch für ihr Farbattribut, welches mit der zugeordneten Klasse korrespondiert und diese visuell von anderen Klassen abgrenzt
- Die Bezeichner einer potenziell leeren Menge an relationierten Annotation, zu welchen die Annotationen in Beziehung steht.

Durch Relationen werden bereits existierende Annotationen in Beziehung gesetzt, auf visueller Ebene repräsentieren sie dabei die ungerichtete Verbindung solcher relationierter Annotationen. Eine Relation, beispielsweise eine binäre Relation, wurde in einer ersten Design Entscheidung zunächst automatisiert erzeugt, wenn mindestens eine Relationenklasse existierte und sobald zwei Texttoken oder Segmente konsekutiv annotiert wurden. Diese wurden dann durch eine Relation der aktuell aktiven Relationenklasse in Beziehung gesetzt. Dieser Ansatz der interaktiven Auszeichnung von Relationen warf aber Probleme auf, da er die Erstellung von Annotationen und Relationen vermischt.

Zur Besserung dieses Ansatzes sollen nun verschiedene Modi eingeführt werden: Durch das Drücken eines oder mehrerer Buttons wechselt der Nutzer zwischen einem Annotations- und einem Relationenmodus, in welchem er jeweils ausschließlich Annotationen oder Relationen erstellen kann. Dies erlaubt die "Vorproduktion" von Annotationen, die dann durch Relationen verbunden werden können. Eine binäre Relation erstellt er dabei konkret, indem er nun zwei bereits existierende Annotationen nacheinander anklickt, sodass diese dann in Beziehung gesetzt werden.

Auch Relationen behandelt das System als getrennte Entitäten, welche jeweils einer Klasse angehörig sind, ähnlich den Annotationen. Die Klassen der Relationen sind eindeutig gekennzeichnet durch jeweils eindeutig gewählte Bezeichner und Farbattribute. Liegt so beispielsweise die Aufgabe vor, unter den zuvor annotierten Informatikern solche in Relation zu setzen, die gemeinsam an einer Publikation gearbeitet haben, kann der Nutzer eine Relation der Klasse "haben publiziert" erstellen und mittels dieser die betroffenen Vorkommen von Informatikern in Beziehung setzen.

Bereits angedeutet wurde, dass die Erzeugung von sowohl Annotationen, als auch Relationen nach dem Prinzip des *slot-fillings* abläuft, das heißt der Nutzer erstellt zunächst die Klasse als Repräsentation der gemeinsamen Attribute, in diesem Fall derselbe Tag und dasselbe Farbattribut und anschließend die konkreten Instanzen. Konkret werden somit zuerst die "blanken" Annotations- oder Relationenklassen erstellt und nach Selektion der ausgewählten Texttoken wie oben beschrieben, wird eine Instanz der betroffenen Klasse erzeugt. Der Prozess kann durch den Nutzer für jede Annotations- beziehungsweise Relationenklasse beliebig wiederholt werden, das heißt beliebig viele Instanzen einer Klasse erzeugt werden. Insbesondere gibt es keine Einschränkung bezüglich der Kompatibilität von Annotations- und Relationenklassen, sodass jede Annotation einer beliebigen Klasse mit einer anderen Annotation derselben oder sich unterscheidenden Klasse im Rahmen einer Relation einer beliebig gewählten Klasse in Beziehung stehen kann.

Bei der Auszeichnung von Relationen sind neben binären Relationen, also einer Relation mit genau zwei Teilnehmern, auch n-äre Relationen mit beliebiger Anzahl an teilnehmenden relationierten Annotationen zu betrachten. N-äre Relationen werden im Ansatz als eine Kette von binären Relationen realisiert. Will der Nutzer eine Folge von Annotationen  $a_1, a_2, \dots, a_n$  im Rahmen einer Relation in Beziehung setzen, so erreicht er dies, indem er dieselbe Relation paarweise für  $a_1$  und  $a_2$ ,  $a_2$  und  $a_3 \dots a_{n-1}$  und  $a_n$  auszeichnet. In der Darstellung der einzelnen Verbindungen unterscheiden sich diese nicht von binären Relationen, sodass der im Folgenden erläuterte Highlight-Mechanismus beispielsweise nur ein Paar statt der gesamten Kette hervorhebt. Der Ansatz behandelt die n-äre Relation somit als separate, binäre Relationen.

N-äre Relationen ließen sich expliziter darstellen und bietet außerdem im Vergleich zu binären Relationen weiteren Spielraum bezüglich der Visualisierung an, beispielsweise hinsichtlich sich unterscheidender Klassen von Relationen innerhalb derselben Kette oder der Auflösung überlappender Relationen, das heißt die Auszeichnung von sich unterscheidenden Relationen für dieselbe Annotationsmenge.

Da die Besprechung möglicher Ansätze zur Lösung dieser Probleme den Rahmen dieser Arbeit sprengen würde, wird der Fokus fortgehend auf die Auszeichnung und Visualisierung von Annotationen und den binären Relationen zwischen diesen gesetzt. N-äre Relationen werden weiterhin implizit erzeugt und die im Folgenden besprochenen Ansätze werden für die binären Relationen angewandt. Für das System bedeutet dies, dass das Feld der Bezeichner der relationierten Annotationen für eine Annotation entweder leer ist oder genau ein Element enthält.

### 4.2.5 Visualisierung der Annotationen und Relationen

Sowohl Annotationen, als auch Relationen bringen verschiedene Schwierigkeiten bei der Visualisierung mit sich. Während vor allem die Annotation langer Abschnitte einer speziellen visuellen Lösung bedürfen, falls diese innerhalb einer einzelnen Ansicht nicht zu überblicken sind, ist die Visualisierung von Relationen herausfordernd hinsichtlich ihrer allgemeinen Darstellung, die auch



das Verhalten bei Überschneidungen und Überlagerungen bestimmt. Die Darstellung muss sich dabei den Einschränkungen anpassen, die durch die Quantität ausgezeichneter Relationen, die Interaktion des Nutzers und der umgebenden Bestandteile der Benutzeroberfläche auferlegt werden.

Eine geeignete Darstellung soll gefunden werden, indem Annotationen und Relationen eindeutig und voneinander unterscheidbar visualisiert werden. Für die Darstellung der Annotationen wird eine simple Hervorhebung im Text gewählt, während Relationen abhängig von der Lokation der entsprechenden, relationierten Annotationen gezeichnet werden. Eingeführt wird die Unterscheidung zwischen lokalen Relationen, das sind Relationen zwischen Annotationen, die sich innerhalb derselben Textansicht befinden und relationierten Annotationen, die jeweils über verschiedene Textansichten verteilt sind und als entfernte Relationen bezeichnet werden. Bewusst werden dabei unterschiedliche Ansätze zur Visualisierung der zwei Varianten von Relationen gewählt, um diese visuell unterscheiden zu können. Dies garantiert eine bessere Übersicht im Falle sich überschneidender oder überlagernder Relationen.

### **Grundlegende Visualisierung**

Die Visualisierung der Annotationen und der Relationen erfolgt anhand ihrer Klassifikation, welche zur Gruppierung dieser dient. Die Gruppierung geschieht dabei durch die gewählten Tags und Farbattribute, welche als eindeutige, gemeinsame Merkmale jeder Annotation beziehungsweise Relation der zugehörigen Klasse funktionieren.

Nach der Selektion des designierten Textsegments durch den Nutzer, wird die Annotation farblich anhand des Farbattributs der zugehörigen Klasse hervorgehoben. Zur Hervorhebung von Annotationen und Relationen soll der Highlight-Mechanismus genutzt werden. Dazu wird die Transparenz der hervorgehobenen Annotationen initial erhöht. Hebt der Nutzer dann eine Relation und ihre Annotationen per Highlight hervor, wird die Opazität erhöht und die Helligkeit für die Dauer der Interaktion stark verstärkt. Einen Sonderfall bilden Token, die Teil mehrer Annotationen unterschiedlicher Klassen sind. Diese werden grau hervorgehoben, um den Nutzer die Unterscheidung der Klassen zu signalisieren. Das Auszeichnen von Annotationen verschiedener Klassen geschieht, indem der Nutzer abwechselnd aktiviert, wobei zu jeder Zeit stets nur eine Klasse aktiv ist. Dazu werden aktivierbare Buttons für jede vorkommende Klasse innerhalb der Menüleiste der Benutzeroberfläche angeboten.

Die Auszeichnung und Visualisierung langer Annotationen wird ähnlich umgesetzt. Um die Selektion eines langes Textabschnitts zu vereinfachen, kann der Nutzer per Doppelklick Anfang und Ende einer solchen Annotation markieren. Das System erkennt den Abschnitt fortgehend als eine Annotation und hebt sie farblich hervor. Es bleibt jedoch das Problem bestehen, dass Annotationen, die länger sind als der Sichtbereich der Ansicht darstellen kann, nur schwer oder gar nicht überblickt werden können. Es bietet sich deswegen an, sehr lange Annotationen implizit in Form einer Relationen darzustellen. Die Idee hierbei ist, das Token zu Beginn des betroffenen Textabschnitts und das letzte Token dessen als separate Annotationen zu markieren, ehe diese mittels einer Relation mit passendem Tag in Beziehung gesetzt werden. Die Relation markiert damit Anfang und Ende der langen Annotation, während die Tokens dazwischen implizit Teil dieser werden, ohne visuell hervorgehoben zu werden. In Verbindung mit der Aufteilung des Textes auf die Splitansichten, lassen sich lange Annotationen so effektiv auszeichnen und visualisieren.

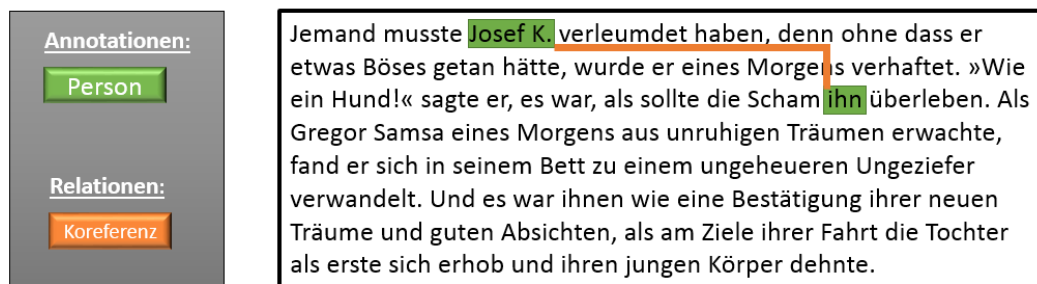
Wird nun eine Relation durch den Nutzer ausgezeichnet, werden die Teilnehmer der Relation visuell mittels einer ungerichteten Kante verbunden. Die Verbindung nimmt dabei die Farbe des Farbattributs der Klasse an, welcher die Relation angehört ist. Ähnlich den Annotationen bietet die Menüleiste auch für die Relationen Buttons zur Aktivierung der verschiedenen Relationenklassen.

### Relationen in verschiedenen Ansichten und Umgang mit Überlagerung

Die Visualisierung der Verbindung zwischen Annotationen als Repräsentation einer Relation variiert abhängig davon, ob die betroffenen Annotationen sich innerhalb derselben oder in verschiedenen Ansichten befinden.

Lokale Relationen werden direkt durch eine Kante in der Farbe der Klasse verbunden, wie Abbildung 4.6 verdeutlicht. Die Kante verläuft dabei innerhalb der Ansicht, welche die relationierten Annotationen enthält und vermeidet Informationsverlust durch Überschneidung des Textes zwischen den betroffenen Textsegmenten indem sie, sofern möglich, zwischen den Zeilen verläuft. Nicht zu Vermeiden ist bei diesem Ansatz, dass Teile der direkten Kanten über den Text verlaufen und diesen verdecken. Die Nutzung des Zeilenzwischenraums wirkt dem bereits zu Teilen entgegen, darüber hinaus profitiert die Darstellung, wenn heuristisch der kürzeste Weg zwischen den visuell zu verbindenden Annotationen bestimmt wird. Mit der Wahl der passenden Heuristik kann hier anschließend die Überschneidung des Textes weiter reduziert werden.

Einen alternativer Ansatz beschreibt die Bogendarstellung der Relationen, die im Rahmen der entfernten Relationen erneut aufgegriffen und erläutert wird. Da sie dort als Lösungsansatz für Überlagerung von Relationen verwendet wird, wird bei der Darstellung lokaler Relationen von diesem Ansatz abgesehen und der Ansatz der direkten Kanten gewählt.



**Abbildung 4.6:** Eine lokale Relation, dargestellt durch eine direkte Kante zwischen den Teilnehmern. Die Menüleiste zeigt die bisher ausgezeichneten Annotationen und Relationen inklusive ihrer Tags, die per Button aktiviert werden können.

Für die entfernten Relationen bietet es sich an, diese ähnlich den lokalen Relationen über eine direkte Verlinkung der Teilnehmer mittels einer Kante über den kürzest möglichen Weg zu visualisieren. Durch die direkte und unmittelbare Verbindung der annotierten Textsegmente kann der Nutzer die Relation und ihre Teilnehmer auf einen Blick nachverfolgen und ausfindig machen.

Zur selben Zeit tritt für beide Arten der Relation jedoch das Problem auf, dass eine solche, direkte Verbindung gegebenenfalls den darunter liegenden Text sowie im Falle der entfernten Relationen, das als Mittelstück zwischen den Ansichten fungierende Trennfeld verdeckt. Dieser Effekt verstärkt sich zudem potenziell mit jeder neuen Relation, deren Kante sich mit der vorherigen Relation kreuzt. Das Resultat sind sich überlagernde Relationen, die nicht nur den darunter liegenden Text verdecken, sondern auch schwerer nachzuverfolgen sind, sodass die Identifikation der annotierten Textsegmente, die durch die betroffenen Relationen verbunden sind, erschwert wird.

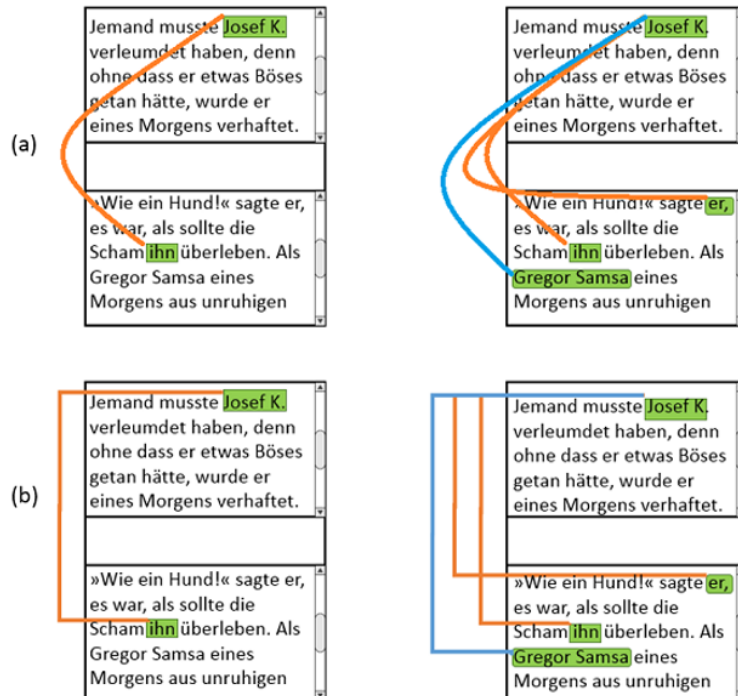
### Relationen als Bögen

Zur Vermeidung von Überlagerung bei der Visualisierung von Relationen bieten sich verschiedene Möglichkeiten an. Eine Lösung wird erreicht, indem die verbindenden Kanten der Relationen als Bögen durch den Zeilenzwischenraum über die freien Visualisierungsflächen an beiden Seiten der Ansicht geführt werden. Die Bögen können dabei als abgerundete Bögen oder als eckige Komposition zweier horizontal verlaufender und einer vertikal verlaufenden Kante realisiert werden. Abgerundete Bögen können je nach Präferenz weiter in den Raum gezogen werden, sodass Anfangs- und Endstück innerhalb der Ansichten zwischen den Zeilen geführt werden können. Müssen dabei mehrere Relationen dargestellt werden, werden die Bögen unterschiedlich weit in den Raum gezogen, um die direkte Überlagerung zu vermeiden. Dabei kann es aber, wie in Abbildung 4.7a zu sehen, zu Überschneidung kommen. Bei den eckigen Bögen werden die horizontal verlaufenden Kanten durch den Zeilenzwischenraum geführt, sodass der Text lesbar bleibt. Wie Abbildung 4.7 b zeigt, werden bei mehreren Relationen die horizontal verlaufenden Kanten konsekutiv gezeichneter Relationen verlängert, sodass sich die vertikal verlaufenden Kanten dieser Relationen nicht überlagern.

Beide Varianten vermeiden die Überschneidung des Trennfelds, die Überschneidung des Textes bleibt für abgerundete Bögen jedoch bestehen. Dafür legen die runden Bögen eine kürzere Strecke als die eckigen Bögen zurück, zudem werden Mehrdeutigkeiten durch die Überlagerung von Relationen zwischen den Zeilen vermieden, wie Abbildung 4.7a zeigt.

Beide Ansätze erlauben einen simplen Mechanismus zur Vermeidung von Überlagerung bei sich überlappenden oder überschneidenden Relationen. Hierbei wird jeder Bogen der Relationen unterschiedlich weit in den Visualisierungsraum links oder rechts vom Text geführt. Die Bögen erhalten somit ein horizontales Offset von wenigen Pixeln bezüglich dem Abstand zu den Textansichten. Für konsekutiv gezeichnete Relationen wächst das Offset, sodass keine Relation denselben Abstand zu den Ansichten besitzt.

Während dies eine Lösung für vertikale Überlagerungen bietet, treffen beide Ansätze weiterhin auf Konflikte bezüglich horizontaler Überlagerung. Befinden sich die durch die jeweilige Relation verbundenen Annotationen zweier Relationen in derselben Zeile des jeweiligen Textausschnitts, überlagern sich Anfangs- beziehungsweise Endstücke der Bögen. Gehören die jeweiligen Relationen unterschiedlichen Klassen an und sind damit unterschiedlich gefärbt, kann die Überlagerung der Relationen zur schlechteren Sichtbarkeit einer der beiden Relationen führen. Stammen beide Relationen von derselben Klasse ab, ist gegebenenfalls nicht zu unterscheiden, welche der Ausgangs- mit welchen der Ziel Annotationen verbunden sind. Durch die Überlagerung der Bögen in den Zeilenzwischenräumen der Textansichten wird somit die Verfolgbarkeit dieser stark vermindert.



**Abbildung 4.7:** Zwei Varianten der Bogendarstellung und ihr Verhalten, wenn mehrere, unterschiedliche Relationen ausgezeichnet werden.

- (a) Die Relation wird als abgerundeter Bogen dargestellt, der die Überschneidung des Trennfelds teilweise vermeidet.
- (b) Die Relation wird als eckiger Bogen dargestellt, der die Überschneidung des Textes gänzlich vermeidet.

Intuitiv ließe sich hier ein vertikales Offset für Relationen innerhalb derselben Textzeile einführen, doch ist der Zeilenzwischenraum begrenzt und das Zeichnen der Bögen über die Grenzen hinaus führt erneut zur Überdeckung des Textes. Effektiver lässt sich das Problem mittels des Highlight-Mechanismuses umgehen. Dazu wird der verbindende Bogen visuell hervorgehoben, wenn der Nutzer mit der Maus über die relationierten Annotationen fährt.

Beide Ansätze stoßen bei der Darstellung der Bögen an ihre Grenzen bezüglich der Skalierbarkeit: Da die Bögen in die Visualisierungsflächen wachsen, hängt die Anzahl an darstellbaren Relationen, welche Überlagerungen und die visuelle Überladung des Raumes gering hält, von der zuzuweisenden Breite dieser Räume ab. Werden absolute horizontale Offsets für die Bögen verwendet, besteht eine lineare Abhängigkeit zur Breite der Visualisierungsräume, wodurch diese ab einer festen Anzahl an zu zeichnenden Bögen überschritten werden. Bei einer Wahl der Offsets relativ zu den Bögen von

Nachbarrelationen wird bei steigender Anzahl an Relationen zwangsläufig der Punkt erreicht, an dem die Abstände zwischen den Bögen zu klein werden. In der Folge tritt erneut Überlagerung auf, sodass einzelne Bögen visuell schwer oder gar nicht voneinander zu unterscheiden sind.

Stammen die vielen Relationen zusätzlich von sich unterscheidenden Klassen ab, im schlechtesten Fall gehört jede Relation zu einer anderen Klasse und hat dadurch eine andere Farbe, mindert dies die Übersichtlichkeit der Visualisierung als Ganzes durch die vielen, unterschiedlich gefärbten Bögen, welche das Blickfeld des Nutzers überladen.

Die Bögen als Visualisierung der entfernten Relationen lösen das Problem der Überlagerung somit für eine kleine bis moderate Anzahl an ausgezeichneten Relationen derselben oder unterschiedlicher Klassen. Die Wahl zwischen gerundeten oder eckigen Bögen hängt von dem Kompromiss ab, welchen sich der Nutzer zugestehen will. Gerundete Bögen steigern die Verfolgbarkeit einzelner, visualisierter Relationen, indem sie sich stärker an die direkte, gradlinige Interpretation annähern und dadurch eine kürzere Strecke zurücklegen. Dies bietet sich insbesondere für die implizite Darstellung langer Annotationen als Relationen an.

Im Gegenzug überschneiden sie den umgebenden Text beim Aus- beziehungsweise Eintritt und vermeiden Überschneidungen mit anderen Relationen wesentlich ineffizienter als eckige Bögen, sodass die Wahrscheinlichkeit für stark verflochtene Konstrukte größer ist. Eckige Bögen schneiden durch die orthogonalen Überschneidungen mit anderen eckigen Bögen in diesem Punkt besser ab, büßen ihrerseits jedoch durch die expliziten Knicke im Übergang der Kanten in ihrer Verfolgbarkeit ein. Auch belegen eckige Bögen mehr Platz durch die größere Distanz, die sie bei der Verbindung der relationierter Annotationen zurücklegen.

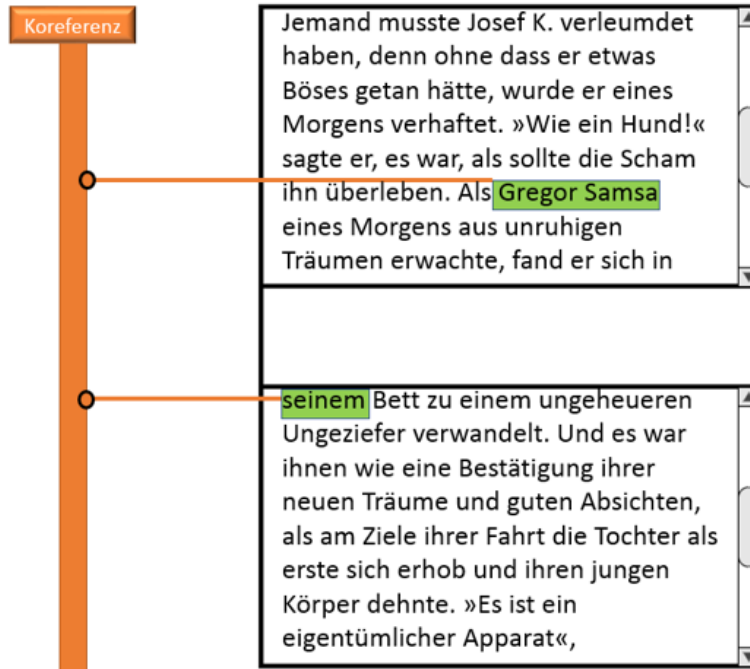
In der Summe eignet sich die Variante der eckigen Bögen besser bezüglich der Anforderungen an die Darstellung der Relationen. Der geringeren, aber dennoch in moderatem Rahmen vorhandene Verfolgbarkeit schließen sich der deutlich bessere Umgang mit Überschneidungen und Überlagerungen an, sodass die Wahl der Darstellung der Relationen zunächst auf die eckigen Bögen fällt.

### **Relationen in der Schwimmliniendarstellung**

Um nun auch die Skalierbarkeit in Anzahl und verwendeter Klassen der Relationen zu begünstigen, soll der Ansatz der eckigen Bögen erweitert werden.

Hierzu werden die Visualisierungsflächen zunächst statt für die Bögen, für sogenannte Schwimmlinien verwendet. Eine solche Schwimmlinie wird jeweils für jede unterschiedliche Relationenklasse in Form einer vertikal verlaufenden Linie von der oberen Kante der Benutzeroberfläche zur unteren Kante im freien Raum erzeugt. Mehrere Schwimmlinien werden dabei horizontal mittels eines relativen Abstands ausgehend vom rechten Rand der Visualisierungsfläche über diese verteilt.

Abbildung 4.8 zeigt, dass auch in der Schwimmliniendarstellung horizontal verlaufende Kanten von den zu verbindenden Annotationen weggeführt werden, welche dann über den Zeilenzwischenraum in die Visualisierungsfläche geführt werden. Statt mit einer dritten, vertikalen Kante für jede einzelne Relation wie bei den Bögen, werden die horizontal verlaufenden Kanten mit der Schwimmlinie verbunden, welche die Klasse der zu visualisierenden Relation repräsentiert. Die Schwimmlinien dienen also als Sammelpunkt sämtlicher Relationen derselben Klasse. Am Schnittpunkt der Kanten mit der Schwimmlinie wird ein Knoten erzeugt, der die vertikale Position der Kante und somit der verbundenen Annotation festhält.



**Abbildung 4.8:** Die Darstellung der Relation und ihrer Klasse als Schwimmlinie. Die Relation der Klasse "Koreferenz" erhält eine Schwimmlinie in der entsprechenden Farbe, sodass jedes Annotationspaar, für das diese Relation ausgezeichnet wird, mittels einer Kante mit der Schwimmlinie verbunden wird. Die Kante nimmt die Farbe der Klasse an.

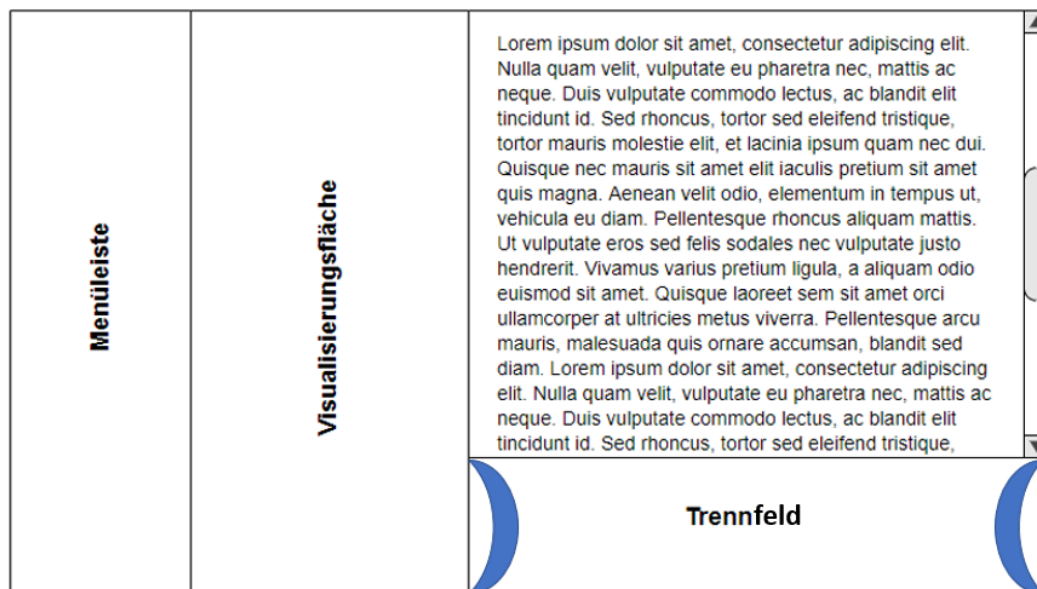
Die Ansammlung der Kanten von Relationen derselben Klasse an den Schwimmlinien führt zur strikten, visuellen Abgrenzung der Kanten voneinander. In der Folge lassen sich eng benachbarte Relationen unterschiedlicher Klassen leichter voneinander unterschieden und nachverfolgen. In Verbindung mit dem Highlight-Mechanismus kann der Nutzer jeder Relation eindeutig eine Klasse zuordnen.

Zugleich lösen die Schwimmlinien das Problem der Skalierung für Relationen derselben Klasse bis zu einem gewissen Grad: Durch die Verteilung der Kanten über die Länge der entsprechenden Schwimmlinie, wächst der durch die Visualisierung der Relationen belegte Raum bei steigender Anzahl an Relationen nicht mehr horizontal in den freien Raum. Stattdessen hängt die Anzahl visuell darstellbarer Relationen derselben Klasse von dem für die Schwimmlinie verfügbaren vertikalen Platz innerhalb der Benutzeroberfläche ab.

Der Ansatz der Schwimmlinien als Visualisierung der Relationen bietet also bereits eine mögliche Lösung bezüglich der Darstellung und der Skalierbarkeit von Relationen derselben Klasse und dahingehend auch der Überlagerung dieser. Probleme mit der Skalierbarkeit bezüglich mehrerer auftretender Klassen bleiben jedoch auch für diesen Ansatz bestehen. Da die Schwimmlinien unterschiedlicher Klassen horizontal über den freien Raum verteilt werden, hängt die Darstellung erneut von der Breite dieser ab. Auch mit der Verfolgbarkeit mehrerer Relationen unterschiedlicher Klassen hat der Ansatz der Schwimmlinien zu kämpfen, ähnlich dem Ansatz der Bögen.

Die Zuweisung der Annotationen, die durch die Relation verlinkt sind, wird dabei uneindeutig, wenn ausgezeichnete Relationen eng benachbart sind. Die Kanten der Relationen überlagern sich dabei im Zeilenzwischenraum, sodass die sich gegenseig verdecken. Zur Identifikation einzelner Relationen und ihrer verbundenen Annotationen wird dann der Highlight-Mechanismus benötigt.

### Relationen im kombiniertem Ansatz



**Abbildung 4.9:** Statt den Raum auf beiden Seiten der Ansicht frei zu lassen, werden Textansicht und Trennfeld an den rechten Rand verschoben, sodass nun der gesamte Raum zwischen Menüleiste und Textansichten als Visualisierungsfläche für Relationen dient.

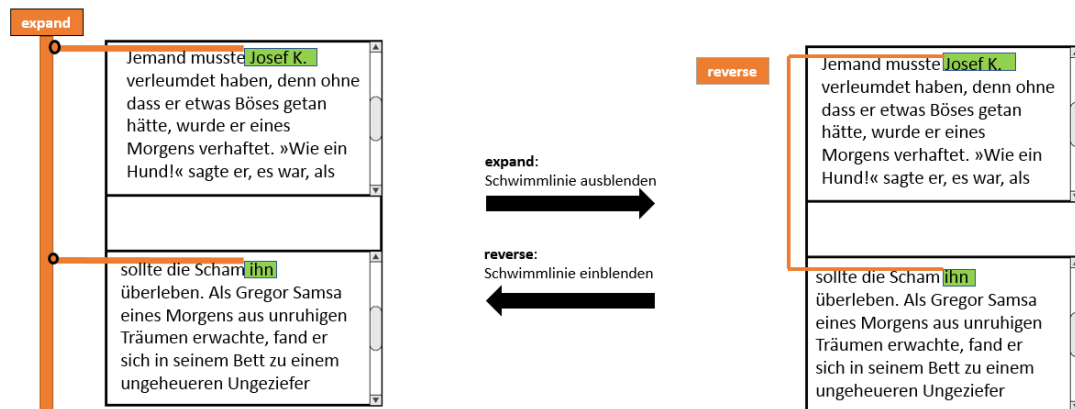
Vor der endgültigen Wahl für oder gegen die Schwimmliniendarstellung respektive den Bögen als Visualisierung der Relationen, kam zunächst die Frage auf, ob die Schwimmlinien im bisherigen Entwurf der Oberfläche auf beiden Seiten der Textansichten oder nur auf einer Seite gezeichnet werden sollen. Beide Varianten bringen Nachteile mit sich: Bei der Darstellung auf beiden Seiten würden beide Schwimmlinien dasselbe visualisieren, sodass Relationen doppelt dargestellt werden. Alternativ würden die zu zeichnenden Relationen alternierend an der einen

oder der anderen Schwimmlinie dargestellt werden, wobei das Entscheidungsmuster dem Nutzer transparent und verständlich erklärt werden müsste. Wird die Schwimmlinie dagegen nur in einer der Visualisierungsflächen gezeichnet, bleibt die andere Fläche ungenutzt.

In der Folge wurde die Benutzeroberfläche für die Darstellung der Schwimmlinien optimiert, indem die beiden Visualisierungsflächen zusammengelegt wurden. Abbildung 4.9 zeigt die veränderte Benutzeroberfläche.

Durch das Zusammenlegen der Visualisierungsflächen wird der Raum für und von den Schwimmlinien optimal genutzt, da diese weder dupliziert werden müssen, noch Raum ungenutzt bleibt. Zugleich schließt ein einzelner Visualisierungsraum die Bogendarstellung nicht aus: Diese können allesamt über die einzelne Fläche geführt werden, statt alternierend über zwei Flächen. Dies birgt potenziell den Nachteil, dass die Darstellung mehrerer, sich möglicherweise überschneidender Bögen unübersichtlich wird. Gleichzeitig aber erlaubt die Ansammlung der Bögen auf einer Seite einfachere Aussagen über die Quantität, schnellere Vergleiche untereinander und die bessere Verfolgbarkeit, da der Nutzer seine Aufmerksamkeit nicht mehr an zwei verschiedene Flächen widmen muss.

In der Summe bringt das Zusammenlegen der Flächen mehr Vorteile als Nachteile für die Bogendarstellung mit sich. Inklusiv ihrer Vorteile bezüglich der Schwimmlinienendarstellung, nutzt diese Variante der Benutzeroberfläche den verfügbaren Raum weitaus effektiver als der initiale Ansatz. Als positiver Nebeneffekt kommt hinzu, dass die visuelle Kapselung der Hauptbereiche so gefördert wird. Da sowohl die Schwimmlinienendarstellung, als auch die Bogendarstellung von der Umsetzung mit nur einer Visualisierungsfläche profitieren, ermöglicht sie zuletzt die effektive Kombination beider Ansätze, wie Abbildung 4.10 zeigt.



**Abbildung 4.10:** Die Kombination der Schwimmlinien und der Bogendarstellung. Über die Buttons "expand" und "reverse" kann zwischen den Darstellungen gewechselt werden.

Im kombinierten Ansatz kann der Nutzer zwischen beiden Darstellungen der Relationen wechseln. Wird eine Relation zwischen zwei Annotatoren ausgezeichnet, so wird die resultierende Verbindung zunächst in der Schwimmlinienendarstellung visualisiert. Über der betreffenden Schwimmlinie findet sich dann ein Button, mittels dessen die Schwimmlinie ausgeblendet werden kann. In der Folge wird jede Relation dieser Klasse anschließend als Bogen gezeichnet. Die Schwimmlinie wird während ihrer Ausblendung am linken Rand der Visualisierungsfläche mit stark erhöhter Transparenz



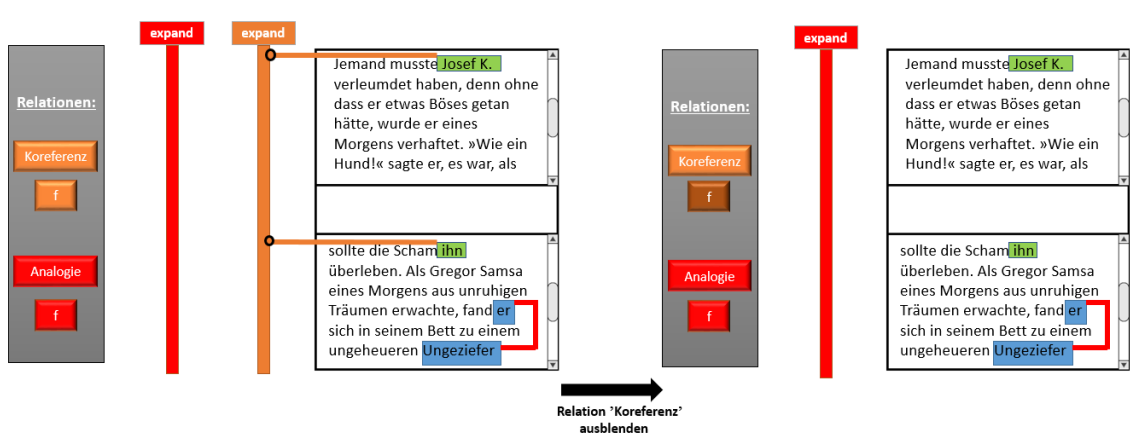
angedeutet. Werden neue Relationen dieser Klasse ausgezeichnet, werden sie ebenfalls als Bogen gezeichnet, bis die Schwimmlinie wieder eingeblendet wird. Die Wiedernutzung des Buttons bewirkt die Rückkehr zur Schwimmliniendarstellung.

Durch das Zusammenspiel von Bögen und Schwimmlinien verspricht der kombinierte Ansatz Variabilität in der Visualisierung der Relationen, sodass der Nutzer situativ und je nach Bedarf die passendere zwischen den beiden Varianten auswählen kann:

- Sind die ausgezeichneten Relationen in ihrer Anzahl gering bis moderat, so bietet es sich an, die Relationen als Bögen darzustellen. Besonders zum Tragen kommt dann die Stärke der Bogendarstellung bezüglich der Verfolgbarkeit, beispielsweise zur schnellen Identifikation beteiligter Annotationen. Zudem verspricht der Ansatz der Bögen dann eine bessere Übersicht über die Verteilung der Relationen. Durch die Verschiebung kosnekutiv gezeichneter Relationen lassen sich diese besser voneinander unterscheiden.
- Zuletzt kann es effizienter sein, bei einer geringen Varianz an vorkommenden Klassen die Bogendarstellung zu bevorzugen. Es existiert dann mehr Raum zur Darstellung der Bögen, gleichzeitig muss nicht für jede Klasse eine Schwimmlinie gezeichnet werden.
- Werden dagegen viele, gegebenenfalls unterschiedliche Relationen ausgezeichnet, kann die Darstellung durch Schwimmlinien diesen Sachverhalt besser darstellen. Wie zuvor erörtert, skalieren diese im Vergleich besser mit steigender Anzahl an ausgezeichneten Relationen.
- Tritt der zuletzt genannte Fall ein, leidet die Verfolgbarkeit individueller Relationen durch die größere Menge an Relationen im Blickfeld zwangsläufig, unabhängig von der gewählten Darstellung. Oft reicht dann eine Andeutung der Position der Annotationen, sodass hier die Vorteile der Darstellung via Schwimmlinien besser zur Geltung kommen.
- Im Falle dicht bevölkerter relationierter Annotationen können beide Darstellungen hilfreich sein. Interessiert sich der Nutzer für einen Textbereich, der viele solcher Annotationen aufweist, so muss er abwägen, welche Art der Information er priorisiert. Will er eine geeignete Übersicht über die Relationen und ihre Klassen, so bietet sich die Schwimmliniendarstellung an. Will der Nutzer jedoch die Existenz der Relationen zwischen ganz bestimmten Annotationen überprüfen, so hat er im kombinierten Ansatz die Option, zur Bogendarstellung zu wechseln.

### **Filterung von Relationen**

Neben der Vermeidung von Überlagerung durch die Wahl einer geeigneten Darstellungsweise der Relationen, kann dies auch durch interaktive Maßnahmen erreicht werden. Untersucht werden soll die Idee, dass es in vielen Anwendungen hilfreich sein kann, bestimmte Informationen auszublenden, um einen besseren Fokus auf andere Teile zu generieren. Im Kontext der Relationen soll dem Nutzer dabei die Möglichkeit gegeben werden, die visuelle Repräsentation von Relationenklassen temporär auszublenden, um die Darstellung auf eine oder mehrere, ausgewählte Klassen zu begrenzen. Die Ausblendung der anderen Klassen bewirkt so die gezielte Filterung relevanter Klassen von Relationen und kann dahingehend verwendet werden, um sich überlagernde oder überschneidende Relationen individuell zu betrachten. Der Ansatz soll also so erweitert werden, dass der Nutzer interaktiv sämtliche Instanzen beliebig gewählter Klassen von Relationen aus- und wieder einblenden kann.



**Abbildung 4.11:** Es wurden die Klassen "Koreferenz" und "Analogie" und Instanzen dieser erstellt. Durch das Drücken des designierten Buttons werden sämtliche Instanzen der Klasse "Analogie" ausgeblendet, inklusive der Schwimmlinie. Die Ausblendung wird dem Nutzer zusätzlich signalisiert, indem die Opazität des Filter Buttons vermindert wird. Die Annotationen verbleiben.

Die Filterfunktion wird integriert, indem zunächst designierte Buttons für jede einzelne Relationenklasse in der Menüleiste eingeführt werden. Drückt der Nutzer nun den Filter-Button einer Relationenklasse, werden sämtliche Instanzen der Klasse ausgeblendet. Die Ausblendung funktioniert unabhängig von und für beide Darstellungen der Relationen, sodass in der Schwimmlinienendarstellung neben den Instanzen selbst auch die Schwimmlinie der betroffenen Klasse ausgeblendet wird. Die Annotationen, die durch die Relationen in Beziehung gesetzt werden, bleiben indes erhalten und sind weiterhin visuell hervorgehoben. Ein erneutes Drücken des Buttons der betroffenen Klasse bewirkt dann das erneute Einblenden der Instanzen und der Schwimmlinie.

Mithilfe der Filterfunktion kann der Nutzer temporär, zu beliebigen Zeitpunkten, die Darstellung der Relationen auf relevante Klassen begrenzen. Sie ermöglicht dem Nutzer, priorisierte Informationen aus einer temporär reduzierten Visualisierung der Relationen zu gewinnen. Neben der bereits genannten Vermeidung von Überschneidung und Überlagerung von Relationen, kann die Filterung auch genutzt werden, wenn ein ausgewählter Ausschnitt des Textes mit relationierten Annotationen überladen ist. Die Ausblendung einzelner Klassen erhöht dann die Übersichtlichkeit. Die Ausblendung von Relationen hilft zudem temporär bei der Darstellung vieler Relationen, da so Platz für die zu zeichnenden Verbindungen geschaffen wird. Hiervon profitiert insbesondere die Bogendarstellung, die mehr horizontalen Platz benötigt als die Schwimmlinienendarstellung.

### Verhalten der Annotationen und Relationen bei Scroll Operationen

Durch die Trennfelder wird es dem Nutzer ermöglicht, den zu bearbeitenden Text zunächst im Rahmen eines Splits an einer beliebigen Stelle aufzubrechen und auf zwei neue Ansichten zu verteilen. In beiden Ansichten kann er in der Folge ausgewählte Textsegmente annotieren und in Beziehung stehende Textsegmente mittels einer Relation hervorheben. Werden die Annotationen in unterschiedlichen Ansichten ausgezeichnet, ermöglichen die Bänder der Trennfelder Aussagen über die Textmenge, die zwischen den relationierten Annotationen liegt.

Die Visualisierung der entfernten Relationen passt sich hierbei an den sich in der Scroll Operation bewegenden Text an, indem sie Form, Länge und Position der Verbindung an die Bewegung anpasst. Selbiges gilt, wenn sich die Position der betroffenen Annotationen durch einen Split innerhalb der Ansichten verändert. In der Schwimmliniendarstellung verschieben sich die verbindenden Kanten zwischen Annotation und Schwimmlinie vertikal mit der betroffenen Annotation mit. Ähnliches gilt für Relationen als Bögen, wo die vertikal verlaufenden Kanten länger oder kürzer werden. Der Prozess endet vollständig, wenn der Nutzer die Scroll Operation beendet oder teilweise, wenn eine oder beide Annotationen aus dem Sichtbereich gescrollt wurden. Visuell lässt sich das Verlassen des Sichtbereichs der Annotationen als Transition dieser in den Bereich unter oder über der Ansicht übersetzen.

Im diesem Fall werden in beiden Darstellungen der Relationen die verbindenden Kanten zu den nicht mehr sichtbaren Annotationen ober- oder unterhalb der Textansicht aggregiert und als einzelne Kante dargestellt. Dem Nutzer wird dadurch signalisiert, dass sich innerhalb diesen Bereichs nicht sichtbare relationierte Annotationen auffinden lassen.

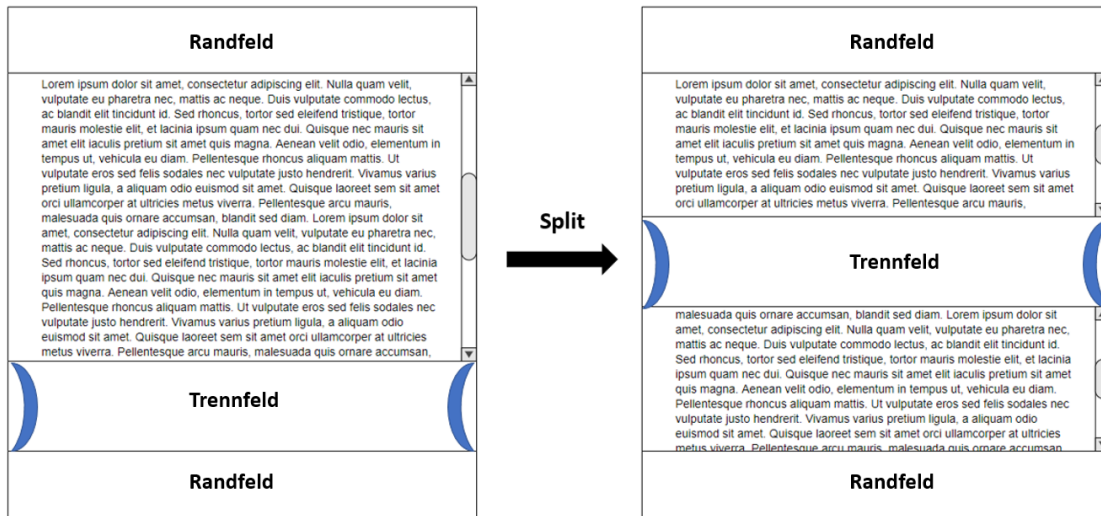
Informationen über die Relationen und ihre Klasse, beispielsweise die Anzahl der Annotationen, die durch Relationen einer bestimmten Klasse verbunden wurden, gehen jedoch temporär verloren, wenn die Annotationen aus dem Sichtbereich der betroffenen Ansicht gescrollt werden.

Im Falle der entfernten Relationen dient die zuvor beschriebene Aggregation als Indikator für die Existenz der Relationen. Für lokale Relationen geht jedoch auch diese Information verloren, sobald sich neben den relationierten Annotationen auch die visuelle Verbindung der Annotationen nicht mehr im aktuell sichtbaren Bereich der Textansicht befindet.

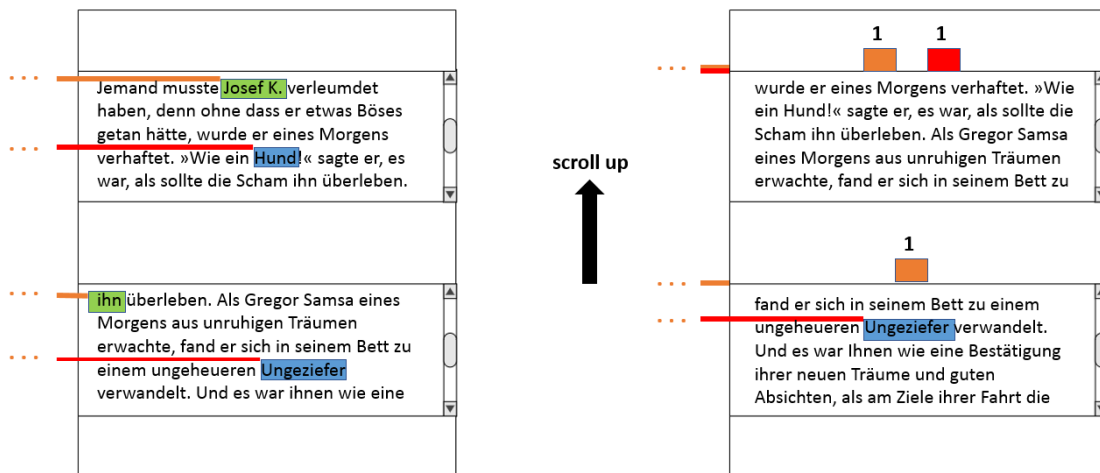
Um dem Verlust der Informationen vorzubeugen, soll die Oberfläche des Systems um zwei weitere Grenzflächen, ähnlich dem Trennfeld, erweitert werden. Die Größe und Positionierung dieser Randfelder ist fix und verändert sich nicht: Ein Randfeld oberhalb aller Textansichten, das andere Randfeld unterhalb aller Textansichten. Abbildung 4.12 zeigt die Benutzeroberfläche mit den Randfeldern vor und nach dem Split.

Das bisher für den Split der Ansichten und zur Visualisierung der Distanz genutzte Trennfeld, erhält nun eine weitere Aufgabe, welcher sich auch die Randfelder widmen: Werden annotierte Textsegmente, die Teil einer Relation sind, aus dem Sichtbereich der Ansicht gescrollt, wird die Transition der Annotationen in das angrenzende Randfeld visuell hervorgehoben. Hierbei wird die Quantität der verborgenen Annotationen visualisiert, indem gezählt wird, wie viele Annotationen von Relationen einer bestimmten Klasse betroffen sind. Die Summe wird für jede vorkommende Klasse der Relationen ausgewertet und der resultierende Datensatz als Balkendiagramm innerhalb des Randfeldes umgesetzt, wie Abbildung 4.13 zeigt. Die Zahl der verborgenen Annotationen übersetzt sich in die Höhe des Balkens. Wurde eine Relationenklasse ausgeblendet, so wird diese im Balkendiagramm ausgeschlossen.

Die Anzeige der Anzahl verborgener Relationen beantwortet die Frage, ob und wie viele relationierte Annotationen nicht zu sehen sind. Ist dabei nur eine Annotation der Relation verborgen und die andere sichtbar, gibt es bisher keine Möglichkeit zuzuweisen, welche der sichtbaren Annotationen einer Relation betroffen sind. Um dem entgegenzuwirken, werden die Balkendiagramme interaktiv ergänzt: Fährt der Nutzer über die Balken der jeweiligen Klassen, werden die Verbindungen zu den betroffenen Annotationen der Instanzen dieser Klasse per Highlight hervorgehoben. Das gilt auch für die Annotationen, falls diese sichtbar sind.



**Abbildung 4.12:** Durch die fixe Positionierung der Randfelder wird jede der Splitsichten nach dem Split oben und unten jeweils durch eine Grenzfläche eingeschlossen. Die Breite der Randfelder wird gleich den Textansichten gewählt.



**Abbildung 4.13:** Es wurden die Klassen "Koreferenz" (orange) und "Analogie" (rot) und Instanzen dieser erstellt. Durch das Scrollen in beiden Splitsichten verändert sich die Position der relationierten Annotationen: In der oberen Ansicht verschwinden sowohl das Mitglied der "Koreferenz" Relation, als auch das Mitglied der "Analogie" Relation im Bereich des oberen Randfelds. Das Diagramm zeigt diesen Sachverhalt in dem Randfeld an. In der unteren Ansicht gilt Selbiges nur für das Mitglied der "Koreferenz" Relation, entsprechend zeigt Trennfeld nur einen Balken.

Die interaktive Visualisierung der verborgenen relationierten Annotationen, die durch die iterative Auswertung der Quantität bei jeder neuen Scroll Operation des Nutzers adaptiv und konsistent ist, nimmt dem Nutzer die Aufgabe ab, Relationen und ihre Annotationen händisch nachverfolgen zu müssen. Über das Balkendiagramm in den Randfeldern erhält der Nutzer eine quantitative Aussage über die Relationenklassen in einem bestimmten Textabschnitt.

Über die Interaktion mit den Balken hat der Nutzer zudem die Möglichkeit, die jeweiligen Teilnehmer der verborgenen Relationen ausfindig zu machen, da im besten Fall der sichtbare Teil der Relation durch das Highlight hervorsteht, im schlechtesten Fall verrät das Highlight der Verbindungen, wo und in welchen Ansichten sich die betroffenen Annotationen befinden.

Ein interessanter Punkt ist weiterhin, wie weit sich nicht sichtbare relationierte Annotationen bei Scroll Operationen vom aktuell sichtbaren Textabschnitt wegbewegen. Zur Visualisierung dieser Information wurden relationierte Annotationen ursprünglich nach ihrer Transition in die Grenzflächen innerhalb dieser dargestellt. Nach der Transition wurden die Annotationen als einzelne, kleine Knoten im betroffenen Trenn- oder Randfeld dargestellt, die sich abhängig von der Scroll Richtung des Nutzers in der Grenzfläche auf- und ab bewegten. Die vertikale Position des Knotens gab dabei an, wie weit die betroffene Annotation vom sichtbaren Textabschnitt entfernt war: Je höher der Knoten sich innerhalb des Trennfelds befand, desto weiter weg war die Annotation. Der Ansatz wurde letztendlich nicht in der Umsetzung berücksichtigt, da die exakte Berechnung der Position für eine potenziell hohe Anzahl an relationierten Annotationen kostenaufwändig ist. Die Position müsste mit jeder Scroll Operation neu ausgewertet werden, um die Konsistenz zu wahren.



# 5 Implementierung

Nachdem im vorangegangenen Kapitel die getroffenen Designentscheidungen und die Umsetzung dieser gründlich beleuchtet wurden, soll in diesem Kapitel die Implementierung präsentiert werden. Dazu werden zunächst der Aufbau des Systems inklusive seiner Datensätze und die verwendete Technologie näher beschrieben. Im Anschluss wird eine Übersicht über die fertige Benutzeroberfläche erstellt und die Bestandteile dieser detailliert vorgestellt.

## 5.1 Technologie

Im folgenden werden die im entwickelten Ansatz verwendeten Technologien beschrieben und ihre Einsatzgebiete erläutert.

### 5.1.1 System

Das System ist als Single-Page-Webapplikation realisiert. Entsprechend wird das HTML-Dokument, auf dem das System basiert, folgendermaßen aufgebaut, manipuliert und erweitert:

- Im Kern liegt der zu bearbeitende Text als HTML-Text vor. Mithilfe von `Tokenizer.js`<sup>1</sup> wird der Text dabei zunächst in Token unterteilt. Die Token werden jeweils als einzelne `Span`-Elemente in die Dokument Struktur eingefügt. Dies hat zur Folge, dass jedes Token als Knoten innerhalb des Baumes, welcher die Hierarchie des *Document Object Model* (DOM) repräsentiert, existiert. Jeder dieser Knoten dient anschließend als Ansatzpunkt für den Split des Textes und für die Erstellung und Visualisierung von Annotationen. Jedes `Span`-Element erhält eine eindeutige Identifikationsnummer, die folgend als ID abgekürzt werden soll.
- Die verschiedenen Textansichten, das heißt die initial Ansicht und alle aus Splits resultierenden Splitansichten, und die Grenzbereiche, bestehend aus dem Trennfeld und den beiden Randfeldern, sind zu Interaktionszwecken als separate DOM-Objekte in die Struktur eingebettet. Dies ermöglicht das Trennfeld im Rahmen einer Drag-and-drop Aktion zu bewegen, während die mögliche Fläche zum Ablegen des Trennfelds auf die Textansichten begrenzt werden kann. Fehlerzustände, in welchen das Trennfeld auf einen der anderen Grenzflächen abgelegt wird, werden so vermieden.

---

<sup>1</sup><https://gist.github.com/borgar/451393>

- Mithilfe von JavaScript, insbesondere JQuery.js<sup>2</sup> und JQuery-UI.js<sup>3</sup>, werden die wichtigen Funktionen zur Umsetzung interaktiver Veränderungen an dem Dokument und seiner Struktur realisiert. Dazu gehören die Bewegung des Trennfelds, der Split der Textansichten und die dadurch folgende Verteilung des Textes auf zwei neue Splitansichten, Berechnungen bezüglich der darzustellenden Distanz respektive der Formveränderung der Bänder und die Berechnung der Position und Länge der Kanten der Relationen. Auch das Highlight von Annotationen und Relationen beim Bewegen der Mauszeiger über diese wurde mit JQuery.js umgesetzt.
- Die farbliche Hervorhebung der Textsegmente als Visualisierung der Annotationen und das Styling der Applikation und ihrer Bestandteile wie den verschiedenen Textansichten, den Grenzbereichen und der Menüleiste inklusive ihrer Buttons wird hauptsächlich mittels CSS erreicht. Für die Darstellung der Relationen, der Balkendiagramme innerhalb der Grenzbereiche und den Bändern wird D3.js<sup>4</sup> in Kombination mit SVG<sup>5</sup> verwendet. Dabei wird eine SVG-Ebene hinter den HTML-Objekten positioniert und dort die Relationen eingezeichnet. Dies hat den Vorteil, dass sich die Relationen stets hinter dem Text befinden, aber sichtbar bleiben und somit den Text nicht komplett verdecken.

### 5.1.2 Datensatz

Der Datensatz, mit welchem das System arbeitet, umfasst im wesentlichen

- den zu bearbeitenden Text
- die erstellten Annotationen
- und die ausgezeichneten Relationen

Jede durch den Nutzer erstellte Annotation wird temporär als JavaScript-Objekt der folgenden Form abgespeichert:

```
{
  "id" : "",
  "class": {
    "tag" : "",
    "color" : ""
  },
  "token" : [],
  "relation" : {
    "member" : []
  }
}
```

Eine Instanz der aktuell aktiven Annotationsklasse erzeugt der Nutzer, indem er im Annotationsmodus einzelne zu markierenden Texttoken anklickt oder den Mauszeiger über ganze Abschnitte zieht und diese so selektiert. Zusätzlich hat der Nutzer in diesem Modus die Option, eine lange Annotation über den Doppelklick der betroffenen Annotationen zu erzeugen. Damit das System die

---

<sup>2</sup><https://jquery.com/>

<sup>3</sup><https://jqueryui.com/>

<sup>4</sup><https://d3js.org/>

<sup>5</sup>[https://de.wikipedia.org/wiki/Scalable\\_Vector\\_Graphics](https://de.wikipedia.org/wiki/Scalable_Vector_Graphics)



jeweilige Interaktion unterscheiden kann, wartet dieses nach einem einzelnen Klick einige hundert Millisekunden ab, ob gegebenenfalls ein weiterer Klick im Rahmen eines Doppelklicks folgt und ruft entsprechend die geeignete Methode auf. Das Warten des Systems äußert sich dabei in einer kurzen, aber notwendigen Verzögerung für jede Annotation.

Der eigentlichen Annotation vorangehend, erzeugt der Nutzer zunächst eine neue Klasse. Der eindeutige Tag und das eindeutige Farbattribut der Klasse füllen das "class" Feld für die neu erstellte Annotation aus. Das Feld "token" listet die vom Nutzer selektierten Token des Textes auf, genauer werden hier die Span-Elemente der Token vermerkt. Wird die zugrundeliegende Annotation mit einer weiteren Annotation im Rahmen einer binären Relation verlinkt, so wird die ID der durch die Relation referenzierten Annotation im Feld "member" abgelegt.

Nach der Erstellung einer neuen Relationenklasse, wird eine neue Instanz dieser Klasse im Relationenmodus durch das Anklicken zweier bereits existierender Annotationen erzeugt.

```
{
  "ids" : "",
  "class": {
    "tag" : "",
    "color" : ""
  },
  "type" : "",
  "visibility" : true
}
```

Jede Relation wird dabei als Objekt mit den IDs der durch die Relation verbundenen Annotationen im Feld "ids", der Art der Relation im Feld "type", ihrer Sichtbarkeit im Feld "visibility", sowie ihrem eindeutigen Tag und eindeutigen Farbattribut im Feld "class" realisiert. Die Art einer Relation beschreibt hierbei, ob es sich um eine lokale oder entfernte Relation handelt. Die Sichtbarkeit wird für die Auswertung der Balkendiagramme benötigt und wird initial auf *true* gesetzt.

Sämtliche ausgezeichneten Tags und Farbattribute der Klassen, sowohl für Annotationen als auch für Relationen, werden in separaten Maps abgespeichert, um effizient die Eindeutigkeit neuer Klassen überprüfen zu können.

## 5.2 Benutzeroberfläche

Abbildung 5.1 hebt mittels Nummerierung die bereits im Rahmen des Entwurfs spezifizierten Bestandteile der Benutzeroberfläche hervor, welche nun implementiert wurden. Folgend sollen sie detailliert beschrieben werden:

- Die übergreifende Textansicht in (1) enthält initial den kompletten zu bearbeitenden Text ohne Struktur. Per Drag-and-drop des Trennfelds wird diese Ansicht durch zwei neue Textansichten ersetzt, auf welche der Text der Split Position entsprechend aufgeteilt wird. Strukturell gleichen die Splitansichten der ursprünglichen Ansicht und lassen sich als Ziel konsekutiver Splits mithilfe des Trennfeldes auswählen. Wird eine Ansicht im Rahmen eines Splits



Abbildung 5.1: Die finale Benutzeroberfläche des entwickelten Ansatzes.

ersetzt, so gilt, dass die Summe der Höhe der beiden Splitansichten wieder die Höhe der ursprünglichen Ansicht ergibt. Die beiden Splitansichten müssen jedoch nicht gleich hoch sein. Zur Veranschaulichung kann 6.2 in Anwendungsszenario 6.1.1 betrachtet werden.

- Das Trennfeld in (2) beinhaltet die als *SVG-Path* realisierten Bänder. Beim Drag-and-drop des Bandes im Rahmen eines Splits, dient dem Nutzer eine horizontal verlaufende, gestrichelte Linie innerhalb des Trennfeldes als Indikator für die aktuelle Position der Maus und damit für die aktuelle Splitposition. Die Linie ist nur während des Drag-Prozesses sichtbar. Die Formveränderungen der Bänder bei steigender oder sinkender Distanz wird bei jeder Scroll Operation des Nutzers aktualisiert. Dazu werden innerhalb der jeweiligen Splitansichten die Position der Span-Elemente, welche die Token enthalten, mit der Höhe der Ansichten verglichen und so das jeweils letzte sichtbare Token mit ID  $i$  und das erste sichtbare Token mit ID  $j$  der oberen respektive unteren Ansicht bestimmt. In Grenzbereichen verborgen liegen dann sämtliche Token mit ID  $k$ ,  $i \leq k \leq j$ , sodass die Texttoken unter diesen zusammengerechnet die Distanz ergeben. Die konkrete Zahl wird dem Nutzer dabei nach jeder Scroll Operation mittig innerhalb des Trennfeldes angezeigt, ehe die Anzeige nach einer vordefinierten Zeitspanne wieder verschwindet. Abbildung 6.8 in Anwendungsszenario 6.1.4 zeigt den eben beschriebenen Prozess.
- Wie im Entwurf spezifiziert, ist die Position der Randfelder in (3a) und (3b) fix. Bei der Transition von relationierten Annotationen in die Randfelder wird mittels D3 ein Balkendiagramm in das betroffene Band gezeichnet. Über den jeweiligen Balken wird die Anzahl betroffener Annotationen als Zahl angezeigt. Die Breite und die Positionierung der einzelnen Balken wird anhand der verfügbaren Breite des Bandes und der Anzahl an Relationenklassen, die betroffen sind, berechnet. Die Höhe der einzelnen Balken ist beschränkt auf die Höhe des Randfeldes minus einiger weniger Pixel, damit die Anzeige

der Zahl Platz findet. Die Nachteile dieser Entscheidung werden im Kapitel Diskussion aufgegriffen. Anwendungsszenario 6.1.6 zeigt die Erzeugung der Balkendiagramme in einem konkreten Anwendungsfall.

- Die Visualisierungsfläche in (4) dient zur Darstellung der entfernten Relationen, sei es in der Bogen- oder der Schwimmliniendarstellung. Unabhängig von der Art der Darstellung sind die Relationen als Zusammenschluss von horizontalen und vertikalen *SVG-Lines* realisiert. Buttons zum Ausblenden der Schwimmlinien, was den Übergang in die Bogendarstellung initiiert, finden sich hier über den korrespondierenden Schwimmlinien. Dieselben Buttons werden auch zum erneuten Einblenden und der entsprechenden Rückkehr in die Schwimmliniendarstellung der Relationen genutzt.
- Die Menüleiste in (5) enthält Buttons mit verschiedenen Funktionalitäten.
  - (5a) zeigt die beiden Buttons zum interaktiven Wechsel zwischen dem Annotations- und dem Relationenmodus. Wie zuvor beschrieben, können Annotationen ausschließlich im Annotationsmodus erstellt werden. Dieser ist standardmäßig aktiv, was durch die farbliche Hervorhebung des dazugehörigen Buttons signalisiert wird. Wechselt der Nutzer per Druck der entsprechenden Buttons in den Relationenmodus, können nun existierende Annotationen im Rahmen einer Relation verbunden werden.
  - (5b) dient zur Erstellung neuer Relationen, für welche das entsprechende Farbattribut und ein Tag ausgewählt werden können. Nach der Erstellung wird die neue Relation initial in Form einer assoziierten Schwimmlinie derselben Farbe in (4) dargestellt. (5c) realisiert auf ähnliche Weise die Erzeugung neuer Annotationen, wobei mittels Farb- und Tag Wahl eine neue Klasse erstellt wird. Für jede neu erstellte Klasse an Relationen und Annotationen werden in den designierten Fenstern neue Buttons angezeigt. Werden die Buttons für Relationen respektive Annotationen angeklickt, so aktiviert dies die dazugehörige Klasse. Der Nutzer erkennt dabei die aktuell aktive Annotations- bzw. Relationenklasse dadurch, dass der Button der Klasse per Highlight hervorgehoben wird. Für die aktive Relationenklasse wird zusätzlich die Schwimmlinie der aktiven Klasse visuell hervorgehoben, indem sie breiter und heller gezeichnet wird. Neben jedem Klassen-Button einer Relation wird zudem ein Filter-Button für die Relationenklasse erstellt. Das Drücken dieses Buttons führt zur Ausblendung sämtlicher Instanzen der Klasse und ihrer Schwimmlinie. Die Erstellung neuer Relationen einer ausgeblendeten Klasse ist möglich, jedoch sind diese ebenfalls nicht sichtbar, bis die Relationen der Klasse mithilfe desselben Buttons wieder eingeblendet werden. Die *redo*-Buttons entfernen jeweils die letzte erstellte Annotation beziehungsweise Relation. Im Falle einer entfernten Annotation, die Teil einer Relation war, wird auch die Verbindung zum anderen Teilnehmer der Relation entfernt. Abbildung 6.4 in Anwendungsszenario 6.1.2 zeigt die Benutzeroberfläche nach der Erstellung von Annotations- und Relationenklassen.
  - (5d) dient dem Nutzer zur Spezifikation der Split Position, wenn der Text anhand des Trennfelds aufgeteilt wird. Im Modus "onWord" wird der Text an dem Token respektive dem Word geteilt, an welchem sich der Mauszeiger beim Ablegen der Bandfläche befindet. Im Modus "next" wird ausgehend von diesem Wort zum nächstgelegenen Satzende gesprungen und das Token dort als Ausgangspunkt ausgewählt. Analog verhält es sich im Modus "prev" und dem Satzanfang. Wurde der Split initiiert, wird der Text

der betroffenen Ansicht am Ausgangspunkt auf die neuen Splitansichten verteilt. Stets gilt dabei, dass die ursprüngliche Ansicht durch die Splitansicht mit dem oberen Teil des Textes ersetzt wird, während der untere Teil in der neuen Ansicht unterhalb des Trennfeldes platziert wird. Nach Abschluss des Splits wird die obere Ansicht initial maximal nach unten gescrollt, während die untere Ansicht maximal nach oben gescrollt wird. Dadurch wird sichergestellt, dass die Distanz zwischen den Ansichten initial 0 ist.

- Mittels (5e) kann der Nutzer die Ausrichtung der Bänder und ihr Verhalten bei Distanzveränderung invertieren. In ihrer Initialform, das heißt wenn der User noch keine Scroll Operationen innerhalb der Splitansichten vollzogen hat, sind die Bänder gestreckt. Veränderungen in der Distanz zwischen den Texten durch das Scrollen des Nutzers bewirken anschließend die Abrundung der Bänder. Wird in der Folge die Ausrichtung und das Verhalten mittels des Buttons invertiert, wird die Visualisierung unmittelbar angepasst, indem die Bänder der aktuellen Distanz entsprechend gestreckt werden. Für weitere Scroll Operationen werden die Bänder dann weiter gestreckt, umso mehr die Distanz steigt.
- (5f) wird zu einem aktiven Button, sobald die initiale Textansicht im Rahmen eines Splits geteilt wurde. Dann kann der Nutzer durch das Anklicken dieses Buttons sämtliche Splits rückgängig machen und in die initiale Darstellung des Textes in einer umfassenden Ansicht zurückkehren. Das Trennfeld wird an seine ursprüngliche Position gesetzt, während erstellte Annotationen wie Relationen erhalten bleiben. Für Relationen gilt dabei, dass jede entfernte Relation nun als lokale Relation innerhalb der verbleibenden Ansicht dargestellt wird. Anschließend wird der Button wieder deaktiviert und ihre Fläche ausgegraut, bis ein neuer Split vollzogen wird.

## 6 Ergebnisse

Dieses Kapitel widmet sich den Ergebnissen dieser Arbeit. Der Beginn des Kapitels beschäftigt sich mit der Besprechung einiger ausgewählter Anwendungsfälle, welche die Nutzung des entwickelten Ansatzes verdeutlichen. Darauf folgend werden die Ergebnisse einer Expertenbefragung zum entwickelten Ansatz präsentiert. Den Schluss dieses Kapitels bildet die Diskussion und die Bewertung des entwickelten Ansatzes.

### 6.1 Anwendungsfälle

Nachfolgend wird die Nutzung des entwickelten Ansatzes anhand von einigen Beispielaufgaben der Textannotation und Visualisierung verdeutlicht. Als Beispieltext werden die ersten 5 Kapitel des Buches "Die Verwandlung" von Franz Kafka <sup>1</sup> genutzt, welche vorab in das HTML-Dokument geladen und zusammenhängend in der initialen Ansicht angezeigt werden.

#### 6.1.1 Annotation des Textes in einer Splitansicht

Das folgende Szenario soll zeigen, wie der entwickelte Ansatz die Annotation eines Textes in mehreren Ansichten unterstützt. Dazu werden zunächst einige Personenerwähnungen und Referenzen auf diese Personen annotiert, beispielsweise die Person "Gregor Samsa" und das Pronomen "er".

Um Annotationen erstellen zu können, werden einige neue Annotationsklassen erzeugt. Zunächst wird eine Klasse mit dem Tags "Person" und der Farbe orange mithilfe des designierten "create"-Buttons in der Menüleiste erstellt. Für die neu erstellte Klassen wird dann im Fenster unter dem "create"-Button ein neuer Klassen-Button mit dem Tag der Klasse als Bezeichner erstellt, wie Abbildung 6.1 zeigt. Nach der Erstellung ist die Klasse unmittelbar aktiv, sodass nun Instanzen der Klasse "Person" erstellt werden können. Letztlich muss noch der Annotationsmodus aktiv sein, um den Prozess der Annotation zu starten. Dies ist standardmäßig der Fall, was durch die farbliche Hervorhebung des "Ann. Mode"-Buttons verdeutlicht wird.

Um den Fokus auf einen kleineren Ausschnitt zu legen und die Annotation in diesem Teil des Textes zu beginnen, wird der Text mithilfe des Trennfeldes geteilt. Abbildung 6.1 zeigt, dass für den Split der Modus "next" aktiv ist. Entsprechend wird der Text am Satzende geteilt, ausgehend vom Wort, über welchem sich der Mauszeiger beim Ablegen des Trennfeldes befindet. Nach dem Split wird dieser Textausschnitt annotiert, indem die Erwähnungen von Personen angeklickt werden. Abbildung 6.2 zeigt die Benutzeroberfläche nach dem Split des Textes und der anschließenden Annotation der vorkommenden Personen "Gregor Samsa", "Dame" und "Beschauer".

---

<sup>1</sup><http://gutenberg.spiegel.de/buch/die-verwandlung-9760/1>

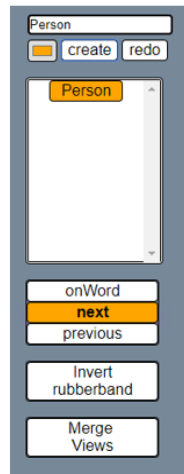
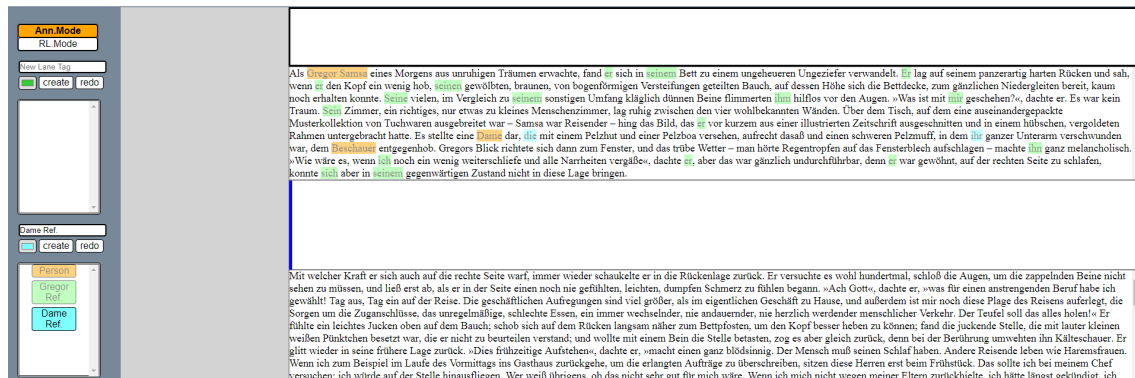


Abbildung 6.1: Die Menüleiste nach der Erstellung der Annotationsklasse "Person"



Abbildung 6.2: Das Trennfeld wird in den Text gezogen und am Wort "Lage" losgelassen. Wegen dem "next"-Modus wird der Text dann am nächsten Satzende getrennt. Anschließend werden die drei Personenerwähnungen annotiert. Die Erwähnungen "Gregor" und "Samsa" werden dabei ausgelassen, da sie dieselben Referenzen mit "Gregor Samsa" teilen.

Die Annotation des Abschnitts wird fortgesetzt, indem nun die Referenzen annotiert werden. Dazu erhalten die abhängigen Referenzen jeder unterschiedlichen Person ihre eigene Annotationsklasse. Erstellt werden die Klasse "Gregor Ref." für Referenzen auf Erwähnungen von Gregor Samsa und die Klasse "Dame Ref." für Referenzen auf die Dame. Da keine Referenzen auf "Beschauer" im Laufe des Textes erscheinen, wird hier keine zusätzliche Klasse erstellt. Abbildung 6.3 zeigt die Menüleiste nach der Erstellung der neuen Annotationsklassen und den Text nach der Annotation der Referenzen.



**Abbildung 6.3:** Für die Klasse "Gregor Ref." wird die Farbe grün und für "Dame Ref." die Farbe blau gewählt, wie anhand der erstellten Klassen-Buttons zu erkennen ist. In der Folge werden zuerst die Referenzen von Gregor Samsa und dann die Referenzen von Dame annotiert. Entsprechend ist die Klasse "Dame Ref." zuletzt aktiv, was an der Hervorhebung des korrespondierenden Klassen-Buttons zu erkennen ist.

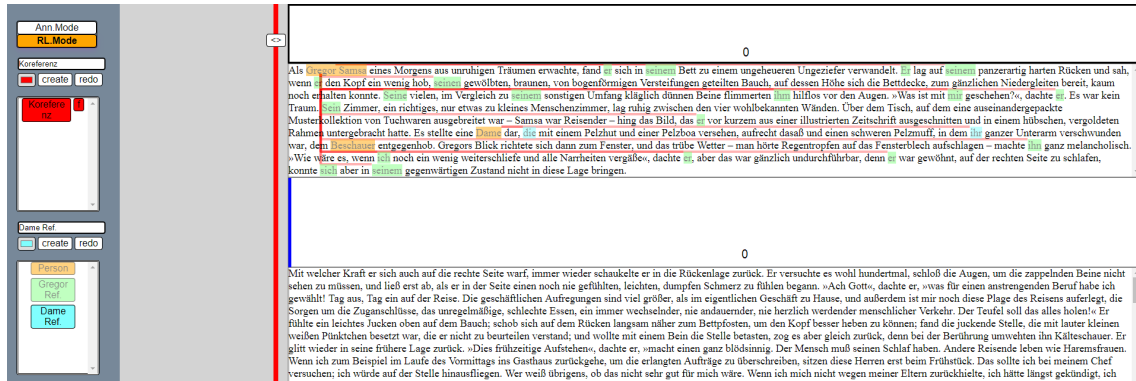
### 6.1.2 Auszeichnung von Relationen

Im folgenden Szenario soll die Auszeichnung von Relationen demonstriert werden. Dazu wird das vorherige Szenario fortgesetzt, indem nun die Verbindungen zwischen den Personen und ihren abhängigen Referenzen visuell verdeutlicht werden. Dazu wird eine neue Relation mit dem Tag "Koreferenz" und der Farbe rot erstellt und zwischen den jeweiligen Personenerwähnungen und ihren Referenzen ausgezeichnet. Die neue Relationenklasse wird mit dem designierten "create"-Button erstellt, sodass ein neuer Klassen-Button im Feld darunter erscheint. Neben dem Klassen-Button wird zudem der Filter-Button der Klasse "Koreferenz" angezeigt, welcher zur Ausblendung sämtlicher Instanzen dieser Klasse genutzt werden kann. Mit der Erstellung der neuen Klasse, wird auch eine neue, rot gefärbte Schwimmlinie für die Klasse "Koreferenz", rechtsbündig in der Visualisierungsfläche erzeugt. Oberhalb der Schwimmlinie befindet sich zudem der Button zum Wechsel in die Bogendarstellung für die Instanzen der neu erstellten Klasse. Zu sehen ist zusätzlich die Anzeige der Ziffer 0 in den Grenzflächen. Diese repräsentiert die aktuell nicht sichtbaren Annotationen, die Teil einer Relation der Klasse "Koreferenz" sind.

Um nach der Erstellung der neuen Relationenklasse Instanzen dieser auszeichnen zu können, muss zunächst der Modus gewechselt werden. Dazu wird der Button "RL.Mode" in der Menüleiste angeklickt, sodass der Button aktiv wird und orangefarbig aufleuchtet. Es können nun annotierte Personen und annotierte Referenzen verbunden werden, indem diese nacheinander angeklickt werden. Die Relation sind ungerichtet, sodass die Reihenfolge, in welcher die Annotationen angeklickt

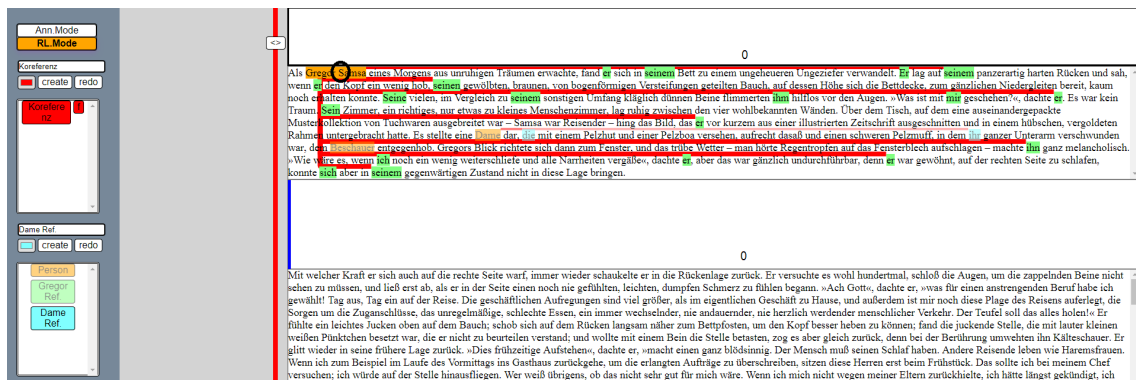


werden, unbedeutend ist. Da es sich bei den ausgezeichneten Instanzen um lokale Relationen handelt, werden sie innerhalb des Textes gezeichnet. Abbildung 6.4 zeigt die Benutzeroberfläche nach der Erstellung der neuen Relationenklasse und nach der Verbindung der Vorkommen von "Gregor Samsa" und "Dame" mit den jeweils abhängigen Referenzen.



**Abbildung 6.4:** Die Benutzeroberfläche nach der Erstellung der Relationenklasse "Koreferenz". In der Menüleiste wurde der Relationenmodus aktiviert, anschließend wurden "Gregor Samsa" und "Dame" jeweils mit ihren Referenten verbunden.

Bei jeder der dargestellten Beziehungen handelt es sich um eine binäre Relation. Dies wird deutlich, wenn die Maus beispielsweise über eine Referenz von "Gregor Samsa" gefahren wird, da dann sowohl Referenz und "Gregor Samsa", als auch die Verbindung zwischen den beiden Annotationen per Highlight hervorgehoben wird. Fährt man mit der Maus über "Gregor Samsa", werden dagegen sämtliche Referenten und die Verbindungen zu diesen visuell herausgestellt. "Gregor Samsa" ist hierbei Teil einer eigenen, binären Relation für jede individuellen Referenz. Wie Abbildung 6.5 verdeutlicht, kann dies genutzt werden, um eine Übersicht über alle Referenzen von "Gregor Samsa" zu gewinnen.

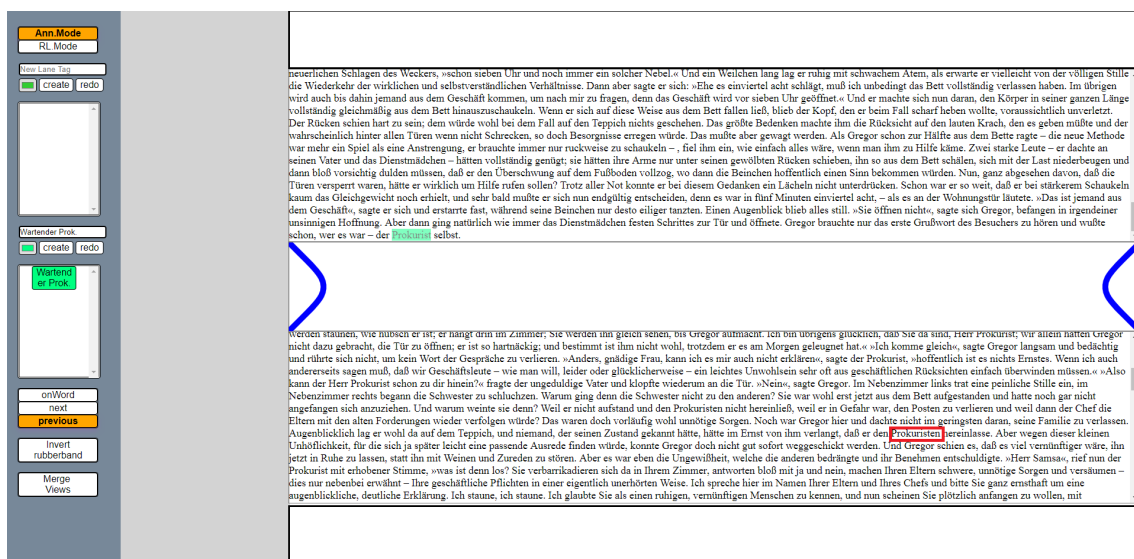


**Abbildung 6.5:** Durch das Führen der Maus über "Gregor Samsa" wird die Opazität der annotierten Referenten und der Verbindungen erhöht, damit diese klar hervorstechen. Dasselbe Ziel verfolgend, wird darüber hinaus die Dicke der Verbindungen etwas erhöht. Der schwarze Kreis dient als Indikator für die Mauszeigerposition.



### 6.1.3 Lange Annotationen

In diesem neuen Szenario soll die Annotation von langen Annotationen vorgestellt werden. Dazu soll der Text nach der ersten Erwähnung des Prokuristen im Text und dessen letzten Erwähnung vor seiner ersten Interaktion mit Gregor Samsa untersucht werden. Die erste sowie letzte Erwähnung gefunden, soll der gesamte Textabschnitt zwischen den Erwähnungen annotiert werden. Es wird dazu zunächst eine neue Annotationsklasse "wartender Prok." erstellt. Anschließend werden die erste und letzte Erwähnung des Prokuristen vor seiner Interaktion mit Gregor gesucht. Da der Abschnitt dazwischen lang ist, wird der Text mithilfe des Trennfeldes aufgeteilt, sodass innerhalb der oberen Ansicht die erste Erwähnung und innerhalb der unteren Ansicht die letzte Erwähnung vor der Interaktion zu sehen ist. Abbildung 6.6 zeigt die Benutzeroberfläche nach dem Split und nachdem die erste Erwähnung innerhalb der oberen Ansicht annotiert wurde.

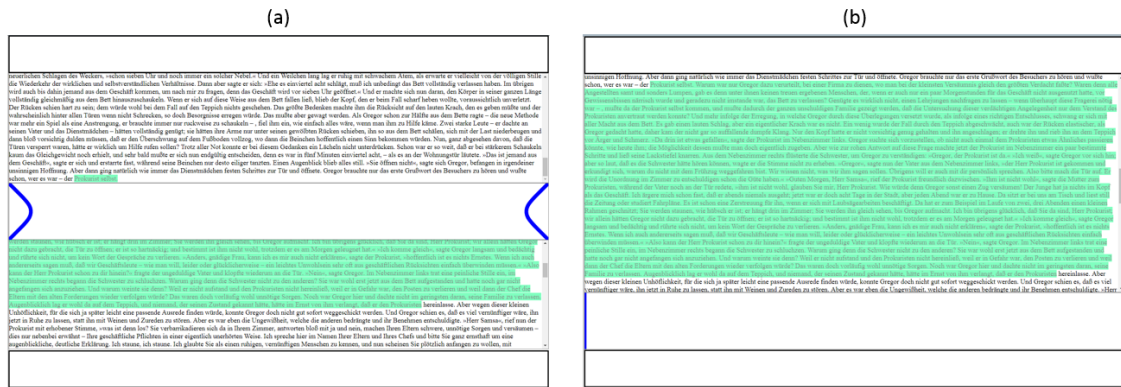


**Abbildung 6.6:** Die letzte Erwähnung innerhalb der unteren Ansicht ist mittels einer roten Umrandung hervorgehoben.

Die erste Erwähnung wurde dabei mittels Doppelklick annotiert. Dies startet den Prozess der Erstellung einer langen Annotation. Wird nun auch die letzte Erwähnung mit einem Doppelklick selektiert, so wird das gesamte Segment zwischen den zwei Erwähnungen annotiert. Abbildung 6.7a zeigt den Text nach Abschluss der der langen Annotation. Die Annotation kann dann zusammenhängend betrachtet werden, indem der Split mithilfe des "Merge Views"-Buttons rückgängig gemacht wird, wie Abbildung 6.7b zeigt.

### 6.1.4 Relationen zwischen entfernten Annotationen

Innerhalb dieses neuen Szenarios soll die Annotation von entfernten Textsegmenten und die Auszeichnung von Relationen zwischen solchen Segmenten verdeutlicht werden. Der Text wird dazu an einer beliebigen Stelle mithilfe des Trennfeldes getrennt, anschließend wird die untere Ansicht nach unten gescrollt. Es ist der Ansatz der Streckung bei sinkender Distanz aktiv, sodass Abbildung 6.8 zeigt, wie die Bänder beim Scrollen der unteren Ansicht und dem damit verbundenem



**Abbildung 6.7:** (a) Der Text nach Abschluss der langen Annotation. Die gesamte Textmenge zwischen den beiden Erwähnungen wurde annotiert. (b) Nach anklicken des "Merge Views"-Buttons werden die beiden Ansichten und ihre Texte wieder zusammengeführt und das Trennfeld auf seine ursprüngliche Position zurück versetzt. Die lange Annotation wird zusammenhängend angezeigt.

Anstieg der Distanz abgerundet werden. Im Trennfeld wird zudem die nicht sichtbare Textmenge kurz vor Beendigung der Scroll Operation angezeigt. Der gesamte Text umfasst etwa 4900 Wörter, der Split des Textes erfolgte etwa in der Mitte. Die Anzeige unterstützt die verhältnismäßig geringe Abrundung der Bänder in ihrer gemeinsamen Aussage, dass nicht weit nach unten gescrollt wurde.

Nach der Erstellung der Annotationsklasse "Person", werden in der Folge in beiden Ansichten Erwähnungen von Personen annotiert. Anschließend werden drei neue Relationenklassen erstellt: Die Klasse "sind verwandt", um verwandte Personen zu verbinden und die beiden Klassen "arbeitet für" und "angestellt bei", welche jeweils unterschiedliche Arbeitsverhältnisse zwischen zwei Personen in Form einer Relation ausdrücken. Anschließend werden die Relationen ausgezeichnet. Zu Beginn werden einige lokale und entfernte Relationen der Klasse "sind verwandt" zwischen verwandten Personen ausgezeichnet. Da die Schwimmliniendarstellung aktiv ist, werden die entfernten Relationen realisiert, indem die betroffenen Annotationen über eine Kante mit der grünen Schwimmlinie verbunden werden. Dass diese Klasse aktiv ist, erkennt man an der im Vergleich zum Rest dicker gezeichneten Schwimmlinie und dem heller leuchtenden Button der Klasse. Abbildung 6.9 zeigt die Benutzeroberfläche nach der Auszeichnung der "sind verwandt" Relationen. Zusätzlich zeigt Abbildung 6.10, wie das Highlight einer Annotation ihre Verbindungen hervorhebt.

Die Auszeichnung der Relationen wird fortgesetzt, indem Annotationen nun auch im Rahmen einer Relation der Klasse "arbeitet für" und "angestellt bei" in Beziehung gesetzt werden. Um die Arbeitsverhältnisse zu unterscheiden, werden Relationen der Klasse "arbeitet für" zwischen Erwähnungen des Dienstmädchens und von Personen aus der Familie Samsa ausgezeichnet. Relationen der Klasse "angestellt bei" werden dann auf Gregor Samsas Beruf beschränkt, sodass sie beispielsweise zwischen Erwähnungen des Chefs, des Prokuristen oder von Gregor ausgezeichnet werden. Abbildung 6.11 zeigt die Benutzeroberfläche nach der Auszeichnung der Relationen der "arbeitet für" und der "angestellt bei" Klassen.



Abbildung 6.8: Der Text wurde aufgeteilt, anschließend wird in der unteren Ansicht nach unten gescrollt. Durch die steigende Distanz werden die Bänder abgerundet.



Abbildung 6.9: Die Relationen der Klasse "verwandt mit" wurden zwischen Erwähnungen von "Gregor", "Vater" und "Eltern" ausgezeichnet.

## 6 Ergebnisse



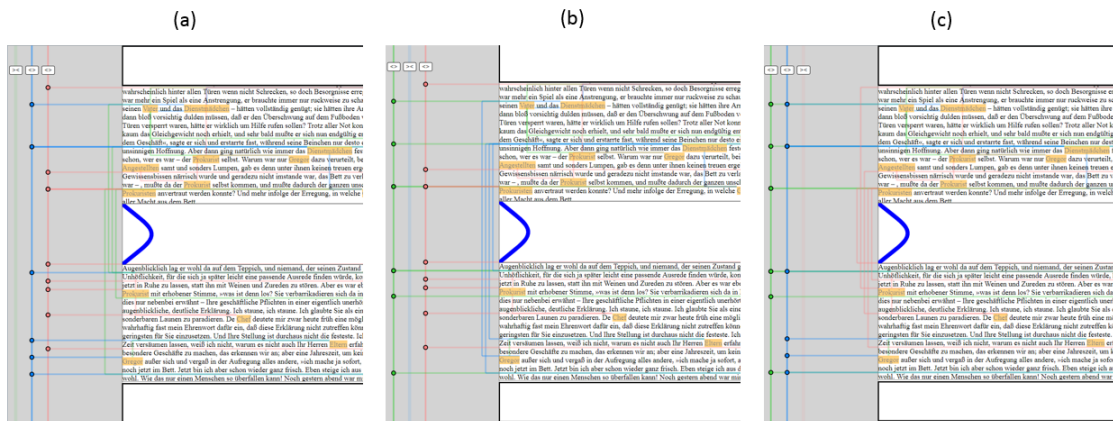
**Abbildung 6.10:** Durch das Fahren der Maus über eine Erwähnung von "Gregor", werden die Erwähnungen von "Eltern", mit denen "Gregor" verbunden ist und die Verbindungen selbst visuell hervorgehoben. Der rote Kreis dient als Indikator für die Mauszeigerposition.



**Abbildung 6.11:** Die Relationen der anderen beiden Klassen wurden nun auch eingezeichnet. Aktuell aktiv ist dabei die Klasse "angestellt bei", was an ihrem Klassen-Button und ihrer Schwimmlinie zu erkennen ist.



Ersichtlich ist nun, dass sich viele der Verbindungen von den Annotationen zur korrespondierenden Schwimmlinie überlagern und dadurch individuelle Verbindungen schwer zu verfolgen sind. Insbesondere ist die Verteilung der Relationen nicht zu erkennen, da sich die Kanten der Schwimmlinien in einzelnen Zeilen überlagern. Um eine bessere Übersicht über die einzelnen Verbindungen zu erhalten, wird die Darstellung der Relationen für die jeweiligen Klassen in die Bogendarstellung gewechselt. Die Abbildungen in 6.12 zeigen die Zustände der Benutzeroberfläche nach jeder einzelnen Ausblendung einer Schwimmlinie. In Abbildung 6.12a wurde die Schwimmlinie der Klasse "verwandt mit" ausgeblendet, sodass die Relationen als Bögen dargestellt werden. Dies verbessert die Verfolgbarkeit einzelner Relationen, außerdem ist nun gut zu erkennen, dass die meisten Relationen dieser Klasse im Bereich unmittelbar über und unter dem Trennfeld ausgezeichnet wurden. Abbildung 6.12b verdeutlicht selbiges für die Klasse "arbeitet für" und Abbildung 6.12c für die Klasse "angestellt bei". Abbildung 6.13 zeigt letztlich, wie die gleichzeitige Ausblendung aller drei Klassen aussieht.



**Abbildung 6.12:** (a) Wechsel in die Bogendarstellung für die Klasse "verwandt mit".  
 (b) Wechsel in die Bogendarstellung für die Klasse "arbeitet für".  
 (c) Wechsel in die Bogendarstellung für die Klasse "angestellt bei".

In jedem der Wechsel werden die Schwimmlinien an den linken Rand der Visualisierungsfläche verschoben, um den Bögen genügend Raum zu bieten.

Die gleichzeitige Ausblendung der Klassen führt jedoch zu vielen Überschneidungen und mindert dadurch die Übersichtlichkeit. Auch wird deutlich, wieso die Schwimliniendarstellung besser mit der Anzahl auszeichnender Relationen skaliert: Die vielen Bögen der verschiedenen Klassen belegen wesentlich mehr Raum, als die drei Schwimmlinien.

### 6.1.5 Filterung von Relationen

Im folgenden Szenario sind die Filterfunktion bezüglich der Relationen vorgestellt werden. Dazu wird das Szenario aus dem Unterkapitel zuvor fortgesetzt und weiter die Relationen der drei Klassen "verwandt mit", "arbeitet für" und "angestellt bei" betrachtet. Im Fokus stehen nun sämtliche Arbeitsverhältnisse, die innerhalb beider Ansichten zu sehen sind. Es sollen dabei die Relationen der Klassen "arbeitet für" und "angestellt bei" isoliert betrachtet werden. Dazu werden sämtliche



Abbildung 6.13: Die gleichzeitige Darstellung aller drei Klassen in der Bogendarstellung.

Relationen der Klasse "verwandt mit" ausgeblendet. Abbildung 6.14 zeigt die Ausblendung der Klasse "verwandt mit" in der Bogendarstellung. Neben der Schwimmlinie und den Verbindungen selbst wird auch die Anzeige zu den nicht sichtbaren Relationen der Klasse temporär entfernt.

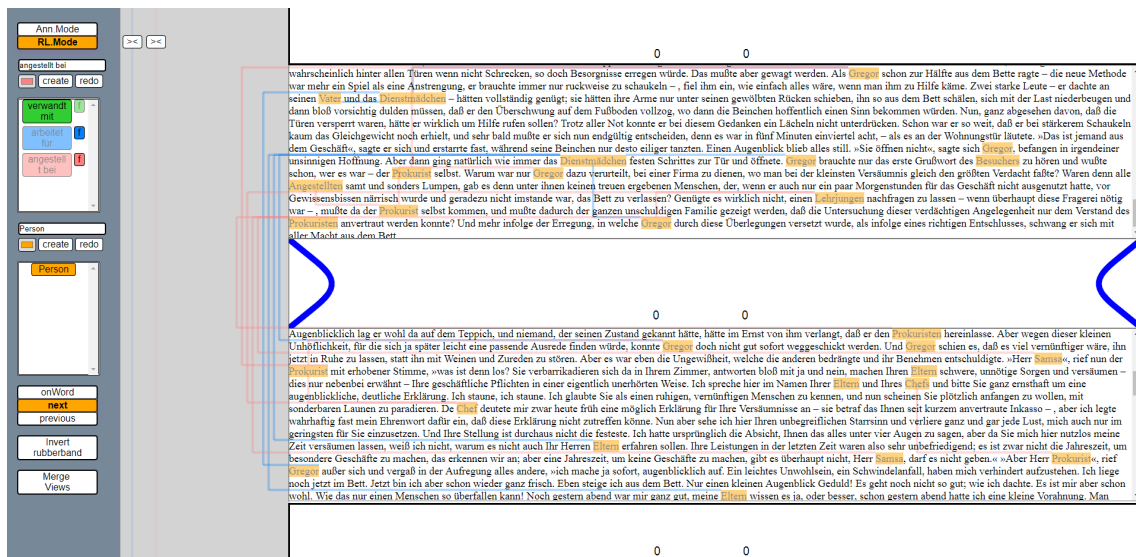


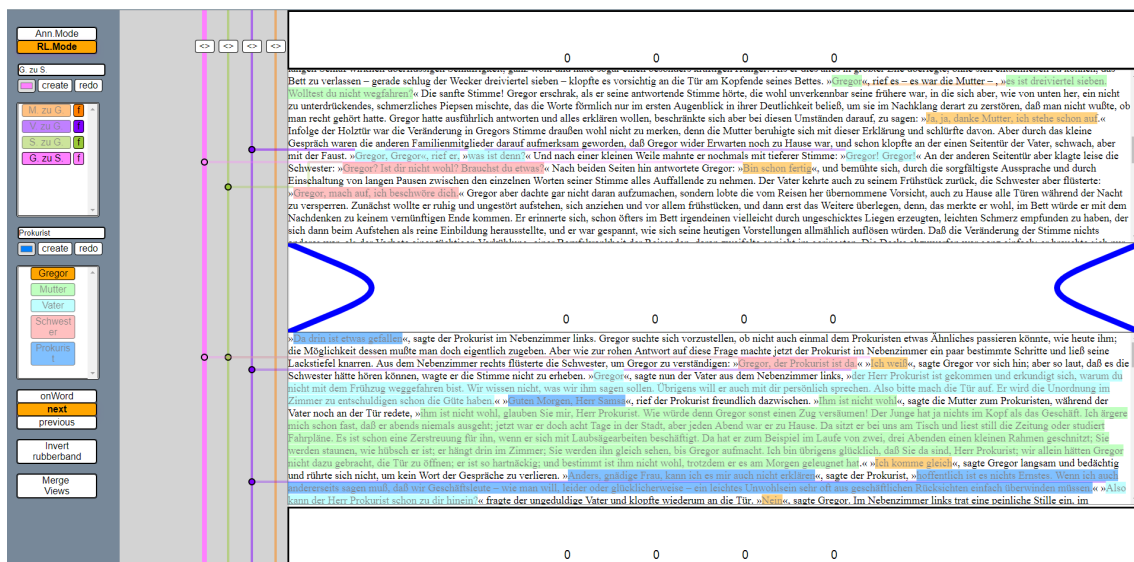
Abbildung 6.14: Die Benutzeroberfläche nach der Ausblendung der Klasse "verwandt mit" mithilfe des Filter-Buttons neben dem Klassen-Button. Die Ausblendung der Klasse wird zusätzlich durch die Verringerung der Opazität des Filter-Buttons signalisiert.

Die Ausblendung verbessert die Übersicht über die verbleibenden Relationen in beiden Darstellungen, vor allem innerhalb des Textes. Besonders wertvoll ist die Filterung insbesondere für die Bogendarstellung, weil so mehr Raum für die zu zeichnenden Bögen verbleibt.

### 6.1.6 Nicht sichtbare Annotationen

Im letzten Szenario soll gezeigt werden, wie die Informationen bezüglich nicht sichtbarer Annotationen im entwickelten Ansatz präsentiert werden.

Der Text wird dazu geteilt, anschließend werden einige Annotations- und Relationenklassen erstellt. Es werden Vorkommen von direkter Rede annotiert, wobei für jede unterschiedliche Person eine separate Annotationsklasse erstellt wird. In diesem Beispiel werden Klassen für "Gregor", "Mutter", "Vater", "Schwester" und "Prokurist" erstellt. Danach werden die Vorkommen von direkter Rede annotiert, wobei die Klasse entsprechend der Person ausgewählt wird, von welcher die direkte Rede abstammt. Im Anschluss werden einige solcher annotierten Vorkommen durch eine Relation verbunden, welche denselben Adressanten und denselben Adressat teilen. Beispielsweise wird die Relationenklasse "M. zu G." für sämtliche annotierte Vorkommen von direkter Rede erstellt, die von der Mutter ausgehen und an Gregor gerichtet sind. Nach demselben Prinzip werden die Klassen "V. zu G.", "S. zu G" und "G. zu S." erstellt. Wie die Benutzeroberfläche nach der Auszeichnung einiger der beschriebenen Relationen aussieht, zeigt Abbildung 6.15.



**Abbildung 6.15:** Die Benutzeroberfläche nach der Erstellung der fünf Annotationsklassen und der vier Relationenklassen.

Nach der Auszeichnung der Relationen in den gezeigten Textabschnitten, wird in der unteren Ansicht nach unten gescrollt, um nach weiteren Abschnitten mit direkter Rede zu suchen. Dadurch verschiebt sich, wie in Abbildung 6.16 zu sehen, der Textabschnitt inklusive seiner Annotationen nach oben. Die resultierende Transition der Annotationen in das Trennfeld initiiert die Anzeige des Balkendiagramms in diesem. Abbildung 6.16 zeigt durch den lila farbigen Balken, dass vier Vorkommen von direkter Rede, die vom Vater ausgehen und an Gregor gerichtet sind, aktuell nicht zu sehen sind und sich im Bereich des Trennfelds befinden. Dasselbe gilt für ein Vorkommen von direkter Rede zwischen der Schwester und Gregor, sowie umgekehrt.

Für das Vorkommen von direkter Rede zwischen der Schwester und Gregor zeigt Abbildung 6.17 zusätzlich, wie das Fahren der Maus über den grünen Balken preis gibt, welche Annotation in der oberen Ansicht mit der nicht sichtbaren Annotation der unteren Ansicht verbunden ist.

## 6 Ergebnisse



Abbildung 6.16: Die Anzeige der Balkendiagramme wird angepasst, da nun nicht sichtbare relationierte Annotation existieren. Entsprechend werden drei verschiedene Balken angezeigt.



Abbildung 6.17: das Highlight hebt die Verbindung und die relationierte Annotation hervor. Der rote Stern dient als Indikator für die Mauszeiger Position.



## 6.2 Expertenbefragung

Im Rahmen einer Expertenbefragung stellten zwei ausgewiesene Experten der maschinellen Sprachverarbeitung respektive Visualisierung ihre Meinung zum entwickelten Ansatz hinsichtlich seines Nützens und seiner Verwendung, den jeweiligen Stärken und Schwächen und möglichen Verbesserungs- und Erweiterungsvorschlägen zur Verfügung. Die ermittelten Ergebnisse der beiden Befragungen werden im Folgenden vorgestellt. Der Experte der maschinellen Sprachverarbeitung wird dabei als Experte A und der Experte der Visualisierung als Experte B bezeichnet.

Beiden Experten wurde der entwickelte Ansatz vorgestellt, indem zunächst eine Übersicht über die Benutzeroberfläche und ihre Komponenten erstellt wurde. Die Funktionalität und Nutzung der einzelnen Komponenten wurde im Anschluss erläutert, ehe die Experten selbst im Rahmen einiger Beispielaufgaben mit dem Ansatz arbeiten sollten. Als Beispieltext wurde ein zufällig generierter Blindtext gewählt.

### 6.2.1 Annotation des Textes

Zu Beginn sollten die Experten eine oder mehrere Annotationsklassen erstellen und den vorliegenden Text innerhalb einer oder mehrerer Ansichten beliebig annotieren. Zur Repräsentation der Annotationen durch die farbliche Hervorhebung wurden keine Kommentare geäußert. Es wurde hauptsächlich der Mechanismus zur Erstellung der Annotationen diskutiert. Beide Experten sahen die Erstellung von Annotationen über das Anklicken der Texttoken kritisch, da dies viel Raum für Fehler bieten würde. Ein entsprechender Mechanismus zur Behebung von Fehlern sei hier essentiell. Die Erstellung von langen Annotationen über den Doppelklick der Ausgangs- und der Zielannotation wurde von Experte A als sinnvolle Funktion erachtet, da dies nach ihm die Arbeit mit langen Texten vereinfachen würde. Die visuelle Markierung des kompletten Textbereichs sei dabei nicht von Nöten, Indikatoren bezüglich Beginn und Ende der Annotation seien hier ausreichend.

Experte A sprach sich zudem für die Integration einer differenzierten Möglichkeit zur Zuweisung mehrerer Klassen von Annotationen an dieselbe Tokenmenge aus. Dies sei hilfreich für die Annotation von Koreferenzen, wo es gegebenenfalls unvorteilhaft sei, einzelne Textsegmente mehrfach annotieren zu müssen.

### 6.2.2 Split des Textes und Distanzvisualisierung

Nach der Trennung des Textes mithilfe des Trennfelds und der Interaktion mit den Textteilen in den verschiedenen Split Ansichten, wurden die Experten zu der Nützlichkeit des Trennfeldes und seiner Bänder befragt. Insbesondere wurde nach der Metapher gefragt, welche sie mit den Bändern und der Formveränderung der Bänder bei der Interaktion mit dem Text assoziieren.

Die Experten erachteten die Funktion der Trennung des Textes durch das Trennfeld als sinnvoll. Nach Experte A sei dies hilfreich, wenn lange Texte in verschiedenen Ansichten annotiert werden sollen. Beide Experten sahen die fixe Höhe des Trennfelds als Problem an und würden die Höhe entweder so gering wie möglich wählen oder die Änderung der Höhe durch den Nutzer erlauben.

Die Darstellung der Distanz, genauer die Kodierung der Menge der nicht sichtbaren Textmenge in der Visualisierung, wurde von beiden Experten positiv bewertet. Dies helfe dabei im Laufe der Bearbeitung des Texts bestimmen zu können, wo man sich im Text befindet. Bezüglich der Metapher interpretierten beide Experten die Streckung der Bänder bei sinkender Distanz als das Zusammenschieben eines Stück Papiers, wobei das Trennfeld und seine Bänder gemeinsam das Papier darstellen. Die im Entwurfskapitel beschriebene Metapher des dünner werdenden Gummibandes sahen sie nicht in der Visualisierung. Generell wurde die Darstellung kritisch bewertet, da die Visualisierung nicht klar verdeutliche, um was es sich bei dem Trennfeld handelt. Experte B fügte hier hinzu, dass die Verfeinerung der Visualisierung, beispielsweise über das Entfernen der Ränder des Trennfelds und der korrekten Schattierung dessen, die Darstellung natürlicher gestaltet werden könnte. Zur Metapher hinter dem Ansatz der Streckung bei steigender Distanz wurden keine Kommentare geäußert, jedoch beschrieb Experte A den zuerst besprochenen Ansatz als intuitiver, da dieser die Zunahme der nicht sichtbaren Textmenge besser darstelle.

Die mögliche Hinzunahme weiterer Trennfelder bewertet Experte A positiv. Diese würden bei der weiteren Aufteilung des Textes helfen, sodass mehrstellige Relationen in den verschiedenen Ansichten bearbeitet werden können. Experte B sah dagegen keine Notwendigkeit für weitere Trennfelder, bedingt durch den Vorschlag einer Übersichtsansicht.

Die Übersichtsansicht, welche den gesamten Text und seine Annotationen und Relationen komprimiert zeigt, wurde von beiden Experten als eine sinnvolle Ergänzung zum Trennfeld vorgeschlagen. Beide Experten sahen in einer Übersichtsansicht eine hilfreiche Komponente, um sowohl eine Übersicht über den aktuell sichtbaren Textabschnitt im gesamten Text, als auch über die Annotationen und Relationen in diesem zu gewinnen. Beide Experten fügten darüber hinaus hinzu, dass die Visualisierung der Distanz über die Bänder mit der Hinzunahme einer Übersichtsansicht nicht mehr von Nöten wäre. Während Experte A die Bänder im Trennfeld durch eine aggregierte Darstellung des nicht sichtbaren Textes ersetzen würde, würde Experte B das Trennfeld auf eine bewegbare Linie reduzieren oder komplett entfernen.

### 6.2.3 Auszeichnung und Darstellung von Relationen

Als nächstes wurden die Experten gebeten verschiedene Relationenklassen zu erstellen und Instanzen dieser in einer oder mehreren Ansichten auszuzeichnen. Danach wurden sie zur ihrer Meinung bezüglich der allgemeinen Darstellung lokaler und entfernter Relationen und hinsichtlich ihrer Präferenz zwischen den Schwimmlinien und den Bögen bei der Darstellung der entfernten Relationen befragt.

Beide Experten äußerten sich positiv gegenüber der unterschiedlichen Darstellung von lokalen und entfernten Relationen. Dies helfe dabei, die unterschiedlichen Relationen auf einen Blick unterscheiden zu können. Ähnlich positiv sahen sie die Vermeidung der Überschneidung des Textes durch die Führung der Verbindungen über den Zeilenzwischenraum in die Visualisierungsfläche. Dieser Aspekt wurde als notwendig empfunden, um die Relationen innerhalb des Textes darzustellen.

Im direkten Vergleich hoben beide Experten die Skalierbarkeit der Darstellung der Relationen in der Schwimmliniendarstellung hervor. Zwar helfe die Verschiebung der Bogen in der Bogendarstellung bei der Vermeidung von Überlagerung, jedoch nutze die Schwimmliniendarstellung den verfügbaren Raum effizienter. Bezüglich der Verfolgbarkeit präferierte Experte A die Bogendarstellung, da die Bögen einfacher nachzuverfolgen seien. Experte B nahm dagegen eine andere Stellung ein und

beschrieb individuelle Relationen in der Schwimmliniendarstellung als einfacher zu verfolgen. Er begründete dies damit, dass der Bogen einer Relation komplett nachverfolgt werden müsse, um die miteinander verbundenen Annotationen ausfindig zu machen. In der Schwimmliniendarstellung reiche es aus, ausgehend von der Schwimmlinie der betroffenen Klasse nach den verbindenden Kanten zu suchen und dann zu den gesuchten Annotationen zu "springen". Dies erlaube, die Schwimmlinie selbst nicht verfolgen zu müssen. In der Summe beurteilten beide Experten jedoch die Interaktion mit den Relationen, genauer das Highlight der Relationen beim Fahren der Maus über diese, als wichtigste Komponente für die Verfolgbarkeit. Die Darstellung der Relationen sei letztlich für die Verfolgbarkeit zweitrangig, solange einzelne Relationen über den Highlight-Mechanismus hervorgehoben werden können. Hinsichtlich der Überlagerung von Relationen präferierten beide Experten die Schwimmliniendarstellung, da diese bei sehr vielen Relationen einen besseren Überblick verschaffe.

Zur Verbesserung der Schwimmliniendarstellung schlug Experte B vor, die Quantität ausgezeichneter Instanzen einer Relationenklasse in die Dicke der Schwimmlinien zu übersetzen, da die jeweiligen Schwimmlinien selbst ansonsten keine relevanten Informationen preisgeben würden. Ein Problem sah er darüber hinaus in der Ansammlung horizontaler Kanten verschiedenfarbiger Relationen in den Zeilenzwischenräumen, da dies zu Falschfarben führen würde. Eine mögliche Lösung hierzu sei die Hinzunahme eines Offsets für konsekutiv gezeichnete Kanten oder die Darstellung der Kanten als gestrichelte Linien. Als sinnvolle Erweiterung sah er zudem die Möglichkeit an, einzelne Wörter aus dem Text herauszuziehen und gegebenenfalls in einem 90 Grad Winkel an die Schwimmlinien hängen zu können. Beispielsweise nach Häufigkeit sortiert, gäbe dies einen hilfreichen Überblick über die annotierten Texttoken.

Die Filterfunktion der Relationen wurde nicht besprochen, da sie zum Zeitpunkt der Befragung noch nicht integriert war.

#### 6.2.4 Darstellung nicht sichtbarer Annotationen und Relationen

Nach der Auszeichnung der Relationen und der anschließenden Interaktion in den Textansichten wurden die Experten zur Darstellung nicht sichtbarer Annotationen und Relationen befragt.

Beide Experten sahen einen großen Nutzen in der Visualisierung der nicht sichtbaren Informationen, beispielsweise die verborgenen Annotationen. Das Konzept der Darstellung nicht sichtbarer, relationierter Annotationen als Balkendiagramm in den Grenzflächen bewerteten beide Experten entsprechend positiv, die genaue Umsetzung jedoch sahen sie kritisch. Die Information, wie viele relationierte Annotationen verborgen sind, sei zwar wertvoll, es sei aber nicht intuitiv ersichtlich, dass die Balken diese Information präsentieren würden. Insbesondere wäre eine zusätzliche Anzeige, wie viele Annotationen im Rahmen einer Relation der betroffenen Klasse verbunden wurden, zum Vergleich hilfreich. Bezüglich der Aggregation der Kanten an den Seiten der Grenzflächen wurde vorgeschlagen, diese weiter in das Trennfeld zu führen, um die Transition der betroffenen Annotationen in die Grenzbereiche noch mehr hervorzuheben. Die Idee der Übersichtsansicht erneut aufgreifend, schlugen beide Experten zudem vor, eine Interaktionsmöglichkeit mit den nicht sichtbaren Annotationen in der Übersichtsansicht zu integrieren. Hier sei es hilfreich, wenn man beispielsweise über das Anklicken dieser Annotationen das betroffene Textsegment in den Sichtbereich bringen kann.

### 6.2.5 Fazit

In der Summe äußerten sich beide Experten positiv gegenüber dem entwickelten Ansatz. Der Ansatz sei hilfreich zur Annotation von langen Texten, besonders durch die vereinfachte Auszeichnung langer Annotationen. Dabei könne man den Mechanismus der Auszeichnung kurzer sowie langer Annotationen noch verbessern. Die Visualisierung der Distanz sei nützlich, um die gescrollte Textmenge jederzeit im Auge zu behalten. Die Visualisierung über die Bänder müsse hierbei angepasst werden, damit sie verständlicher wird. Die Darstellung der Relationen gäbe im Zusammenspiel mit dem Highlight-Mechanismus einen geeigneten Überblick über die ausgezeichneten Relationen. Während die Schwimmlinien-Darstellung dabei gut mit steigender Anzahl an ausgezeichneten Relationen skaliere, biete die Bogendarstellung eine alternative Sicht auf die Relationen, die situativ die Verfolgung individueller Relationen vereinfache. Die Darstellung der Anzahl nicht sichtbarer, relationierter Annotationen sei eine wertvolle Information zur Bearbeitung langer Texte. Hilfreich wäre hier, wenn es noch eine Möglichkeit gäbe, die Gesamtanzahl relationierter Annotationen zu sehen und mit nicht sichtbaren Annotationen interagieren zu können. Besonders die Integration einer Übersichtsansicht wäre hier nach den Experten besonders sinnvoll.

## 6.3 Diskussion und Bewertung

In diesem Kapitel sollen die Designentscheidungen des Entwurfs und die Umsetzung dieser Ideen in der Implementierung diskutiert werden. Angerissen werden dabei künftige Ideen zur Verbesserung und Erweiterung des Ansatzes, die dann im Kapitel Zusammenfassung und Ausblick näher beschrieben werden. Abgeschlossen wird dieses Kapitel, indem die Nützlichkeit des entwickelten Ansatzes zur interaktiven Annotation und Visualisierung relationierter Textsegmente bewertet wird.

Die Möglichkeit zur Trennung des zu bearbeitenden Textes durch die Trennfelder war ein wichtiger erster Schritt zur Bearbeitung langer und gegebenenfalls weit entfernter, relationierter Annotationen. Die Integration der Bänder in den Trennfeldern ermöglicht die effektive Darstellung der Distanz zwischen zwei Textbereichen und damit der nicht sichtbaren Textmenge. Die Expertenbefragung hat verdeutlicht, dass diese Information wertvoll sein kann, wenn lange Texte mit vielen zu annotierenden Textsegmenten in mehreren Ansichten bearbeitet werden müssen. Die konkrete Visualisierung durch die Bänder und deren Formveränderung stellte sich jedoch als nicht aussagekräftig genug heraus, sodass die Metapher hinter den Bändern verloren geht. Ein künftiges Ziel ist damit, die Darstellung der Bänder zu überdenken und zu überarbeiten, sodass sie die Metapher der sich streckenden beziehungsweise dehnenden Gummibänder verdeutlicht. Alternativ könnten die Bänder durch eine geeignetere Darstellung der Distanz ersetzt werden.

Sowohl die Annotation kurzer, als auch langer Annotationen wurde im entwickelten Ansatz umgesetzt. Für die Annotation langer Textsegmente wurde eine effiziente Lösung gefunden, die zudem gut im Zusammenspiel mit der Aufteilung des Textes auf mehrere Ansichten funktioniert, wie das Anwendungsszenario 6.1.3 zeigt und die Expertenbefragung bestätigt. Die freie Kombinierbarkeit von Annotationen und Relationen beliebiger Klassen bietet große Freiheiten bei der Bearbeitung von Texten. Ähnlich der simplen Visualisierung der Annotationen, wurde die freie Kombinierbarkeit unter anderem dadurch begünstigt, dass die Behandlung sich überlappender Annotationsklassen nur

oberflächlich behandelt wurde. Die Expertenbefragung hat zudem gezeigt, dass der konkrete Mechanismus, nämlich das Anklicken der ausgewählten Texttoken, die Auszeichnung mehrerer Klassen an eine einzelne Annotation nicht unterstützt und darüber hinaus fehleranfällig ist. Entsprechend müssen der Mechanismus und die bereits bestehende Funktion, die zuletzt hinzugefügte Annotation entfernen zu können, erweitert werden, damit die Auszeichnung mehrerer Annotationsklassen an ein Textsegment ermöglicht wird.

Bei der Visualisierung der Relationen wurde zunächst beschlossen, unterschiedliche Darstellungen für lokale und entfernte Relationen zu wählen. Wie das Anwendungsszenario 6.1.4 zeigt, stellte sich dieser Schritt als ein wichtiges Element zur Vermeidung der Überlagerung und Überschneidung von Relationen heraus, da lokale und entfernte Relationen so teilweise räumlich getrennt werden. Hierzu verhilft die Entscheidung, entfernte Relationen aus dem Text in die Visualisierungsfläche zu führen, sei es in der Bogen- oder in der Schwimmliniendarstellung. Da die Verbindungen der Relationen zudem durch den Zeilenzwischenraum geführt werden, wird so effektiv die Überschneidung des Textes vermieden. Im nächsten Schritt wurden die möglichen Darstellungen der Relationen bezüglich ihrer Stärken und Schwächen gegenüber gestellt und miteinander verglichen. Aus den Erkenntnissen dieses Vergleichs wurde die Kombination der Schwimmliniendarstellung und der Bogendarstellung abgeleitet. Das Anwendungsszenario zeigt, dass die Entscheidung gerechtfertigt war. Durch die Kombination der Darstellungen wird die Möglichkeit geboten, beim Setzen des Schwerpunkts der Visualisierung dem Bedarf nach zwischen der Skalierbarkeit auszuzeichnender Relationen, der Verfolgbarkeit individueller Relationen und der Übersicht über die ausgezeichneten Relationen zu entscheiden. Die Integration einer Filterfunktion für die Relationen ergänzt den Ansatz und rundet ihn in seiner Aufgabe, viele, verschiedene Relationen auszeichnen und darstellen zu können, ab.

Die verschiedenen Darstellungen bieten aber noch genügend Raum für Verbesserungen. Sowohl die Anwendungsszenarien, als auch die Expertenbefragung haben gezeigt, dass die Interaktion mit den Relationen und das Highlight dieser essentiell für die Entfaltung der beiden Darstellungen ist. Dies ist nicht zwangsläufig ein Nachteil, jedoch können die jeweiligen Darstellung dahingehend verfeinert werden, sodass sie unabhängig von der Interaktion mehr Information präsentieren. Zudem müssen Bestandteile der einzelnen Darstellungen wie die Knoten an den Schwimmlinien überarbeitet werden, da sie keine relevanten Informationen mit sich tragen.

Um Teile der Information von relationierten Annotationen temporär nicht zu verlieren, wenn diese aus dem Sichtbereich verschwinden, wurde der Ansatz um die Grenzflächen und ihre Balkendiagramme ergänzt. Die Balkendiagramme erfüllen ihren Zweck, indem sie die Quantität nicht sichtbarer relationierter Annotationen betroffener Klassen visualisieren. Die Funktion, mit dem Mauszeiger über die Balken fahren zu können und so Teile der verborgenen Relation per Highlight hervorheben zu können, ergänzt die Visualisierung in ihrer Aufgabe, Informationen über die verborgenen Annotationen zu vermitteln. Die Expertenbefragung hat jedoch verdeutlicht, dass der Informationsgehalt der Visualisierung durch die Balkendiagramme gering ist. Zudem skalieren sie nicht gut mit der Anzahl an relationierten Annotationen, da ihre Höhe auf die Höhe der Grenzfläche, in welcher sie sich befinden, beschränkt ist. Die Darstellung der nicht sichtbaren Annotationen innerhalb der Grenzbereiche muss deswegen erweitert werden. Ähnliches gilt für die Aggregation der Kanten, welche die Transition der Annotationen in die Grenzflächen besser hervorheben müssen.

Die Anwendungsszenarien und die Expertenbefragung haben jeweils aufgezeigt, welche Vorteile der entwickelte Ansatz gegenüber den klassischen Annotationswerkzeugen bietet. Es wurde verdeutlicht, dass der entwickelte Ansatz effektiv zur Annotation von Textsegmenten beliebiger Länge in mehreren

Ansichten genutzt werden kann. Zudem wurde gezeigt, dass der Ansatz eine geeignete Lösung zur Darstellung von Relationen in verschiedenen Ansichten und zur Vermeidung von Überlagerungen und Überschneidungen findet.

## 7 Zusammenfassung und Ausblick

Dieses Kapitel resümiert den Entstehungsprozess des in dieser Arbeit entwickelten Ansatzes. Die verschiedenen Entwurfsentscheidungen und die daraus abgeleitete Umsetzung der Komponenten des Ansatzes sollen dazu zusammenfassend präsentiert werden. Abschließend werden dann mögliche Verbesserungen und Erweiterungen des Ansatzes vorgestellt.

### 7.1 Zusammenfassung

Als Ziel dieser Arbeit wurde formuliert, einen Ansatz zu entwickeln, der die interaktive Annotation und Visualisierung relationierter Textsegmente ermöglicht. Im Fokus standen dabei die Annotation langer und gegebenenfalls weit entfernter Textsegmente und die Visualisierung von Relationen zwischen Annotationen mit Blick auf die Herausforderungen, welche die Überschneidung und Überlagerung von Relationen mit sich bringen.

Es wurden zunächst die wichtigsten Grundlagen der Informationsvisualisierung und Textvisualisierung und die Grundlagen der Annotation von Texten zusammenfassend erläutert, um das Fundament des entwickelten Ansatzes besser verstehen zu können. Im Anschluss wurden einige verwandte Arbeiten beleuchtet und ihre Vor- und Nachteile bezüglich der interaktiven Annotation und Visualisierung von relationierten Annotationen untersucht.

Anhand der aus den Grundlagen und den verwandten Arbeiten abgeleiteten Erkenntnisse, wurden dann die wichtigsten Anforderungen an den Ansatz definiert. Die erörterten Anforderungen wurden in der Folge im Entwurf genutzt, um verschiedene Konzepte miteinander zu vergleichen und gegeneinander abzuwägen. Die vielversprechendsten Konzepte wurden dann in der Umsetzung der Komponenten des Ansatzes verwendet.

Die erste Anforderung an den Ansatz war die geeignete Darstellung des zu annotierenden Textes in mehreren Ansichten. Die Anforderung wurde mit der Integration des Trennfelds realisiert, welches die gleichzeitige Betrachtung und Annotation langer und entfernter Textsegmente ermöglicht.

Auf der Aufteilung des Textes aufbauend wurde die Anforderung erhoben, die Distanz zwischen zwei sichtbaren Textbereichen, das heißt die nicht sichtbare Textmenge dazwischen, zu visualisieren. Dazu wurde das Konzept der Bänder im Trennfeld entworfen und umgesetzt, welche die sich verändernde Distanz zwischen zwei Ansichten auf Basis zweier unterschiedlicher Metaphern darstellen.

Die Anforderungen an die Erstellung von Annotationen und Relationen, die Erstellung interaktiv und typunabhängig zu gestalten, wurde umgesetzt, indem zwei verschiedene Modi eingeführt wurden. Die einzelnen Modi erlauben dabei die interaktive Annotation von Textsegmenten beliebiger Länge

und die Verbindung annotierter Textsegmente im Rahmen binärer Relationen. Sowohl Annotationen, als auch Relationen wurden dann in Klassen unterteilt, die beliebig miteinander kombinierbar sind.

Für die Visualisierung der Relationen wurde anschließend gefordert, dass sie informativ und skalierbar ist und eine geeignete Übersicht bietet, selbst wenn sich die Relationen überschneiden oder überlagern. Dazu wurden die verschiedenen Darstellungen von Relationen auf Stärken und Schwächen miteinander verglichen, ehe die Kombination der Darstellung der Relationen als Bögen und der Darstellung mithilfe einer Schwimmlinie umgesetzt wurde. Diese erlaubt den Wechsel zwischen den Darstellungen abhängig davon, ob der Fokus auf der Übersicht, der Skalierbarkeit oder der Verfolgbarkeit von Relationen liegt.

Um die Interaktivität der dargestellten Relationen zu gewährleisten, wurde anschließend gefordert, dass die Visualisierung dynamisch auf Interaktionen des Nutzers reagiert. Ein besonderer Schwerpunkt lag auf der Visualisierung von Informationen, die nicht sichtbare Annotationen enthalten. Realisiert wurde diese Anforderung, indem die Trennfelder um Balkendiagramme ergänzt wurden, welche die Anzahl nicht sichtbarer Annotationen einer Klasse visualisieren und interaktiv die Position der betroffenen Annotationen präsentieren.

Die anhand der Anforderungen entwickelten Komponenten wurden schließlich in der Benutzeroberfläche so angeordnet, dass die Komponenten ihrer Funktionalität entsprechend gekapselt werden.

In mehreren, kurzen Anwendungsszenarien wurde vorgestellt, wie der entwickelte Ansatz zur Lösung bestimmter interaktiver Annotationsaufgaben verwendet werden kann. Ein zusätzlicher Blickwinkel wurde anschließend im Rahmen einer Expertenbefragung geboten, in welcher die Meinungen der Experten bezüglich Stärken und Schwächen des Systems und möglicher Verbesserungen und Erweiterungen zusammenfassend präsentiert wurden. Abschließend wurde dann der entwickelte Ansatz hinsichtlich der gesammelten Ergebnisse diskutiert und bewertet.

## 7.2 Ausblick

In diesem Kapitel sollen einige der in der Expertenbefragung und in der Diskussion angerissenen Verbesserungs- und Erweiterungsmöglichkeiten des entwickelten Ansatzes näher ergründet werden.

Sämtliche Anwendungsszenarien und auch die Expertenbefragung basieren auf einem vorab ausgewählten, gegebenenfalls zufällig generierten Beispieltext. Die Nützlichkeit des Ansatzes ergibt sich somit basierend auf konstruierten Anwendungsbeispielen. Eines der künftigen Ziele ist es, den Ansatz auf Basis realer Annotationsaufgaben zu analysieren und zu bewerten.

Durch die Expertenbefragung inspiriert, wäre es denkbar, den Ansatz um mehrere Trennfelder zu ergänzen, damit der Text in mehr als zwei Ansichten aufgeteilt werden kann. Die damit verbundenen Herausforderungen müssten dafür analysiert werden, insbesondere muss eine geeignete Höhe für jedes einzelne Trennfeld gefunden werden. Denkbar wäre es, die Höhe der Trennfelder variabel zu gestalten, sodass der Nutzer selbst wählen kann, wie hoch sie sind. Dies bringt jedoch den Nachteil mit sich, dass die Darstellung der in den Trennfeldern präsentierten Information mit der Höhe des Trennfelds skalieren müsste.



Während sich die Visualisierung der Distanz im allgemeinen als nützlich erwies, zeigte die Bewertung durch die Experten auf, dass die Darstellung verbessert werden muss. Die visuelle Metapher der sich dehnenen oder streckenden Bänder wurde mehrfach als unintuitiv beschrieben, sodass diese überarbeitet oder gänzlich ersetzt werden muss. Als Beispiel bietet sich die von den Experten vorgeschlagene Metapher eines sich aufrollendes Blatt Papiers an, alternativ könnte man die Metapher gänzlich durch die komprimierte Darstellung des nicht sichtbaren Textauschnitts ersetzen.

Bezüglich seines Datensatzes bietet der Ansatz in seiner aktuellen Form keine Möglichkeit, die erstellen Annotationen und Relationen abzuspeichern und somit die Arbeit an diesen an beliebigen Zeitpunkten fortzusetzen. Der Ansatz muss entsprechend um eine Speicherfunktion ergänzt werden, der die annotierten Textsegmente und ihre Relationen persistent und in einem geeigneten Format abspeichert.

In der Expertenbefragung wurde ersichtlich, dass die Erstellung der Annotationen und Relationen keine ausreichende Fehlertoleranz bietet. Während die zuletzt hinzugefügte Annotation oder Relation gelöscht werden kann, können bereits erstellte Klassen nicht entfernt werden. Die interaktive Erstellung muss so verändert werden, dass sie Veränderungen an den bestehenden Annotations und Relationsklassen erlaubt. Zudem muss die klickbasierte Erstellung von Annotationen und Relationen überdacht werden. Nutzer verwenden die Klickfunktion im Kontext vieler Aktionen, sodass die Wahrscheinlichkeit groß ist, dass der Nutzer versehentlich im Text klickt und dadurch ungewollt eine Annotation oder Relation erstellt. Eine mögliche Alternative wäre die Auslagerung der Erstellung von Annotationen und Relationen jeweils in eigenen Menüs ähnlich den Ansätzen, welche im Kapitel Verwandte Arbeiten vorgestellt wurden.

In Zukunft soll desweiteren die freie Kombinierbarkeit von Annotationen und Relationen verschiedener Klassen ergänzt werden. Im Fokus stehen dabei die Auszeichnung mehrerer Annotationsklassen an dasselbe Textsegment und die Erstellung mehrstelliger Relationen. Bezüglich mehrerer Annotationen muss die Erstellung von Annotationen so ergänzt werden, dass denselben Texttoken verschiedene Klassen zugewiesen werden können, ohne die Token mehrfach annotieren zu müssen. Zusätzlich müssen die verschiedenen Klassen dann geeignet dargestellt werden. Denkbar wäre hier ein Kontextmenü, welches bei der Erstellung einer Annotation geöffnet werden kann. In diesem werden dann die Attribute der Annotation gesetzt. Die Relationen betreffend soll der entwickelte Ansatz künftig um die Erstellung mehrstelliger Relationen ergänzt werden. Diese müssen dazu analysiert und die möglichen Konflikte bei der Erstellung und Darstellung identifiziert werden. Lösungen für die Konflikte könnten dann in einem weiteren, dedizierten Modus nur für die Erstellung von mehrstelligigen Relationen umgesetzt werden.

Die Information, die eine Annotation mit sich trägt, ist auf ihre Texttoken, ihre Klassenzugehörigkeit und die relationierten Annotationen, zu welchen sie in Beziehung steht, beschränkt. Künftig soll der Ansatz um die Möglichkeit ergänzt werden, weitere Metadaten, beispielsweise Kommentare, in eine Annotation zu integrieren. Eine ähnliche Ergänzung bietet sich auch für Relationen an, welche künftig als annotierbare Entität realisiert werden sollen.

Die Visualisierung von Relationen muss verbessert werden, um weitere Konflikte, die bei der Überschneidung und Überlagerung von Relationen auftreten, zu lösen. Der kombinierte Ansatz zur Darstellung von Relationen ermöglicht das Setzen verschiedener Schwerpunkte, jedoch überschneiden sich einige dieser Schwerpunkte, wie die Expertenbefragung gezeigt hat. Entsprechend soll das Zusammenspiel der Bogen und der Schwimmlinien-Darstellung verfeinert werden. Ansätze

hierzu bieten sich beispielsweise in der Darstellung sich überlagernder Kanten in den Zeilenzwischenräumen an. Hier ließe sich ein Offset für konsekutiv gezeichnete Relationen einführen. Alternativ könnte die Visualisierung horizontal verlaufender Kanten angepasst werden, indem diese als gestrichelte Linien umgesetzt werden, wie in der Expertenbefragung vorgeschlagen.

Zu Überdenken ist auch die Visualisierung der Information von nicht sichtbaren Annotationen. In der jetzigen Form des Ansatzes werden außer der Quantität von Relationen einer Klasse und der Position betroffener Annotationen keine weiteren Informationen visuell festgehalten. Wie in der Diskussion beschrieben, skalieren die Balken der Diagramme zudem schlecht mit steigender Anzahl an relationierten Annotationen. Die Balkendiagramme könnten künftig entfernt werden, um die Darstellung der Informationen in eine Übersichtsansicht auszulagern. Diese könnte innerhalb der Benutzeroberfläche an einen der beiden Ränder oder in einem eigenen, zu öffnenden Fenster präsentiert werden. Innerhalb der Übersichtsansicht kann der Text dann inklusive seiner Annotationen und Relationen komprimiert dargestellt werden, sodass er eine globale Sicht auf den gesamten Text bietet. Es ließen sich in der Folge interaktive Funktionen integrieren, die das Abfragen bestimmter Informationen zuließen. Als Beispiel wäre das Fahren der Maus über eine komprimierte Annotation, was zur Anzeige bestimmter Metadaten führt, zu erwähnen. Werden zusätzlich die lokale Textansicht und die Übersichtsansicht synchronisiert, kann es dann dem Nutzer ermöglicht werden, durch das Anklicken einer Annotation in der Übersichtsansicht, innerhalb der lokalen Ansicht diese Annotation anzeigen zu lassen.

## Literaturverzeichnis

- [BM17] A. Breitenfeld, C. Müller-Birn. *A State-of-the-Art Review of Semantic Annotation Tools*. Techn. Ber. Freie Universität Berlin, Fachbereich Mathematik und Informatik, 2017. URL: <https://refubium.fu-berlin.de/handle/fub188/21870> (zitiert auf S. 6, 7).
- [Bur14] M. Burghardt. „Engineering annotation usability - Toward usability patterns for linguistic annotation tools“. Diss. Fakultät für Sprach-, Literatur- und Kulturwissenschaften der Universität Regensburg, 2014. URL: <https://epub.uni-regensburg.de/30768/> (zitiert auf S. 6).
- [Bur18] M. Burghardt. „Visualization as a Key Factor for the Usability of Linguistic Annotation Tools“. In: *Visualisierung sprachlicher Daten*. Hrsg. von N. Bubenhofer, M. Kupietz. Heidelberg: Heidelberg University Publishing, 2018, S. 315 (zitiert auf S. 7, 8).
- [CC16] N. Cao, W. Cui. *Introduction to Text Visualization*. Atlantis Briefs in Artificial Intelligence, 2016. DOI: [10.2991/978-94-6239-186-4](https://doi.org/10.2991/978-94-6239-186-4) (zitiert auf S. 5).
- [Che10] C. Chen. „Information visualization“. In: *Wiley Interdisciplinary Reviews: Computational Statistics* 2.4 (2010), S. 387–403 (zitiert auf S. 3).
- [LCWL14] S. Liu, W. Cui, Y. Wu, M. Liu. „A survey on information visualization: recent advances and challenges“. In: *The Visual Computer* (2014), S. 1373–1393. DOI: [10.1007/s00371-013-0892-3](https://doi.org/10.1007/s00371-013-0892-3) (zitiert auf S. 3, 4).
- [LPV12] F. Landragin, T. Poibeau, B. Victorri. „ANALEC: A New Tool for the Dynamic Annotation of Textual Data“. In: *International Conference on Language Resources and Evaluation (LREC 2012)*. Hrsg. von E. L. R. Association. Istanbul, Turkey, 2012, S. 357–362. URL: <https://halshs.archives-ouvertes.fr/halshs-00698971> (zitiert auf S. 11, 13).
- [MH00] T. Munzner, P. Hanrahan. „Interactive visualization of large graphs and networks“. Diss. Department of Computer Science, Stanford University, 2000 (zitiert auf S. 3, 4).
- [NPW14] J. Nualart, M. Pérez-Montoro Gutiérrez, M. Whitelan. „How we draw texts: a review of approaches to text visualization and exploration“. In: *El Profesional de la Información*, 2014, vol. 23, num. 3, p. 221–235 (2014). URL: <http://dx.doi.org/10.3145/epi.2014.may.02> (zitiert auf S. 5).
- [SPT+12] P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou, J. Tsujii. „BRAT: a web-based tool for NLP-assisted text annotation“. In: *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics. 2012, S. 102–107. URL: <http://dl.acm.org/citation.cfm?id=2380921.2380942> (zitiert auf S. 15).

- [SW08] Y. B. Shrinivasan, J. J. van Wijk. „Supporting the Analytical Reasoning Process in Information Visualization“. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM, 2008, S. 1237–1246. URL: <http://doi.acm.org/10.1145/1357054.1357247> (zitiert auf S. 4).
- [WM12] A. Widlöcher, Y. Mathet. „The Glozz Platform: A Corpus Annotation and Mining Tool“. In: *Proceedings of the 2012 ACM Symposium on Document Engineering*. New York, NY, USA: ACM, 2012, S. 171–180. URL: <http://doi.acm.org/10.1145/2361354.2361394> (zitiert auf S. 13).

Alle URLs wurden zuletzt am 07. 11. 2018 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