

Interactive Visual Analysis of Translations

Mohammad Saqar Alharbi
508205

Submitted to Swansea University in fulfilment
of the requirements for the Degree of Doctor of Philosophy



Swansea University
Prifysgol Abertawe

Department of Computer Science
Swansea University

October 25, 2021

Declaration

This work has not been previously accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed  (candidate)

Date 25/10/2021

Statement 1

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

Signed  (candidate)

Date 25/10/2021

Statement 2

I hereby give my consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed  (candidate)

Date 25/10/2021

*I would like to dedicate this work to the soul of my father;
to my mother for her encouragement and continuous prayers, and
to my wife for her endless patience and support.*

Abstract

This thesis is the result of a collaboration with the College of Arts and Humanities at Swansea University. The goal of this collaboration is to design novel visualization techniques to enable digital humanities scholars to explore and analyze parallel translations. To this end, chapter 2 introduces the first survey of surveys on text visualization which reviews all of the surveys and state-of-the-art reports on text visualization techniques, classifies them, provides recommendations, and discusses reported challenges.

Following this, we present three visual interactive designs that support the typical digital humanities scholars workflow. In Chapter 4 we present VNLP, a visual, interactive design that enables users to explicitly observe the NLP pipeline processes and update the parameters at each processing stage. Chapter 5 presents AlignVis, a visual tool that provides a semi-automatic alignment framework to build a correspondence between multiple translations. It presents the results of using text similarity measurements and enables the user to create, verify, and edit alignments using a novel visual interface. Chapter 6 introduce TransVis, a novel visual design that supports comparison of multiple parallel translations. It incorporates customized mechanisms for rapid and interactive filtering and selection of a large number of German translations of Shakespeare's Othello. All of the visual designs are evaluated using examples, detailed observations, case studies, and/or domain expert feedback from a specialist in modern and contemporary German literature and culture.

Chapter 7 reports our collaborative experience and proposes a methodological workflow to guide such interdisciplinary research projects. This chapter also includes a summary of outcomes and lessons learned from our collaboration with the domain expert. Finally, Chapter 8 presents a summary of the thesis and future work directions.

Acknowledgements

First and foremost I would like to thank my supervisor Dr. Robert S. Laramée for his support and guidance throughout my Ph.D. and my Masters projects before that. We had many interesting discussions over this period covering a large array of topics. I would also like to thank our collaborator Professor Tom Cheesman for his advice, guidance, and for conceiving the project. I also acknowledge and appreciate the support and assistance of Dr. Matthew Roach when my first supervisor moved to Nottingham University.

I gratefully acknowledge funding from my sponsors, the Technical and Vocational Training Corporation (TVTC) and the Ministry of Education of Saudi Arabia. Without their financial support and funding, it would never have been possible to commence the project.

I would like to express my gratitude to my mother for her support and encouragement throughout my life. Thanks also goes to my wife and children for standing beside me and for their endless patience and support. Finally, I would like to thank my extended family, friends, and fellow Ph.D. students for their support, assistance, and needed distractions.



Contributions

- Mohammed Alharbi, Robert S Laramee, and Tom Cheesman, **AlignVis: Semi-Automotic Alignment and Visualization of Parallel Translations**, The 24th International Conference on Information Visualization, (IV2020), 7-11 Sept 2020, Vienna, Austria, DOI: [10.1109/IV51561.2020.00026](https://doi.org/10.1109/IV51561.2020.00026) [1].
- Mohammed Alharbi, Robert S Laramee, and Tom Cheesman, **TransVis: Integrated Distant and Close Reading of Othello Translations**, IEEE Transactions on Visualization and Computer Graphics, DOI: [10.1109/TVCG.2020.3012778](https://doi.org/10.1109/TVCG.2020.3012778) [2].
- Mohammed Alharbi and Robert S Laramee, **SoS TextVis: An Extended Survey of Surveys on Text Visualization**, Computers, Volume 8, Number 1, (February) 2019, pages 1-20, DOI: [10.3390/computers8010017](https://doi.org/10.3390/computers8010017) [3].
- Mohammad Alharbi and Robert S Laramee, **SoS TextVis: A Survey of Surveys on Text Visualization**, The Computer Graphics and Visual Computing (CGVC) Conference 2018, 12-14 September 2018, Swansea, UK, DOI: [10.2312/cgvc.20181219](https://doi.org/10.2312/cgvc.20181219) [4].

Co-author:

- Naif Alharbi, Mohammad Alharbi, Xavier Martinez, Michael Krone, Alexander Rose, Marc Baaden, Robert S. Laramee, Matthieu Chavent, **Molecular Visualization of Computational Biology Data: A Survey of Surveys**, EuroVis Short Papers 2017 , pages 133-137, DOI: [10.2312/eurovisshort.20171146](https://doi.org/10.2312/eurovisshort.20171146), [5]

Contents

List of Tables	x
List of Figures	xi
1 Introduction and Motivation	1
1.1 Data Visualization	1
1.2 Information Visualization	3
1.3 Visualizing Variation in Translations of Shakespeare’s play <i>Othello</i>	9
1.3.1 Experience of Collaboration	9
1.3.2 Description of Parallel Translation Data	15
1.3.3 Domain Problem and Research Objectives	16
1.4 Thesis Overview	18
1.5 Thesis Evaluation	19
1.6 Video Demonstrations	20
2 SoS TextVis: A Survey of Surveys on Text Visualization	21
2.1 Introduction and Motivation	22
2.1.1 Literature Search Methodology	24
2.1.2 Survey Scope	25
2.1.3 Related Work	25
2.1.4 Survey Classification	25
2.2 Summary and Comparison of Surveys	26
2.2.1 Document-Centered Surveys	26
2.2.2 User Task Analysis Surveys	28
2.2.3 Multi-Faceted Text Visualization Surveys	29
2.2.4 Cross-Disciplinary Text Visualization Surveys	33

2.2.5	Satellite-Themed Text Visualization Surveys	35
2.2.6	Survey Recommendations	36
2.3	Discussion of Future Challenges	38
2.4	Chapter Summary	43
3	Background	45
3.1	Related Work of Visible NLP	45
3.2	Related Work of AlignVis	49
3.3	Related Work of TransVis	51
3.4	Chapter Summary	57
4	VNLP: Visible Natural Language Processing	59
4.1	Introduction and Motivation	60
4.2	Definitions and Terminology	61
4.3	Requirement Analysis	64
4.4	Implementation and Design of Visible NLP pipeline	64
4.4.1	Visible Segmentation and Tokenization	67
4.4.2	Visible Stopwords	69
4.4.3	Visible Normalization	71
4.4.4	Visible Embeddings	73
4.5	Evaluation	73
4.5.1	Visible Text Similarity Application	73
4.5.2	Domain Expert Feedback	77
4.6	Chapter Summary	79
5	AlignVis: Semi-automatic Alignment and Visualization of Parallel Translations	81
5.1	Introduction and Motivation	82
5.2	Requirement Analysis	83
5.3	Definitions and Terminology	84
5.3.1	Alignment Preprocessing	85
5.4	Design of AlignVis	85
5.4.1	AlignVis Overview	85
5.4.2	Semi-automatic Alignment Exploration and Verification	88
5.4.3	Domain Expert Refinement	90

5.4.4	Selection and Filtering	92
5.5	Evaluation	93
5.5.1	Domain Expert Feedback	93
5.5.2	Comparison With the Computational and Visual Alignment Tools	95
5.5.3	Comparison With a Standard Alignment Tool	96
5.6	Chapter Summary	98
6	TransVis: Integrated Distant and Close Reading of Othello Translations	101
6.1	Introduction and Motivation	102
6.2	Definitions and Background	103
6.3	Design Requirements and Tasks	105
6.4	TransVis's Design	107
6.4.1	Visual Design Factors	110
6.4.1.1	Filtering based on Derived Alignments	110
6.4.1.2	Detailed View For Close Reading	111
6.4.1.3	Term-Level Comparisons (TLC) View	112
6.4.2	Interaction Design Factors	114
6.4.2.1	Smooth Zooming of Translations	114
6.4.2.2	Filtering, Selection, and Positioning in the Alignment	
	Overview	116
6.4.2.3	Filtering and Selection of Speeches and Translations in Op-	
	tions Panel	117
6.5	Evaluation	119
6.5.1	Domain Expert Feedback	120
6.5.2	Case Studies Using the Design and the Integrated Similarity metrics	123
6.6	Chapter Summary	129
7	Collaborating with Digital Humanities: A Methodology	131
7.1	Introduction and Motivation	132
7.2	Background	132
7.3	Collaborative Workflow	133
7.3.1	Three Spaces and Three Channels	133
7.3.2	Quality Criteria	137
7.4	Collaboration Outcome and Reflection	139

7.5 Chapter Summary	141
8 Conclusions	143
8.1 Outcomes	143
8.2 Future Work	145
Bibliography	149
Appendices	169

List of Tables

1.1	A list of the collaborative and domain expert feedback sessions related to this thesis. . . .	13
1.2	The video demonstrations of our design studies.	20
2.1	A list of literature sources searched for text visualization surveys. We mainly use IEEE Xplore [6], the ACM Digital Library [7], and Google Scholar [8] to search for literature .	24
2.2	Classification of our collection of 14 text visualization surveys. There are five categories: Data source, task analysis, multi-faceted, cross-disciplinary, and satellite-themed into which the literature is grouped.	26
2.3	Summary of the future challenges reported in our collection of surveys. This list contains the common challenges among the text visualization surveys.	41
3.1	A summary of related work. Each paper is characterized by a two-level hierarchy [9]: the derived data that each approach generates and represents and the supported visual encodings in the NLP pipeline. Our approach makes the entire NLP pipeline visible.	46
3.2	A summary table of related work and paper characteristics included in the computational and visual alignment section. The dashes (-) in the distant reading column indicate that the corresponding reference does not feature distant reading.	49
3.3	Summary of visual design characteristics for related work discussed in the sections: 3.3, 3.3 and 3.3. The “-” sign in the table indicates that an approach does not provide a distant or a close reading view. “Arbitrary” means the number is open based on the author’s claims and “Not-specified” means the number of parallel documents is not mentioned in the paper and is not exemplified. The references are ordered based on publication date.	51
5.1	A comparison between AlignVis and the related work.	95
7.1	Example representatives of the results of the implementation stages in the solution space that correspond to our contributions, TransVis [2], AlignVis [1], and VNLP [10].	136
7.2	A table of the encountered pitfalls identified by Sedlmair et al. [11] and their relevance to this project.	141

List of Figures

1.1	(a) The original Anscombe’s quartet dataset. (b) Four scatter plots that correspond to the datasets [12].	2
1.2	A visualization by Minard (1781-1870) illustrates the result of Napoleon’s failed campaign of 1812. Image courtesy of Tufte [13].	3
1.3	An image by Playfair (1759-1823) depicts the balance of trade. Image courtesy of Playfair [14].	4
1.4	Snow’s (1813 – 1858) map of deaths from a cholera outbreak in London in 1854. Image courtesy of Snow [15].	5
1.5	An example of a parallel coordinates plot. Image courtesy of Roberts et al [16].	6
1.6	An example of a treemap plot which represents the change in the U.S. stock values. Image courtesy of Laubheimer [17].	7
1.7	An example of a node-link diagram that represents a small protein interaction network. Image courtesy of Michailidis [18].	7
1.8	An overview of the different brushing approaches that facilitate selection and interaction. Image courtesy of Roberts et al [16].	8
1.9	The integration between data analysis, visualization, and human in the visual analytics process. Image courtesy of Cui [19].	8
1.10	Timeline of key events and outputs from the collaborative work between the school of arts and humanities and the visualization group at Swansea University (2011-2016). This figure illustrates the time prior to the beginning of this thesis.	10
1.11	Timeline of the most key events and contribution of the collaborative work between the school of arts and humanities and the visualization group at Swansea University (2017-2020). This figure illustrates the collaborative events and publications that coincide with our thesis.	11
1.12	The history of the submissions of our paper TransVis [2].	14
1.13	Screenshot of the XML file of the dataset [20].	15
1.14	The domain expert workflow. The first two stages: data collection and curation are performed by the domain expert. Our thesis contributes to the later three stages: data preprocessing (Chapter 4), segment alignment (Chapter 5), and analysis (Chapter 6)	17

2.1	Text visualization surveys from 2010 to 2018. Blue bars indicate the number of methods reviewed in each survey. Orange bars show the number of citations each survey attracts. In term of the number of surveys, 2014 dominates with four. However, with respect to the number of techniques, surveys from 2015 collectively review 480 methods.	23
2.2	Text visualization methods presented by Šilić and Bašić [21]. The table summarizes the methods, their underlying algorithms, the publication year, whether the method includes a temporal presentation or not, and the data type on which the method operates (C: Collection of text, S: Single text, SI: Short intervals). In the ‘Temporal’ column, if the method conveys time, it has (+), (-) if it does not, and (N/A) if not applicable. Reproduced with permission from the author, Knowledge-based and intelligent information and engineering systems; published by Springer, 2010.	30
2.3	The classification of text visualization techniques used in the survey by Kucher and Kerren [22]. Reproduced with permission from the author, IEEE Pacific Visualization Symposium; published by IEEE, 2015.	32
2.4	A snapshot of the recent web-based visualization tool developed by Liu et al. [23]. The visualization and text mining techniques are connected by their shared analysis tasks. On the right, a list of the corresponding literature that matches user preference. The red and blue colors correspond to the fields of visualization and data mining respectively.	32
2.5	Hierarchical classification of research papers reviewed by Jänicke et al. [24]. At the top-right, the intended tasks supported by the visual design and the techniques implemented. On the left, the rows show the paper classification organization. Reproduced with permission from the author, Eurographics Conference on Visualization (EuroVis)-STARs; published by The Eurographics Association, 2015.	34
2.6	A bi-gram word cloud representation of the surveys by Wanner et al. [25], Kucher and Kerren [22], Jänicke et al. [26], and Liu et al. [23] to illustrate the vocabulary used in each one. We apply the word clouds using the script by Müller [27].	37
2.7	A list of the most common bi-grams extracted from the collective surveys. The color mapping of the cells indicates the weights of the corresponding bi-gram.	39
2.8	A snapshot of the web-based parallel coordinates plot developed to explore the vocabulary of the surveys interactively. Each vertical line (coordinate) represents a survey, except The first coordinate on the left corresponds to the occurrence frequency. Each polyline represents a word and the intersection between the polylines and the vertical coordinates depicts the word weights in that survey. Here, the user selects the most common vocabulary between the two surveys from the cross-disciplinary group [28]	40

2.9	Visualization of the number of approaches reviewed by the surveys. Each row represents a survey and each cell represents the number of references in the corresponding year (columns). The color mapping of the cells indicates the number of approaches cited within each survey in the corresponding year. The last column shows the total methods each survey includes.	41
4.1	Our VNLP pipeline illustrates five main NLP stages: text segmentation, tokenization, stop word removal, normalization, and embeddings. We integrate an application to text similarity quantification to demonstrate the usefulness and advantages of our approach. In the lower half, we show the corresponding visual encodings of the VNLP pipeline stages. . .	62
4.2	An overview of the VNLP pipeline. (A) A dialogue that accommodates options to customize the text, update the font size for accessibility, and the color options. (B) The VNLP pipeline GUI where the user controls the parameters of each stage. (C) The user-chosen focus text. (E) The context of the user-selected segment. (D) The visible results pipeline with each result reflecting the parameters chosen in the corresponding GUI component. .	65
4.3	An information dialog view that shows a detailed explanation of the corresponding VNLP pipeline stage.	66
4.4	Top: the visible tokenization and segmentation GUI components. Bottom: the visible tokenization and segmentation results. In each result, metadata analysis is provided for an overview of the results.	67
4.5	Two examples of ambiguous segmentation cases. (a) A period placement in a quote (annotated by arrows) results in a new segment. (b) The second segment was generated due to the period in the word “St. Romain”.	69
4.6	Two examples of erroneous tokenization cases. (a) A inaccurate tokenization of the word “o’clock”. (b) The compound word “well-to-do” is divided into three tokens which are considered stopwords and consequently removed in the following NLP stage.	70
4.7	(a) in the stopword GUI window, the user can include or remove stopwords. The GUI provides a list of the stopwords where the user can add or remove them. (b) From the visible results of both the stopwords and normalization, the user can observe stopwords and add words from the normalization results to the stopwords list and vice versa.	70
4.8	Two cases of visible stopwords. The left shows a case where the entire segment is composed of stopwords. The right shows a segment with five sequential stopwords. The glyphs that accompany each segment illustrate the tokenization delimiters chosen by the user. . .	71
4.9	An example of stopword exploration. (a) In the visible stopword window, the user can explore the stopwords included in the selected segment. (b) In the visible normalization result the user can identify candidate words and add them to the stopword list.	72

4.10	An example of the exploration of user modifications to the visible embedding generation process. (a) The default embedding generation implementation results in common words such as, “one” and “cap” to become distinctive words. (b) After the user changes the TF calculation method, the words “one” and “cap” are considered non-distinctive and other more important words appear.	74
4.11	The two views that are shown when the user hovers over a word in the visible embedding result. (a) A summary of the embeddings values that are derived for this word. (b) A breakdown of the formula that is used to derive the current embeddings.	75
4.12	Three similarity histograms that show the similarity results along the x-axis. The y-axis indicates the number of results for each similarity value. The histogram in (a) indicates that the distance between the first value and the second value along the x-axis is small while it is relatively greater in the other histogram (b). The histogram in (c) shows the effect of showing the similarity values that equate to zero.	76
4.13	The embeddings map window that illustrates the embeddings values in both the focus and target texts. It integrates user options where the user sets the sorting of the embeddings values and navigates to other target segments. The two bar charts represent the embeddings in the focus and target text.	77
4.14	Two examples of the embeddings map which demonstrates the effect of the user interaction in the VNLP GUI. The map in (a) shows the default settings for both the tokenization and normalization. The map in (b) shows the reduction of features after applying more delimiters to the tokenization and stemming.	78
5.1	An overview of AlignVis. The alignment editor canvas (1) illustrates the original English text (T_E), the base German translation (T_B), and the focus German translation (T_F). This view shows the machine-recommended alignments between the German translations and enables manual refinement. The column indicated by a green diamond glyph shows the secondary measurement feature. The post-edit area (2) shows the processed translations; the user can explore the content and move items back to the editor canvas. The user options panel (3) provides filtering and interaction options and includes the translation thumbnail overview (4). The latter view shows the translations in chronological order of publication and enables the user to add translations to the editor canvas. The user options panel provides options that enable the user to interact with the design and change properties such as similarity measurements, filters, and color schemes.	86
5.2	(a) On the left, the alignments without applying the confidence value threshold. On the right, the effect of applying a threshold ($\kappa = 0.75$). Some of the diagonal edges are removed. (b) A screenshot of the deletion action when the user selects multiple alignment edges. . .	88

5.3	A screenshot of the top three candidate segments for a user-chosen T_B segment (RE). The candidate segment edges are rendered in green and the saturation of the color represents the rank of the candidate segments. The order of the segments is based on the ranking of each segment.	90
5.4	The alignment update mode: when the user chooses to update an existing alignment (RE). The original alignment is represented using a dashed gray edge, and a dynamic blue guide edge is rendered to indicate the new alignment.	91
6.1	The alignment overview (A) shows the parallel alignment of translations with the original base text. The highlighted path in (A) shows the distant alignments of a segment of the "Othello" speech starting with "I ran it through...". In the left-bottom, a zoomed-in view magnifies the curved edges. Window (B) shows the options panel that facilitate exploration of the collection and comparison of translations. View (C) is a close reading view (the detailed view) that corresponds to a user-selected speech and each aligned speech. Window (D) shows the Term-Level Comparison (TLC) view. The zoomed-in rectangle is part of the figure, not of the visualization itself.	107
6.2	The detailed view shows the user-selected speech highlighted using a red border and the aligned speeches ordered consistent with the translations appearing in the alignment overview. Each speech is paired with with a colored bar (annotated using a green border) to indicate the similarity distance. In the bottom-left corner, a list of all previously selected speeches. If the user selects any speech from the list, the corresponding text is highlighted in the alignment overview and the edges of alignments are presented above.	111
6.3	An example of aligned segments of the base English segment starting with "Thou art sure of me..." depicted by term-weighting matrices using the TLC view. In this example, the user brushes the three peaks that reflect the distinctive translations of the word "Moor" in the context: "I hate the Moor". The highlighted translations are "Neger" in Buhss (1996) [29], "Schwarze" in Günther (1992) [30], and "Maure" in Schwarz (1941) [31]. The terms list reflects the brushing result and assigns the same colors to the terms. The user-chosen terms are rendered in the focus while the rest are rendered as context.	112
6.4	(a) An example of two corresponding sub-sets of six editions of Baudissin (1832) [32]. The variation of colors in the dashed rectangle in the original version (a) is clearer than the corresponding rectangles in the lemmatized version (b).	113
6.5	An example of three levels of integrated zooming and the smooth changes to the level of detail. (left) Distant reading without zooming. (middle and right) close reading after the user zooms in. The textual content of the speeches fades in smoothly at an increasing level of detail. The grey curved path indicates a user-selected alignment.	114

6.6	Two snapshots of the same region of the collection. In (a) the segments aligned with the user-selected segment are out of view and not visible. In (b) the segments are horizontally aligned with the user-selected segment.	116
6.7	The placement of the unstable translation by Bärffuss (2001) [33] results in multiple disconnected edges between non-adjacent translation. The differently colored edges illustrate the alignment between the two non-adjacent translations.	117
6.8	A subset of a filtered focus+context rendering of alignment overview. The view is filtered by the speaker “ <i>Duke of Venice</i> ”. The high number of matches is due to mistaken correspondence during the original segment alignment process which is easily discovered by the visualization result. In this figure, a segment of speech by “ <i>Othello</i> ” was mistakenly aligned with a segment of speech by “ <i>Duke of Venice</i> ”.	118
6.9	The alignment overview of the translation collection reveals an increase of variation between translations particularly after the second world war, with the exception of Engel (1939) [34].	121
6.10	Focus+context rendering of alignment overview of the results of the search for the word “ <i>Lust</i> ”.	122
6.11	An example of a translation variation between five editions of Baudissin (1832). Wolff (1926) stands out to be the most distinctive translation among the editions.	122
6.12	(a) Five classic canonical Baudissin translations. We can see the Wolff translation stands out due to the high values of Eddy. (b) After de-selecting the 1926 translation, we can see more variation among the four remaining translations.	124
6.13	The TLC view of the path containing the phrase “ <i>the aim reports</i> ”. The colored lines represent terms and each vertical line represents a translation. The annotated words and arrows illustrate the corresponding terms and lines.	124
6.14	A sub-set of the translation selected based on the domain expert knowledge shows three periods of generally high Eddy values. The three periods are highlighted by borders. . . .	126
6.15	Three uses of the TLC view to help the domain expert find variation between different translation. The top shows a common use of the words ‘ <i>Ihr</i> ’ and ‘ <i>Vater</i> ’. The middle and bottom views show that the words: ‘ <i>schätze</i> ’ and ‘ <i>gut</i> ’ which are used only in specific translations.	127
6.16	Two TLC views that show words used distinctively in the translations. In top the word “ <i>Maure</i> ” was firstly used by Schwarz (1941). In bottom, we can see the distinctive words in the translations of Shakespeare’s odd phrase: “ <i>Of love, of worldly matters and direction</i> ”. . . .	128

7.1	Our proposed methodological and interdisciplinary workflow. The workflow consists of three main components: the domain, task, and solution spaces. The tasks are informed by a communication channel between the two users' groups. A pre-visualization channel is attempted between the task and solution spaces to prepare for implementation. In the solution space, one or multiple solutions are implemented to address the predefined tasks. The terms expressiveness, purposefulness, and trustfulness indicate the quality criteria that need to be fulfilled to obtain useful outcomes (Section 7.3.2).	134
7.2	On the left, email that was exchanged in the early stages of the collaboration to enable sufficient understanding of domain-specific terminology. On the right, a screenshot of a formal document that explains the dataset components and terminology.	134
7.3	Samples of different recordings of our domain expert feedback sessions.	137
7.4	The design triangle by Miksch and Aigner [35]. They include factors to be considered during the design and implementation of interactive visualizations. Image courtesy of Miksch and Aigner [35].	138

Chapter 1

Introduction and Motivation

“The greatest challenge to any thinker is stating the problem in a way that will allow a solution.”

–Bertrand Russell (1872-1970)

Contents

1.1 Data Visualization	1
1.2 Information Visualization	3
1.3 Visualizing Variation in Translations of Shakespeare’s play <i>Othello</i>	9
1.4 Thesis Overview	18
1.5 Thesis Evaluation	19
1.6 Video Demonstrations	20

1.1 Data Visualization

Data visualization is defined by the Oxford dictionary as, “the representation of an object, situation, or set of information as a chart or other image” [36]. From this definition, visualization can be understood as a pipeline process that starts with data and ends with a graphical image that represents the information. However, the goal of data visualization is not only to present information but to produce knowledge and guide decision making. Data visualization can and should be viewed as a way to communicate information by graphical representation [37]. Chen et al. [38] formally describe the process of visualization as a function that maps data into images which facilitates a more efficient and effective cognitive process for acquiring information and knowledge.

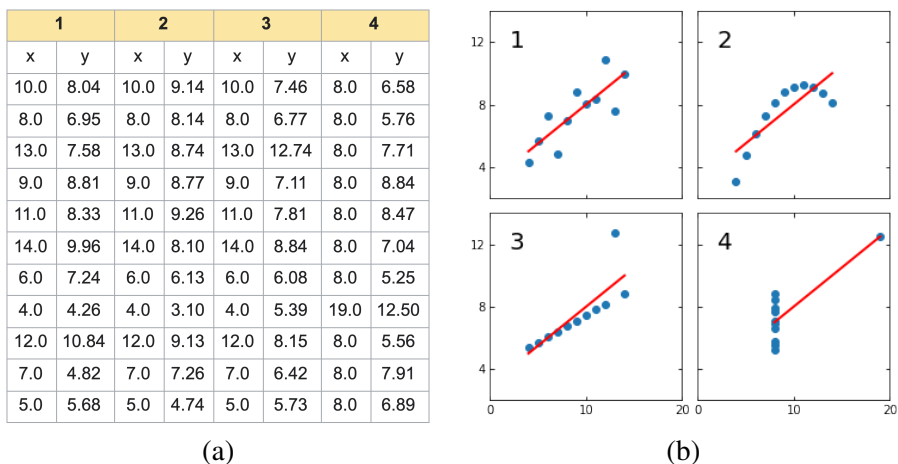


Figure 1.1: (a) The original Anscombe’s quartet dataset. (b) Four scatter plots that correspond to the datasets [12].

Value of visualization: It has become difficult to read a book, article, or newspaper without coming across some form of visualization (e.g. tables, navigation maps, weather charts, financial analyses, medical scans, organizational charts, networks of social media activities, etc.). Many of these can be globally interpreted independent of language. Human beings are visual creatures and process most information through vision. A great deal of information can be contained in a single image and visual imagery can be processed very quickly by the human brain [37].

It is obvious from these facts that visualization can provide more insightful explanation and ready comprehension of information than audible counterpart and make them more readily comprehensible. Patterns and knowledge can remain unrecognized in raw data even if the dataset is small. For example, Anscombe’s quartet dataset [12] consists of four sets of numbers. When we only look at the table of statistics, we might assume these sets are similar, as shown in Figure 1.1A. However, the scatter plots in Figure 1.1B disprove this assumption and show they are clearly different. This demonstrates the important role of data visualization in communicating knowledge and revealing interesting patterns within data.

History of visualization: Although modern data visualization is a relatively new field, humans have been making various graphical representations such as maps and cartography [39]. Friendly demonstrates that humans have employed graphic representations throughout history for a wide variety of reasons and functions. In their project, Friendly organizes the history

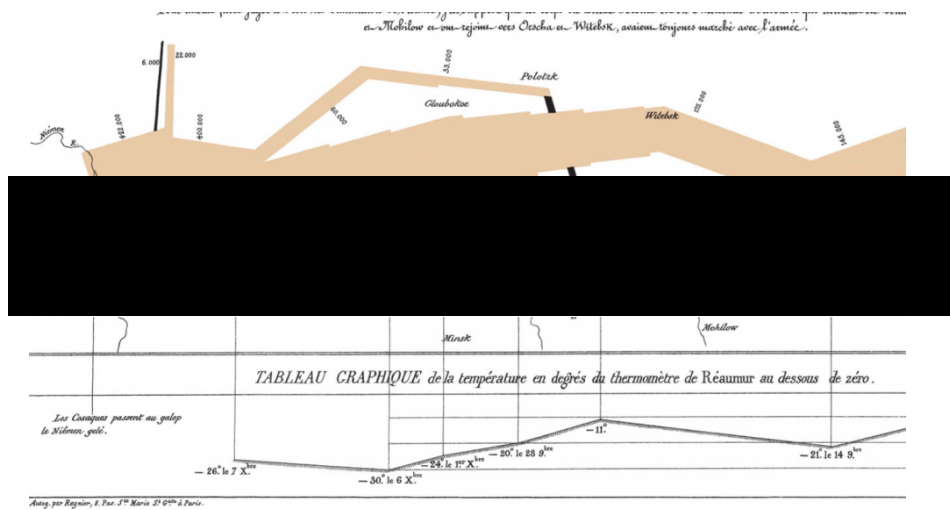


Figure 1.2: A visualization by Minard (1781-1870) illustrates the result of Napoleon's failed campaign of 1812. Image courtesy of Tufte [13].

of visualization into epochs showing uses of visualizations from the 16th century. A classic example is Minard's visualization (Figure 1.2) of the fate of Napoleon's Russian campaign. Playfair [14] produced a time-series graph of financial data in 1786 (Figure 1.3). Another notable early visualization is Snow's visualization (Figure 1.4) of the deaths resulting from a cholera outbreak in London in 1854.

1.2 Information Visualization

According to the ACM Computing Classification System (CCS) [40], information visualization is one of the four application domains: scientific visualization, visual analytics, information visualization, and geographical visualization. Here, we focus on information visualization and visual analytics as our thesis integrates the algorithmic data analysis and the domain user knowledge.

Over the past decade, the popularity of information visualization has grown rapidly. In 2017, McNabb and Laramee [41] reviewed 86 **survey** papers that review information visualization approaches. Rees and Laramee [42] include more than 80 books that focus on information visualization.

What is information visualization: Information visualization is distinguished from other visualization sub-fields as it deals with abstract data which has no spatial attributes. Keim

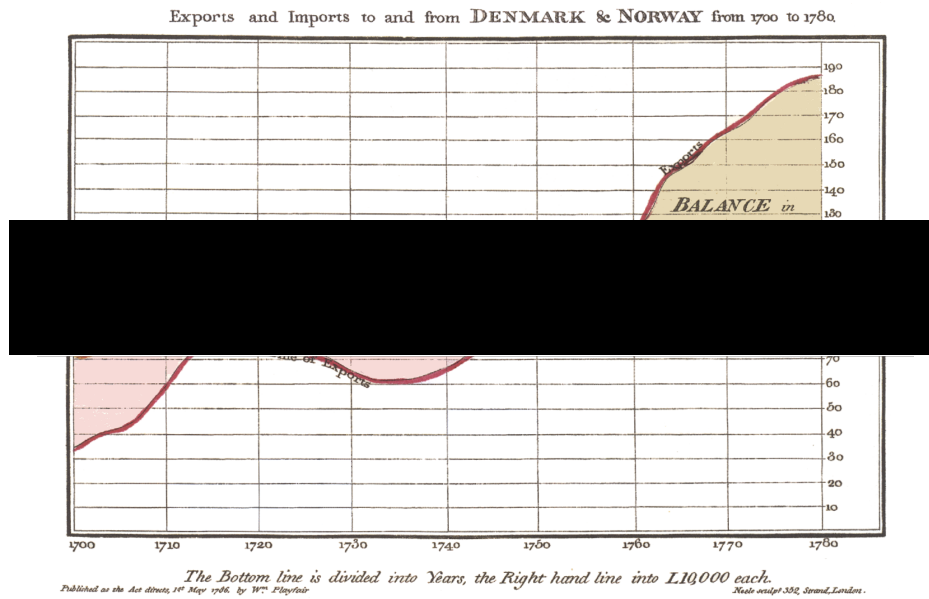


Figure 1.3: An image by Playfair (1759-1823) depicts the balance of trade. Image courtesy of Playfair [14].

et al. [43] define information visualization as, “the communication of abstract data relevant in terms of action through the use of interactive visual interfaces.” Card et al. [44] describe it as, “the use of computer-supported, interactive visual representation of abstract to amplify cognition.”

Challenges in information visualization: In 1755, the French philosopher Denis Diderot said, “As long as the centuries continue to unfold, the number of books will grow continually, and one can predict that a time will come when it will be almost as difficult to learn anything from books as from the direct study of the whole universe. It will be almost as convenient to search for some bit of truth concealed in nature as it will be to find it hidden away in an immense multitude of bound volumes.” The recent advancement of computational technologies, particularly the internet, have fulfilled this prophecy. It is now evident there is information overload and many users and organizations are overwhelmed by this excessive quantity of data. Hersh [45] reports that the existing online information was, in 2006, greater than the total sum of documents from the first 40,000 years of human history. This challenge of information overload is ongoing and increasing. Most smartphone applications collect data from users and internet browsers gather and track user activity. Cultural heritage are digitized in various

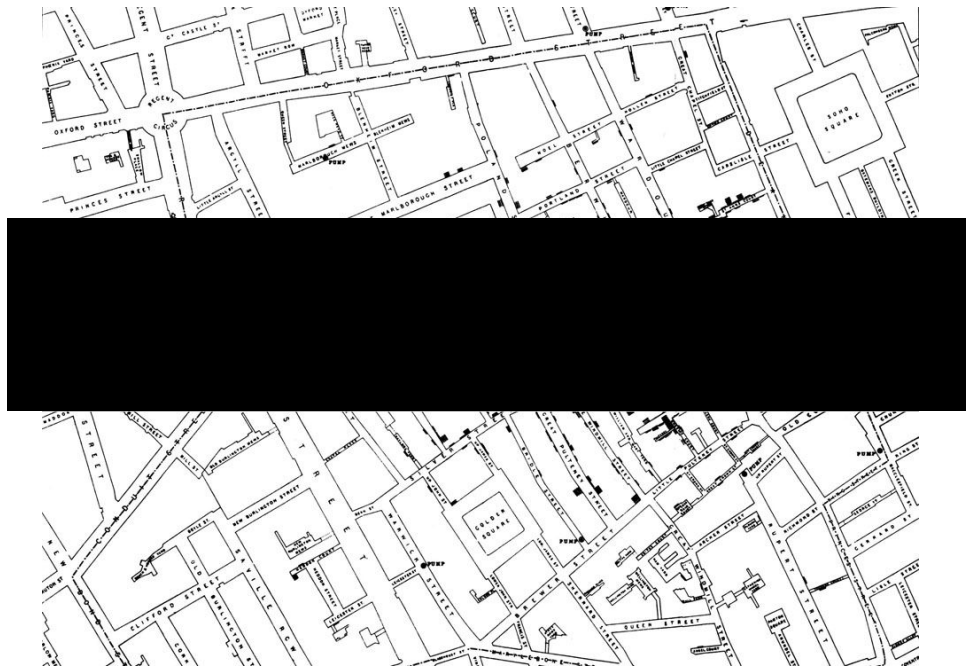


Figure 1.4: Snow's (1813 – 1858) map of deaths from a cholera outbreak in London in 1854. Image courtesy of Snow [15].

formats, and blogging and social media have enabled users to produce massive amounts of content.

Much abstract data is characterized as multivariate, that is, each data sample has multiple values. For abstract data, it is usually the case that each sample is constructed by multiple dimensions. Each variable can be considered a dimension because each sample can be mapped to a given set of properties. For example, if each sample of the data features two properties, we can visualize it using a 2D scatter plot. However, if the number of data dimensions increases, basic visualization techniques show limitations as it is a challenge to intuitively represent multi-dimensional datasets using limited visual encodings. Different visualization and interaction techniques have been proposed to address this. Inselberg's parallel coordinates plot [46] (Fig 1.5), for example, presents each dimension as a vertical axis which represents the range of the dimension. Each polyline represents the data point value across the dimensions.

Another challenge is representing data with relational structure. The relationships between data can come in different forms, and the more complex the interrelationships, the more difficult they are to represent and interpret. Several approaches have been developed to address this challenge. A treemap [47] is proposed to represent hierarchical data. It consists of nested

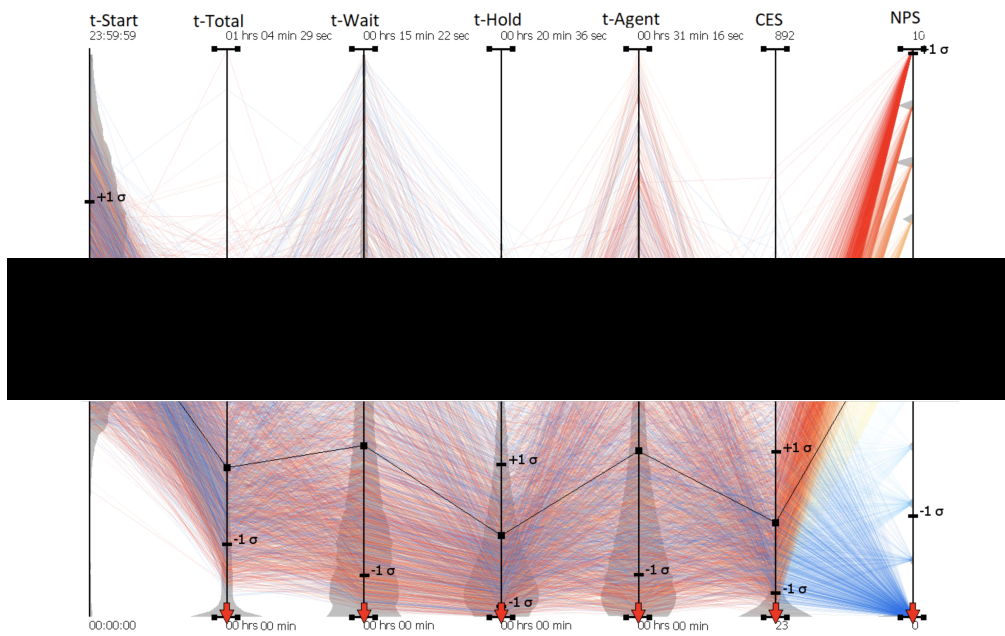


Figure 1.5: An example of a parallel coordinates plot. Image courtesy of Roberts et al [16].

rectangles of sizes proportional to the corresponding data values [17] (1.6). Treemaps are reimplemented and optimized using different algorithms. Bruls et al. [48] created a treemap with more square nodes for easy comparison, while Shneiderman and Wattenberg [49] developed an algorithm that maintains the order of the nodes. Node-link diagrams are also widely used to depict relational structure (Figure 1.7).

A further emerging challenge is that the cognitive and perceptual capabilities of users are challenged when attempting to understand high-dimensional data. In such situations, unavoidably, the visualization can suffer from visual clutter and occlusion. Applying adequate user interaction and exploration techniques is therefore essential to intuitively refining the result and observing the underlying analysis models. For example, focus+context techniques have been applied to treemaps to focus on and enlarge selected nodes [50,51]. As well as this, several brushing and filtering methods have been applied to parallel coordinates to address overplotting and occlusion challenges. Figure 1.8 shows an overview of brushing approaches that facilitate the axis selection and brushing interaction in parallel coordinates plots [52].

What is visual analytics Visual analytics refers to the science of analytical reasoning supported by interactive visual interfaces [53]. It is thought that the first appearance of this term was in 2004 [19]. It is a response to the information overload challenge as the acquisition of



Figure 1.6: An example of a treemap plot which represents the change in the U.S. stock values. Image courtesy of Laubheimer [17].

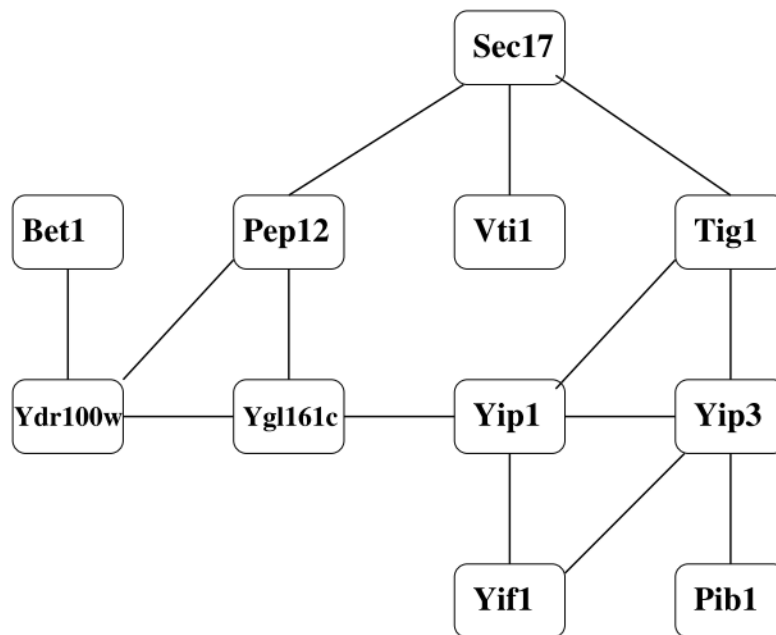


Figure 1.7: An example of a node-link diagram that represents a small protein interaction network. Image courtesy of Michailidis [18].

data is no longer the a problem, however, to utilize the data and turn it to reliable knowledge still is [54]. The goal of visual analytics is to integrate automatic data analysis, interactive visualization, and human knowledge in order to derive new knowledge from the raw data. Figure

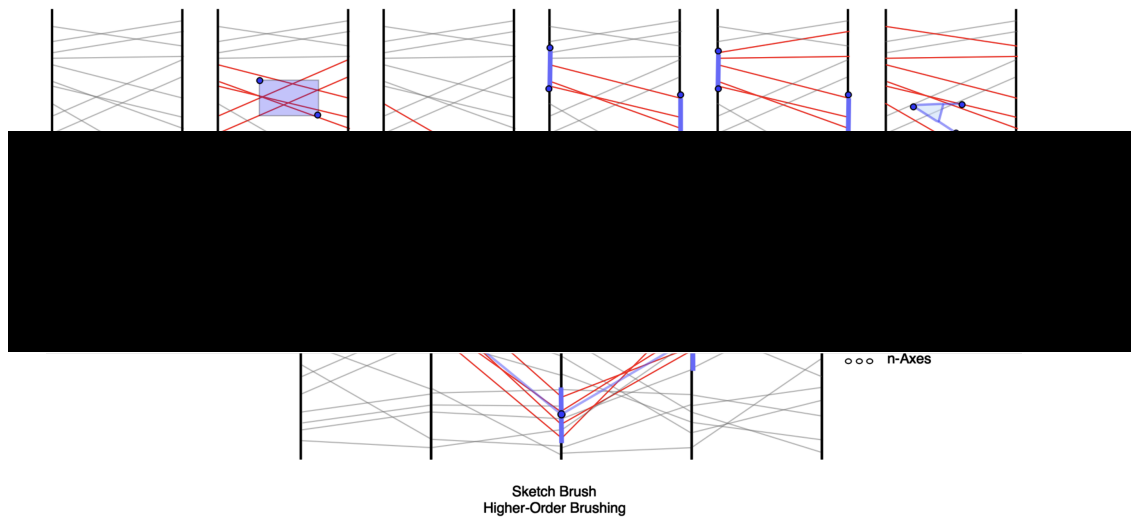


Figure 1.8: An overview of the different brushing approaches that facilitate selection and interaction. Image courtesy of Roberts et al [16].

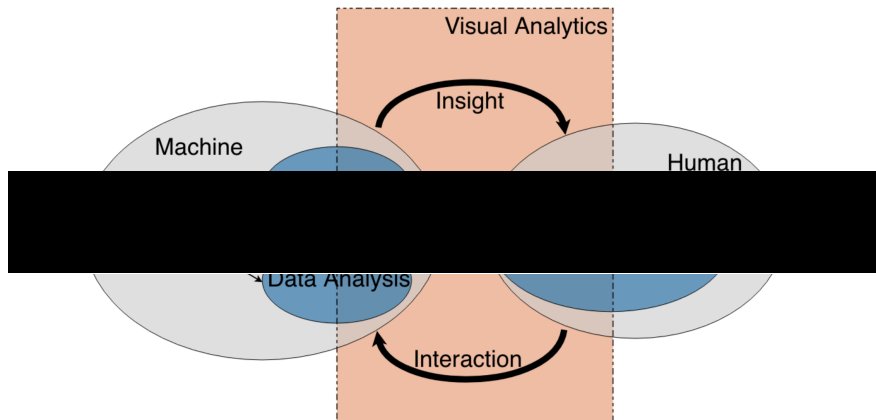


Figure 1.9: The integration between data analysis, visualization, and human in the visual analytics process. Image courtesy of Cui [19].

[1.9] shows how human judgment (cognition, perception, analytical reasoning), interactive visualization, and data analysis are integrated in the visual analytics process to support insight [19].

1.3 Visualizing Variation in Translations of Shakespeare's play *Othello*

This work is carried out in close collaboration with the College of Arts and Humanities in Swansea University under a collaborative project scheme founded in 2011 called ‘Translation Arrays: Version Variation Visualisation’ [55]. The project is responsible for collecting, aligning, and warehousing the dataset under examination along with other ‘multi-retranslation’ datasets. The team has developed prototype online tools [20] for managing such datasets and developing visualization to explore and analyze them. Professor Cheesman is a specialist in modern and contemporary German literature and culture. He has been researching German culture and translating German literature since the early 1980s. Professor Cheesman has been investigating the history of German translations of Shakespeare's *Othello* since 2009, using traditional qualitative methods (contextualised close reading) and experimental, quantitative, digital methods. Relevant online outputs, presentations, and published articles by him and his collaborators are listed on the project's website [20]. The articles include publications in *Digital Scholarship in the Humanities* [56] and *Journal of Data Mining and Digital Humanities* [57].

1.3.1 Experience of Collaboration

In this section, we report the experience of the collaboration between the digital humanities and visualization teams at Swansea and Nottingham Universities. Figures 1.10 and 1.11 illustrate the most important events and contributions brought about by this collaboration. These events span two phases that relate to this thesis: The first begins in 2011, which marks the start of the collaboration, and the second starts in 2017, which is considered the first year of this thesis.

2011 - 2016: This is illustrated in Figure 1.10. The collaboration began in 2011 when Swansea University funded the Translation Arrays: Version Variation Visualization (VVV) project. The project also received funding from the Arts and Humanities Research Council (AHRC) to build the VVV online tool. The online VVV page is being used by several researchers and it an actual functioning tool. The project was mainly led by Professor Tom Cheesman and Dr. Robert S. Laramee and involved Dr. David M. Berry, Professor Andy Rothwell, Dr. Jonathan Hope, Kevin Flanagan, and Stephan Thiel. Dr. Zhao Geng supported the early stages of the collaboration as a Ph.D. student.

1. Introduction and Motivation

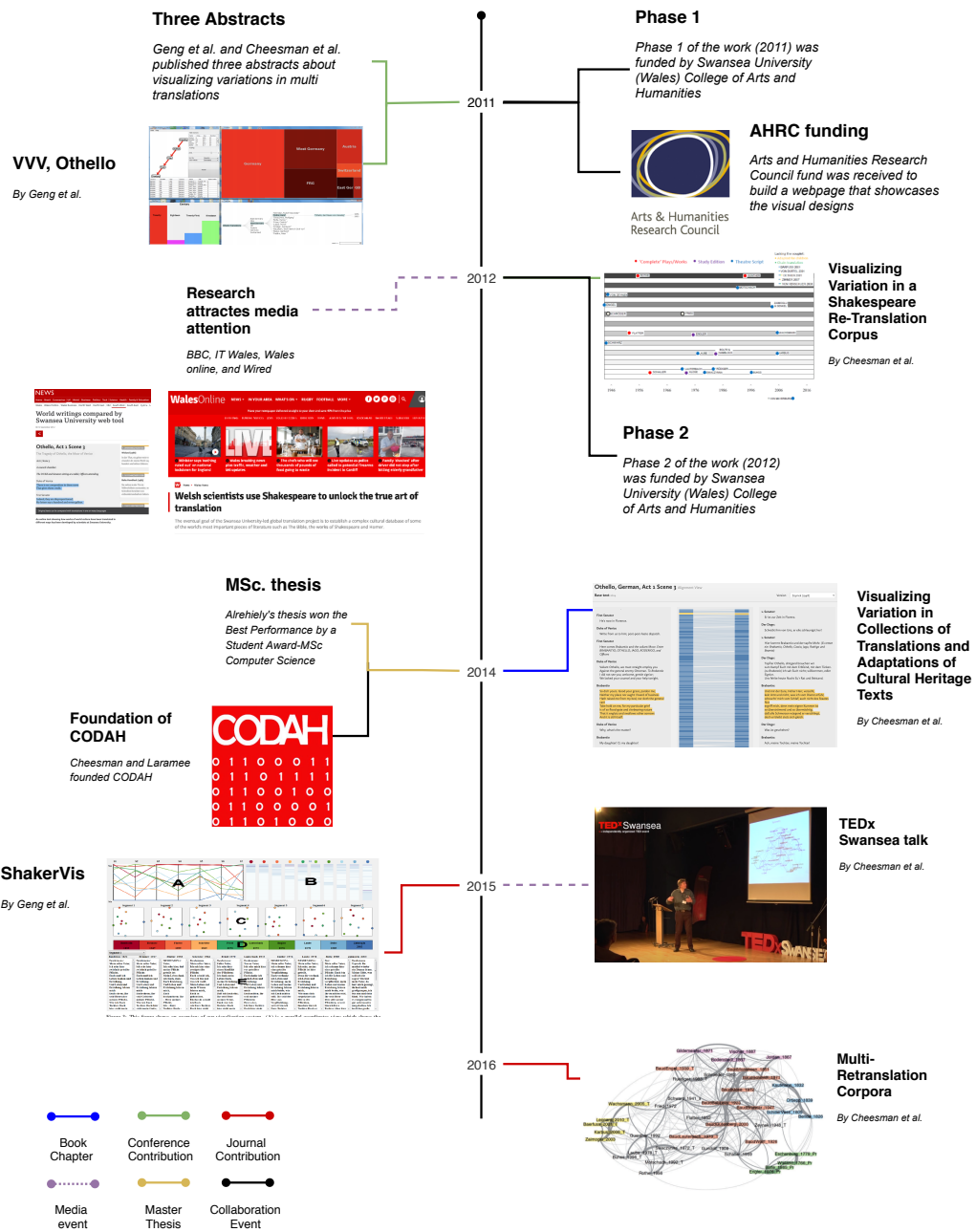


Figure 1.10: Timeline of key events and outputs from the collaborative work between the school of arts and humanities and the visualization group at Swansea University (2011-2016). This figure illustrates the time prior to the beginning of this thesis.

Geng et al. [58] published a book chapter that presents the first published result. They contribute a structure-aware treemap of Shakespeare’s *Othello* collection. They also introduce

1.3. Visualizing Variation in Translations of Shakespeare's play Othello

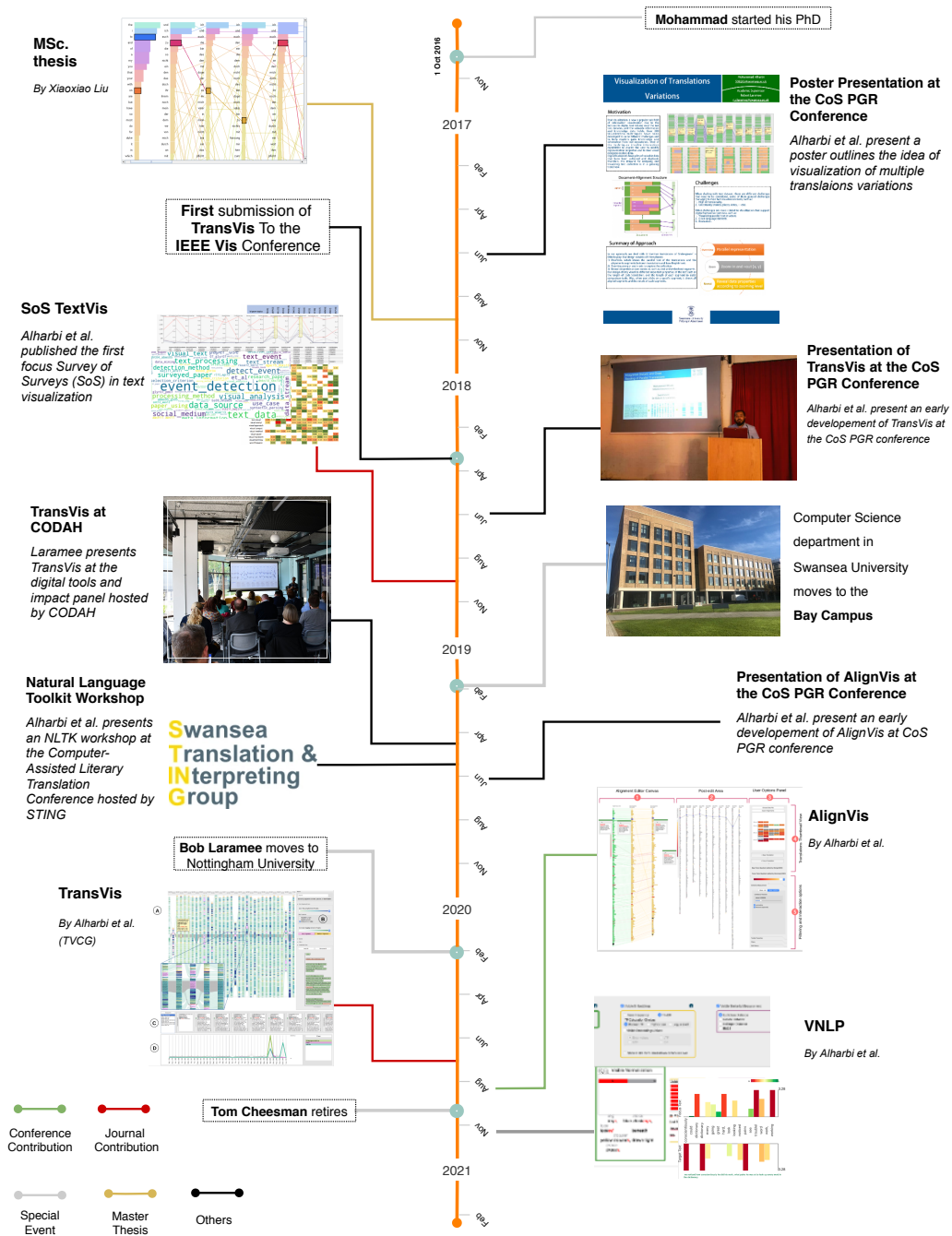


Figure 1.11: Timeline of the most key events and contribution of the collaborative work between the school of arts and humanities and the visualization group at Swansea University (2017-2020). This figure illustrates the collaborative events and publications that coincide with our thesis.

a focus+context parallel coordinates layout for comparing the term frequency of eight segments of the collection. In 2012, an online tool developed by Cheesman et al. [20] that analyses and compares different translations of Shakespeare's *Othello* featured at Shakespeare's Globe Theatre in London. This event attracted media attention and was reported by BBC [59], Wired, and I.T.Wales. In 2014, Cheesman et al. [60] published the results of the development of the project website such as the time-map and the alignment map. They implemented an "Eddy" and "Viv" interface for the first time, integrated on the project website.

The Centre on Digital Arts and Humanities (CODAH) was founded by Cheesman and Laramée in 2014. CODAH aims to develop links and share knowledge between staff and students in Arts and Humanities and Computing (as well as other disciplines) in terms of research, teaching, public impact, resourcing, and strategy [61].

Alrehiely was the first Master's student with a focused thesis on the visualization of Shakespeare's *Othello*, winning the Best Performance by a Student Award-MSc Advanced Computer Science in the College of Science in 2015. She published a follow-up book based on this thesis [62].

In 2015, Cheesman presented the VVV project at TEDx Swansea [63]. In this talk, Cheesman discusses the idea of visualization of translation variation and how visualization could help explore and understand why and how translations differ. In this year, Geng et al. [64] proposed ShakerVis, an interactive focus+context visualization to present, analyze, and explore variation at the segment levels. Cheesman et al. [56] went on to extend their previous review of the online tools they proposed in the VVV project [20] in a journal article published in the *Digital Scholarship in the Humanities* journal.

2017 - 2020: I commenced Ph.D. study in October 2016, highlighted in orange in Figure 1.11, and have engaged in multiple interdisciplinary seminars and presented the results of work across various venues. I presented a proof-of-concept visualization (poster) of *Othello* translations at the first College of Science (CoS) Postgraduates (PGR) Conference in 2017. In 2017, Xiaoxiao Liu completed an MSc project on the *Othello* translations [65].

In 2018, I presented the first focus Survey of Surveys (SoS) in text visualization at the CGVC conference [4], which was then extended and published in the *Computers Journal* [3]. In addition, TransVis [2] has been presented on multiple occasions, including at the second CoS PGR conference in 2018 and in two meetings with Professor Ben Shneiderman (University of Maryland) and Dr. Max Wilson (University of Nottingham). It was also introduced and

1.3. Visualizing Variation in Translations of Shakespeare’s play Othello

Date	Subject	Attendees	Duration (H:M)	Video Url
07-02-2017	preliminary discussions	R. Laramee, T. Cheesman, and M. Alharbi	01:30	–
20-06-2017	preliminary discussions	R. Laramee, T. Cheesman, and M. Alharbi	01:30	–
13-07-2017	preliminary discussions	R. Laramee, T. Cheesman, A. Peljak-Łapińska, L Xiaoxiao, and M. Alharbi	0:36	https://youtu.be/EI-3P2hgFek
16-10-2017	preliminary discussions	R. Laramee, T. Cheesman, A. Peljak-Łapińska, L Xiaoxiao, and M. Alharbi	01:30	–
20-11-2017	preliminary discussions	R. Laramee, T. Cheesman, A. Peljak-Łapińska, L Xiaoxiao, and M. Alharbi	01:31	https://youtu.be/vo-GTYY8MJA
04-12-2017	preliminary discussions	R. Laramee, T. Cheesman, A. Peljak-Łapińska, L Xiaoxiao, and M. Alharbi	00:28	https://youtu.be/1__Dm00qrgg
22-01-2018	TransVis feedback	R. Laramee, T. Cheesman, A. Peljak-Łapińska, and M. Alharbi	01:30	–
15-02-2018	TransVis feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:50	https://youtu.be/w-Ynra_zI04
22-2-2018	TransVis feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:01	https://youtu.be/QQ9khMLbXAM
01-03-2018	TransVis feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:24	https://youtu.be/LTmZ3wGE-BQ
15-03-2018	TransVis feedback	R. Laramee, T. Cheesman, A. Peljak-Łapińska, G. Change, and M. Alharbi	00:51	https://youtu.be/JT_rFquYsLM
24-09-2018	AlignVis preliminary discussion	R. Laramee, T. Cheesman, A. Peljak-Łapińska, and M. Alharbi	01:30	–
05-06-2019	AlignVis feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:30	–
20-09-2019	AlignVis feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:18	https://youtu.be/CI_K7GVrjro
08-11-2019	AlignVis feedback	T. Cheesman, and M. Alharbi	00:58	https://youtu.be/ddiytcdiiGU
22-11-2019	AlignVis feedback	T. Cheesman, and M. Alharbi	00:30	https://youtu.be/iCYBpt_o0so
15-11-2019	AlignVis feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:20	https://youtu.be/X9MYsqAh-YE
13-12-2019	TransVis feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:07	https://youtu.be/19B16jBglbs
06-07-2020	VNLP feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:19	https://youtu.be/9EkGaNR4fPY
03-07-2020	VNLP feedback	R. Laramee, T. Cheesman, and M. Alharbi	01:06	https://youtu.be/QOwNCCxaNhI
24-08-2020	VNLP feedback	R. Laramee, A. Rothwell, and M. Alharbi	00:54	https://youtu.be/uNL22M0UMKk
Total duration			25:13	

Table 1.1: A list of the collaborative and domain expert feedback sessions related to this thesis.

presented at the digital tools and impact panel hosted by CODAH in 2019. In the third CoS PGR, we first presented the idea of AlignVis [1].

In 2019, the Swansea Translation and Interpreting Group organized a two-day workshop in which I took part by leading an NLTK (Natural Language Toolkit) workshop. This afforded an opportunity to meet and engage with the domain scholars of our collaboration as the workshop concerned the tools they employ in their research. This motivated the AlignVis project through efforts to find domain problems and gaps.

In 2020, TransVis was published in IEEE Transactions on Visualization and Computer Graphics (IEEE TVCG) [2] and AlignVis was published at the International Conference on Information Visualization (IV). In the same year, Dr. Laramee moved from Swansea University to the University of Nottingham and Professor Cheesman retired from the School of Arts and Humanities.

History of TransVis: Figure 1.12 illustrates the history of the publication of our milestone paper TransVis [2] (Chapter 6). Many successful publications go through a lengthy submission and review process, yet this story is never told. Throughout our collaborative networking

1. Introduction and Motivation

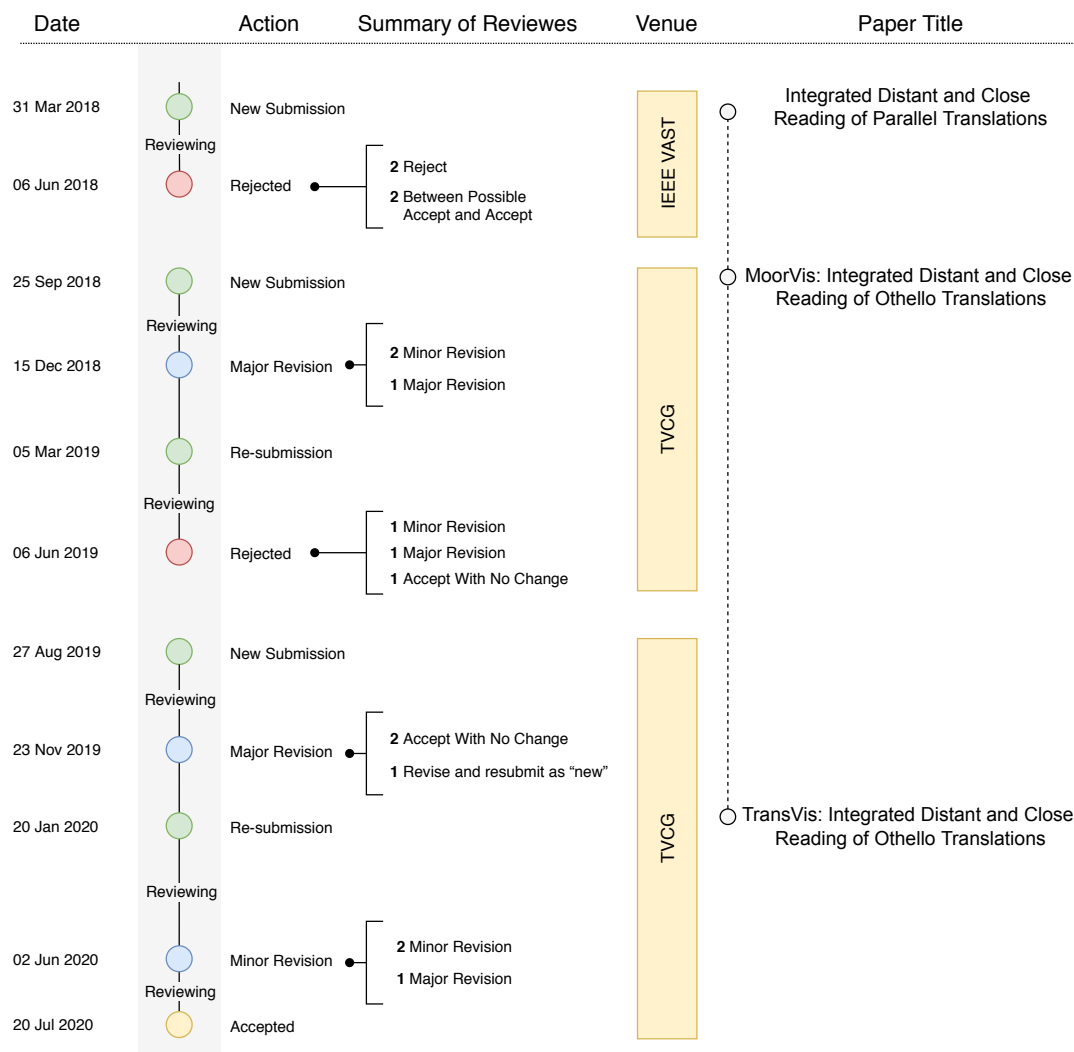


Figure 1.12: The history of the submissions of our paper TransVis [2].

and conference participation, we heard similar stories of impactful work with long histories of submission and review. TransVis is the successful output of our collaboration with digital humanities and it has gone through a considerable process of review and submissions, first submitted to the IEEE VIS VAST track in March 2018, then to the IEEE Transactions on Visualization and Computer Graphics (TVCG). In TVCG, it was submitted twice and underwent two major revisions and one minor revision. The title of the paper itself evolved through this process, as shown in Figure 1.12. Eventually, the paper was accepted by TVCG in July 2020 and presented at the IEEE VIS 2020 conference [66].

Table 1.1 lists our collaborative and domain expert feedback sessions. For each meeting,

1.3. Visualizing Variation in Translations of Shakespeare's play *Othello*

```
'<eblacorporus name="Othello, Act 1 Scene 3" description="The Tragedy of Othello, the Moor o:
  <documents>
    <document name="" authortranslator="William Shakespeare and numerous editors" copyrigh
      (2006) for added dialogue and modern spelling (+ further modernisation e.g. -'d to -ed
      ><doccontent>...</doccontent>
      ><segmentdefinitions>...</segmentdefinitions>
    </document>
    <document name="Bärfuß" authortranslator="Lukas Bärfuss" copyrightinfo="© Hartmann & S
      (2003), have been translated into a dozen languages. His adaptation of 'Othello', subt
      2001 and was later also used as the basis of the script for a film. 'Othello, ein Blue
      a mockumentary about a megalomaniac film director who is making a film out of 'Othello
      play), but a very different approach to language, features in Alvaro García de Zúñiga'
      2001 Edition date:-- Explicit ancestor version:-- Ancestor date:-- Reference name:-
      Complete Works, SE= Study Edition, MS= Manuscript, TS=Theatre Script):-- TS Form group
      year:-- Main author Wikipedia entry?:-- Yes Main author other biographical informatio
      Publication details (source used):-- Othello – Kurze Fassung. Stück in 5 Akten nach Wi
      Stauffacher GmbH, Bismarckstr. 36, 50672 Köln] Reference in Blinn/Schmidt 'Shakespeare
      Blue-Movie, details and clips at: http://400asa.ch/film/archiv/othello/ Clips also upl
      Zúñiga's RadiOthello, http://blablalab.net/en/index.php?title=RadiOthello; orig French
      germany.de/index.php?page=play_det&id=2345&foreigner=true&l=1 Comments:-- Premiere: D
      and layout normalised: TC. Data entry by:-- TC, August 2012" langcode="deu" referenced
    <doccontent>
      <blockquote>
        <i>
          <q data-eblattype="startmarker" data-eblasegid="37239">[</q>
            3. Szene
          <q data-eblattype="endmarker" data-eblasegid="37239">]</q>
          </i>
        </blockquote>
        <blockquote>
          <i>
            <q data-eblattype="startmarker" data-eblasegid="37240">[</q>
              Venedig. Palast des Dogen. Doge. Brabantio. Othello. Desdemona. Roderigo. Jago.
            <q data-eblattype="endmarker" data-eblasegid="37240">]</q>
            </i>
          </blockquote>
        <p>
          <b>DOGE</b>
        </p>
      </blockquote>
```

Figure 1.13: Screenshot of the XML file of the dataset [20].

the date, subject, attendees and duration of the meetings is indicated, and an online link to the recorded meetings and feedback sessions provided. The total duration of the collaborative meetings and sessions is about 25 hours.

1.3.2 Description of Parallel Translation Data

Cheesman has collected 55 translations of Shakespeare's play *Othello* (1604) Act 1, Scene 3 into German. The translations span from 1766 to 2010. To date, 38 translations of the collection have been optically scanned from paper prints, corrected for OCR errors, segmented, and aligned with the English base text to create a parallel corpus. The set of translations studied here was collected over 2-3 years from various sources, including libraries, second-hand book-sellers, archives, theater publishers, and theater companies. The translation data is stored in XML format (Figure 1.13) on the project's website [20].

Each <document> node in the dataset XML file represents a German translation of the base English text and is associated with a variety of metadata such as authors <author-translator> and description of the translation <description>. The <document>

node consists of three nodes: <doccontent>, <segmentdefinitions> and <alignments>. The base English text is also represented as a node of <document>, and does not have the <alignments> sub-node.

1. <doccontent>: contains the actual content of the document using a number of <blockquote> nodes. Each <blockquote> has a number of <q> nodes which contain the actual text segments.

2. <segmentdefinitions>: stores the meta-data associated with each segment such as segment ID, length, speaker, etc.

3. <alignments>: is included in all documents except the base text. It consists of a number of <alignment> nodes which match a given segment of a translation with the corresponding segment of the base text using the *segment ID* elements stored in the <segmentdefinitions> node.

1.3.3 Domain Problem and Research Objectives

Visualization solutions are increasingly adapted in digital humanities as it is evident that they create new modes of knowledge and facilitate more effective discovery of new observations [26, 67, 68]. Moreover, Hinrichs and Forlini [68] propose to integrate visualization within the research process and not just a means to an end.

The initial purpose of the collaboration with the school of Arts and Humanities was to explore the potential value of visualizing their proprietary parallel translations. They aim to investigate the extent to which visualizations enable them to explore, analyze, and compare variation between translations. This is significant, as well as challenging, as it helps to study the language evolution and reforms over the lifespan of the translations and the historical and political contexts of each translations. It can also reveals some distinctive translations and helps understand how and why they vary.

In preliminary discussions with the domain expert, we formulate a set of primary research objectives.

[Ob1] A review of surveys of text visualization: The first text visualization survey article was published in 2010 in response to the increase in text visualization literature. The number of surveys that review and classify existing techniques based on text research methodology is growing. Our objective is to present the first Survey of Surveys (SoS) which **reviews** all of the surveys and state-of-the-art papers on text visualization techniques, **classifies** them, provides **recommendations**, and discusses reported **challenges**.

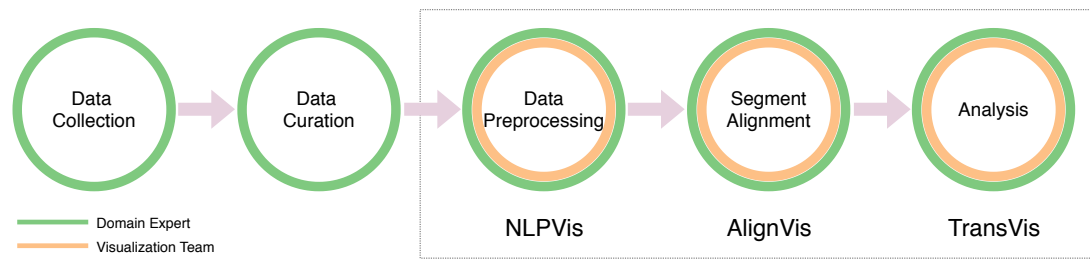


Figure 1.14: The domain expert workflow. The first two stages: data collection and curation are performed by the domain expert. Our thesis contributes to the later three stages: data preprocessing (Chapter 4), segment alignment (Chapter 5), and analysis (Chapter 6)

[Ob2] A transparent and informative visual design that uncovers Natural Language Processing (NLP) results: Visualizations that are usually designed for digital humanities tend to be opaque [69]. Therefore, our objective is to propose a novel interactive design that enables users to explicitly observe the NLP pipeline processes and update the parameters at each processing stage.

[Ob3] A semi-Automatic visual alignment approach: Prior to our collaboration, the domain expert facilitates a manual, time-consuming approach to aligning two translations. It involves the use of different tools and the reading of the translation line-by-line in order to achieve the desired alignment. Therefore, our objective is to design an interactive visual design that helps create, enhance, and accelerate the alignment process and enable modification of the alignments.

[Ob4] Integrated close and distant reading of alignments: Current visual approaches do not support the integration of both close and distant reading and the support of comparison of multiple parallel translations. Our objective is to propose an interactive design that supports (1) integrated distant and close reading in the same view, and (2) comparison of multiple parallel translations to facilitate the exploration of how and why translations vary, historical language evolution and reforms, and the historical contexts of translations.

[Ob5] A methodological and collaborative workflow: Multidisciplinary work is a commonly reported challenge in the literature. We propose a methodological workflow to guide such interdisciplinary research projects.

1.4 Thesis Overview

The chapters in this thesis are organized to fulfill the research objectives stated in Section 1.3.3. They flow with the domain expert workflow illustrated in Figure 1.14. Typically, the domain expert carries out the first two stages, data collection and curation. Therefore, our thesis contributes to the last three stages to help analyze and visualize parallel translations. Chapter 4 proposes a visual and interactive design to facilitate understanding of the NLP preprocessing pipeline. For the segment alignment, Chapter 5 presents a semi-automatic, visual alignment tool. Chapter 6 contributes an interactive design to visualize the parallel translations. The following comprises a detailed overview of this thesis.

In Chapter 2, the first research objective [Ob1] is addressed and an extended survey of the reviews of the literature in the field of text visualization is presented and classified based on five themes: (1) Document-centered, (2) user task analysis, (3) cross-disciplinary, (4) multi-faceted, and (5) satellite-themed. We summarize each survey classification and its features in order to facilitate comparison of the surveys, and provide future work recommendations for researchers in the field of text visualization [4].

Chapter 3 introduces the background research of the following contribution chapters. It also highlights how our work differs from previous research.

Chapter 4 addresses the fourth research objective [Ob2], presenting a framework that enables users to observe and participate in the NLP pipeline processes, explicitly interact with the parameters of each step, and observe the effects on the visible VNLP result [10].

The second research objective [Ob3] is addressed in Chapter 5. It contributes an interactive design that combines interactive visualization with domain knowledge intervention to facilitate the alignment of parallel translations, as well as a visual alignment design that helps translation scholars align multiple texts simultaneously. This also facilitates interaction methods that help create, enhance, and accelerate the alignment process [1].

Chapter 6 addresses the third research objective [Ob4]. An integrated visual design is proposed to support distant and close reading of the collection of parallel translations of *Othello*. The design leverages a range of exploration and interaction techniques to facilitate analysis and exploration of the parallel translations. The design supports the analyses and exploration at both the segment and term levels [2].

Chapter 7 refers to the final research objective [Ob5], proposing a methodological workflow for interdisciplinary research with the digital humanities.

1.5 Thesis Evaluation

We work closely with the domain expert and the project is driven by a real-world historical investigation. With the close observation by the domain expert, we designed three applications to satisfy the user requirements stated in Chapters 4, 5, and 6 and facilitate exploration and interaction to enhance the user experience. We sought feedback from a Modern Languages and Translation expert. We have conducted more than 20 in-person, regular feedback sessions (about 25 hours) to demonstrate our work features. All of the sessions are video-recorded for post-analysis and archiving gathering. Our semi-structured interview questions were previously planned and guided by Hogan *et al.* [70]. Hogan *et al.* proposed an Elicitation Interview technique from a visualization perspective. The Elicitation Interview technique is a qualitative evaluation method that allows users to capture and understand analysis processes and experiences with static and interactive visualizations. This technique follows several iterative phases where the user is encouraged to describe their experience at finer levels of granularity. One of the key elements in such techniques is that the mode of questions used throughout the Interview is non-inductive (content-empty) and directive to obtain precise experience without imposing their own presuppositions and maintain the participant's attention on a singular experience. We facilitated this technique to guide the interviewing discussion and questions. Our analysis methodology, however, does not align with Hogan *et al.*'s technique when it relates to the analysis and documentation of the qualitative data. The analysis of our evaluation qualitative data follows an unstructured theme as we evaluate our design with one domain scholar due to the nature of the dataset examined. We believe that a structured evaluation and content analysis could be useful for future directions. After each session, the videotape and notes are analyzed to extract as much detailed feedback that describes the domain's scholar experience as possible. The feedback is summarized and stated. During the feedback sessions, different patterns are observed, such as the discovery of software bugs, and the discovery of data-level errors. The first few sessions of each project consist mostly of software demonstrations to guide the development of features. However, these gradually turned into active hands-on use of the software by the domain expert.

Beside the domain expert feedback, in the following we state how each chapter is evaluated:

- Chapter 4 includes an application to text similarity quantification to demonstrate its utility and advantages.
- Chapter 5 includes domain expert feedback, a comparison with a standard alignment tool

Chapter	Title	Video Url	Presentation Url
Chapter 3	VNLP: Visible Natural Language Processing	https://youtu.be/DJYvO6QMF6s	
Chapter 4	AlignVis: Semi-Automatic Alignment and Visualization of Parallel Translations	https://youtu.be/SCZ3G3nRH28	https://youtu.be/cQ-ci1Z361k
Chapter 5	TransVis: Integrated Distant and Close Reading of <i>Othello</i> Translations	https://youtu.be/FnA1YbWdiNQ	https://youtu.be/8HsUpCiA4gc

Table 1.2: The video demonstrations of our design studies.

and computational and visual alignment tools.

- Chapter [6](#) includes detailed examples, observations, and a case study.

1.6 Video Demonstrations

Each visual design chapter includes a supplementary video that demonstrates the design features and their utility. We recommend viewing these videos to gain an overall overview of the contributions chapters. Table [1.2](#) lists the video demonstrations and the corresponding video links.

Chapter 2

SoS TextVis: A Survey of Surveys on Text Visualization

“Human beings, who are almost unique in having the ability to learn from the experience of others, are also remarkable for their apparent disinclination to do so.”

–Douglas Adams (1952 – 2001)

Contents

2.1 Introduction and Motivation	22
2.2 Summary and Comparison of Surveys	26
2.3 Discussion of Future Challenges	38
2.4 Chapter Summary	43

This chapter aims to address the first research objective [Ob1]. It presents the results of the survey, a version of which is published in the MDPI journal Computers [71]. The contents of this chapter can be considered as a standalone research chapter rather than providing a background to the research topics discussed in this thesis. The following individual chapters each provide their own background research. When I started in 2016, we discussed potential survey papers for the thesis, we considered general text visualization. However, we found that there are multiple surveys and Kucher et al. [22] has filled the gap. As a result, we also found the cross-disciplinary survey which fits the goal of this thesis and includes techniques to support Digital Humanities Jänicke et al. [24] which fits the goal of this thesis. At this point, we decided that a survey of surveys would be a logical direction forward as it summarizes the the landscape of survey papers in text visualization. We looked at the benefits of surveying

different survey papers which would allow us to gain a good understanding of unsolved problems, specifically for cross-disciplinary collaborations which we discuss throughout the thesis. Also, we facilitate the classification that each survey presents as a guide for us to focus on specific literature. For example, in Jänicke et al. [24], we focused on the parallel text analysis dimension which aligns with the nature of our dataset. Also, in Kucher et al. [22] we engaged with the literature presented in the comparison and text alignment category. As this survey is considered as a standalone research chapter, it was an important step for my thesis as it facilitates the understanding of the field of text visualization, the state-of-the-art literature on text visualization approaches, common challenges and future work.

In this chapter, we present the first Survey of Surveys (SoS) that reviews the state-of-the-art papers on text visualization techniques. The survey categorizes them into five groups: (1) document-centered, (2) user task analysis, (3) cross-disciplinary, (4) multi-faceted, and (5) satellite-themed. In this chapter, survey recommendations are provided for researchers in the field of text visualization. The result is a unique, valuable starting point and overview of the current state-of-the-art in text visualization research literature.

2.1 Introduction and Motivation

Text visualization and visual text analysis is a rapidly growing sub-field of information visualization and visual analytics. Therefore, many approaches and techniques are introduced periodically to help users and researchers with a wide range of tasks. The volume of digital text data is multiplying as a result of the popular demand for digital text and text digitization projects, such as those by Reddy and StClair [72], Andre and Eaton [73], and Mendelsson et al. [74]. As literature and historical documents are digitized for further study and analysis, the volume of digital text data makes understanding and analyzing it extremely challenging. Text documents by their nature bring challenges inherent to natural language such as high dimensionality, irregularity, and uncertainty. Advanced techniques are therefore required to address these challenges.

Kucher and Kerren [22] review over 400 text visualization approaches in their interactive web-based tool ‘Text Visualization Browser’ (at time of writing and the tools are regularly updated). However, the approaches listed in the Text Visualization Browser come mainly from the data visualization community and generally do not include literature from other communities, particularly the digital humanities. The number of text literature surveys has grown since

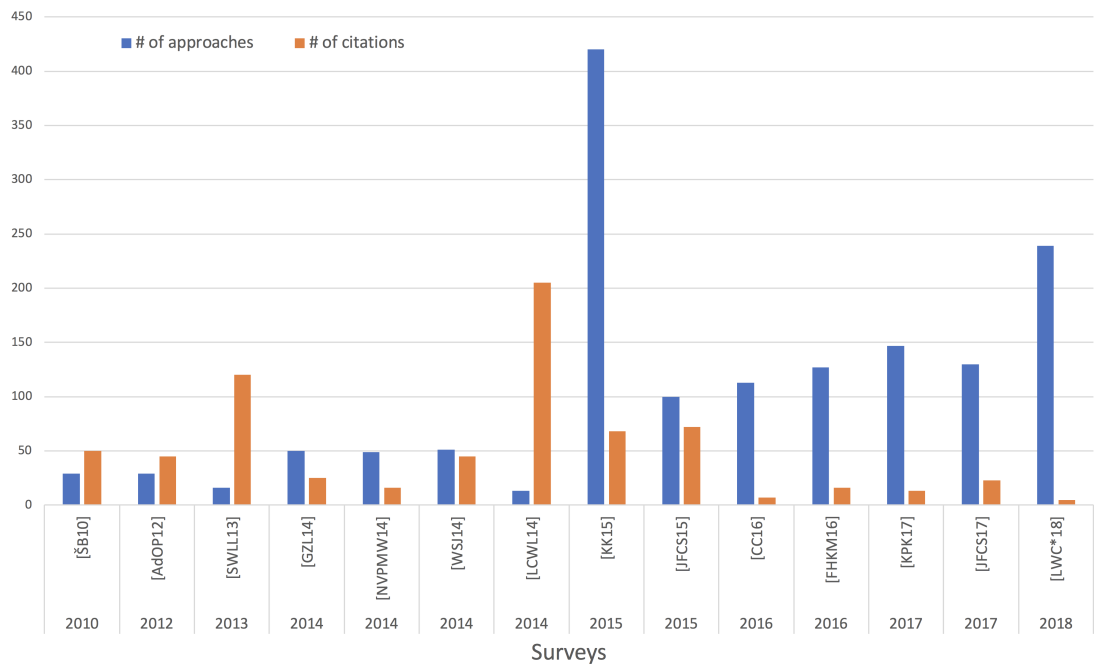


Figure 2.1: Text visualization surveys from 2010 to 2018. Blue bars indicate the number of methods reviewed in each survey. Orange bars show the number of citations each survey attracts. In term of the number of surveys, 2014 dominates with four. However, with respect to the number of techniques, surveys from 2015 collectively review 480 methods.

the first was published in 2010 by Šilić and Bašić [21], as shown in Figure 2.1. Collectively with duplicates, the surveys cite and review 1288 text visualization approaches.

This review provides a meta-survey of the existing surveys that address the exploration, analysis, and presentation of text data.

Our contributions to the field include:

- the first focused Survey of Surveys (SoS) in text visualization,
- a novel classification of text surveys in the reviewed literature,
- helpful survey meta-data to facilitate comparison of the surveys, and
- a unique, valuable starting point and comprehensive overview for both newcomers and experienced researchers in text visualization.

The rest of this chapter is organized as follows: Section 2.1.1 describes the methodology used to collect related research papers and the scope of the literature. Section 2.1.3 introduces

2. SoS TextVis: A Survey of Surveys on Text Visualization

Table 2.1: A list of literature sources searched for text visualization surveys. We mainly use IEEE Xplore [6], the ACM Digital Library [7], and Google Scholar [8] to search for literature

Conferences & Journals	Papers
Springer	4
The Annual EuroVis Conference / Computer Graphics Forum	3
Wiley Online Library	3
IEEE Transactions on Visualization and Computer Graphics	2
El profesional de la información	1
Journal of Visual Languages & Computing	0
Information Visualization Journal	0
ACM Computing Surveys	0
Proceedings of the Annual Conference of the Alliance of Digital Humanities Organizations	0
Literary and Linguistic Computing	0
Digital Humanities Quarterly	0
Total	14

a previous survey of surveys and how our work differs from it. Section 2.1.4 presents our classification of the literature. Section 2.2 discusses and compares the surveys guided by the classification in section 2.1.4. Section 2.3 summarizes and discusses the future challenges reported within our collection. The article ends with conclusions and directions for future work.

2.1.1 Literature Search Methodology

Our search methodology is a variation of work by McNabb and Laramée [41], who collected a large number of surveys in the field of information visualization and visual analytics. We also consulted the survey of information visualization books by Rees and Laramée [42]. However, since the publication of the SoS, more recent surveys have been published in the field of text visualization and visual analytics, such as that of Kucher et al. [75], Jänicke et al. [26] and most recently Liu et al. [23].

In our search of the literature, we began by looking at each individual journal and conference in the data visualization community and performing a keyword search e.g. ‘Text Visualization Survey,’ ‘Text Taxonomy,’ ‘Text Visualization State-of-the-Art,’ or ‘Visual Text.’ All literature sources searched are shown in Table 2.1. As text visualization is of interest to other communities, we searched the digital humanities (DH) digital libraries for surveys, but were unable to find any in the main DH venues (shown in Table 2.1).

2.1.2 Survey Scope

In scope: 14 surveys were found and included in our text SoS. Those dedicated to text analysis and visualization approaches and those that explicitly feature a text visualization category were prioritised in the main literature classification, such as Sun et al. [76] and Liu et al. [77].

Out of scope: We restrict our literature to surveys that include a review of text visualization approaches, and we do not include surveys that review text mining techniques like summarization, such as Gupta and Lehal [78] or text clustering algorithms like Aggarwal and Zhai [79]. Survey papers that focus on text recognition, such as text detection and extraction by Jung et al. [80], are also out of the scope of this survey, as are text visualization books.

2.1.3 Related Work

McNabb and Laramee [41] made early progress in mapping the landscape of survey papers in information visualization. They present eight surveys which focus on analyzing and visualizing text data and classify the papers using an adapted information visualization pipeline by Card et al. [81]. They also identify three characteristics of classifications: the dimensions that each classification of survey adopts, the structure of the classification, and the type of mapping schema the survey incorporates. Kucher and Kerren [22] previously reviewed five surveys that focus on text visualization and compared the visualization taxonomies used in the reviews with their proposed taxonomy.

In our review, we aim to describe the existing surveys in more depth than McNabb and Laramee [41] and more breadth than Kucher and Kerren [22]. This text SoS includes more referenced text-focused surveys and book chapters than McNabb and Laramee [41] or Kucher and Kerren [22]. It is, to our knowledge, the first comprehensive survey of surveys (SoS) in text visualization.

2.1.4 Survey Classification

In order to compare each survey, we classify them into five categories. We study each survey's classification and categorization and group them based on the main focus themes found in each (see Table 2.2). In this way, the following five re-occurring themes were identified:

1. Data source: Surveys that derive their classification based on the underlying text source, e.g. single text or text stream.

2. SoS TextVis: A Survey of Surveys on Text Visualization

Table 2.2: Classification of our collection of 14 text visualization surveys. There are five categories: Data source, task analysis, multi-faceted, cross-disciplinary, and satellite-themed into which the literature is grouped.

Document-Centered	User Task Analysis	Multi-Faceted	Cross-Disciplinary	Satellite-Themed
Alencar et al. [82]	Cau and Cui [85]	Šilić and Bašić [21]	Jänicke et al. [24]	Sun et al. [76]
Gan et al. [83]	Federico et al. [86]	Wanner et al. [25]	Jänicke et al. [26]	Liu et al. [77]
Nualart-Vilaplana et al. [84]		Kucher and Kerren [22]		
		Kucher et al. [75]		
		Liu et al. [23]		

2. Task analysis: Surveys that mainly categorize their related literature based on the task analysis, e.g. showing similarities between texts.
3. Multi-faceted: Surveys that categorize related literature into multi-faceted classifications. In this case, the survey may propose multiple classifications based on a variety of characteristics, e.g. presentation and underlying data mining techniques.
4. Cross-disciplinary: Surveys that survey visualization techniques to support Digital Humanities.
5. Satellite-themed: Surveys that review existing information visualization literature. Also included are surveys that only incorporate text visualization as a sub-section within their classification.

2.2 Summary and Comparison of Surveys

This section discusses the surveys and provides recommendations. Table 2.2 shows the classification of the surveys, where each column represents a focus category and includes the corresponding surveys.

2.2.1 Document-Centered Surveys

The three surveys in this collection are by Alencar et al. [82], Gan et al. [83] and Nualart-Vilaplana et al. [84]. Their classifications are centered around document type, which involves classifying a given document –as the central theme – to a single document, a collection of documents, or a stream of text, etc.

Alencar et al. [82] review visual text analysis approaches. In their classification, there are two main categories. The first is *target input material of approaches*, either a single document (TagCrowd [87] and Wordle [88]) or a collection of text (Cartographic Maps [89], Galaxies [90], InfoSky [91] and Document Cards [92]). The second category is *the focus of the approaches*, such as showing relations (CiteSpace [93]), highlighting temporal changes (SparkClouds [94]) and visualizing query results (TileBars [95]). They describe each approach to obtain meaningful text models, how they extract information to produce representative visual designs, the user tasks supported, the interaction techniques applied and their strengths and limitations.

Gan et al. [83] present an overview of the concept of document visualization, the related research, and representative methods in each category of their hierarchical document classification. They classify the literature clearly based on the data source and do not consider representation.

Each main category of their classification contains detailed sub-classifications. The overview introduces several representative methods for each category. It also summarizes and compares each visual design based on four aspects:

1. Text characteristics depicted as word concordances, semantic relations, contents, or document relations.
2. Design principles satisfied, which refers to the Type by Task Taxonomy (TTT) that Shneiderman proposes [96]. He includes seven tasks: overview, zoom, filter, details-on-demand, relate, history, and extract.
3. Requirements for a document to suit this visual design, such as arbitrary text documents or sequence-based documents.
4. Main features such as interactivity and versatility (designing general visualization models for different tasks).

Nualart-Vilaplana et al. [84] examine 49 approaches to visualize textual data over a 19-year period spanning 1994–2013, in order to provide a classification of text visualization approaches. Similar to Gan et al. [83], Nualart-Vilaplana et al. [84] start their classification with the data source of documents. The classification comprises two main categories: Individual texts and collections of texts. In each category, there are heuristic subdivisions in order to un-

derstand and describe the graphs. The subdivision of the single texts and collections categories includes:

1. Sub-divisions for individual texts:
 - a) Whole or sub-sets: The visualization process includes the whole text or part of it.
 - b) Sequential or non-sequential: The visual layout preserves the same word sequence as that of the original text.
 - c) Discourse structure or syntactic structure: The visual design uses elements from discourse structure which refers to using actual parts of the text, enabling the viewer to read through visualization or syntactic structure using intrinsic elements of the text such as words and phrases.
 - d) Search: The imagery results from a search query.
 - e) Time: Text that changes over time.
2. Sub-divisions for collections of texts:
 - a) Items or Aggregations: The items of the collection used individually or there is some aggregation visualized.
 - b) Pure data or landscape: The text data in the collection is accompanied by graphical content.
 - c) Search: Same as above [\[1\]](#).
 - d) Time: Same as above [\[1\]](#).

2.2.2 User Task Analysis Surveys

In this category, we group surveys that mainly categorize their related literature based on user task analysis. There are two surveys in this category by Cau and Cui [\[85\]](#) and Federico et al. [\[86\]](#).

Cau and Cui [\[85\]](#) present a systematic review of existing text visualization techniques. The volume of the approaches cited is over 200. The overview classifies the approaches into two main categories: (1) Visualization and (2) exploration or interaction. They classify the literature in the visualization category based on the tasks of the visualization (what each is developed for), such as showing similarities, contents, and sentiment. For large document collections, the

review provides the most common exploration techniques which include distortion-based approaches and hierarchical document exploration approaches.

Federico et al. [86] survey interactive visualization approaches that support search and analysis of scientific articles and patents. They classify the visualization approaches according to two orthogonal aspects: data type and analysis tasks. There are four data types identified: Text, citation, authors, and meta-data. The analysis task breakdown [86] adopts the typology of data analysis tasks by Andrienko and Andrienko [97]. The four analysis tasks include elementary lookup and comparison, elementary relation seeking, synoptic tasks, and temporal patterns. Furthermore, the review introduces a breakdown of approaches that handle multiple data types.

2.2.3 Multi-Faceted Text Visualization Surveys

In this category, there are five surveys by Šilić and Bašić [21], Wanner et al. [25], Kucher and Kerren [22], Kucher et al. [75], and Liu et al. [23] that include multi-faceted classifications of text visualization approaches. We consider a survey as multi-faceted if it maps visual text approaches into multiple dimensions, such as tasks, interaction and presentation.

Šilić and Bašić [21] introduce three categorizations of visual approaches according to the visualization process: Data types, text representation, and temporal drawing, as shown in Figure 2.2. They base their classification on the underlying algorithms and data mining techniques, and provide four user interaction methodologies commonly used when exploring text datasets.

Šilić and Bašić [21] specify three data types: A collection of text, single text, and short intervals of a text stream. Additionally, the survey presents the most popular feature extraction methods used to represent text features as follows:

1. Bag-of-words methods extract text features by counting the term occurrences in the text.
2. Entity recognition aims to extract proper names of entities such as people, organizations, places, or countries.
3. Summarization methods shorten the text and present only the most relevant information.
4. Document structure parsing extracts structural information from text, such as titles, author names, and publication dates.

2. SoS TextVis: A Survey of Surveys on Text Visualization

Method name	Basic underlying methods	Data type	Temporal	Year	Ref.
<i>Sammon</i>	Sammon's mapping	C	-	1969	[27]
<i>Lin et al.</i>	SOM	C	-	1991	[28]
BEAD	FDP	C	-	1992	[29]
Galaxy of News	ARN	C	-	1994	[30]
SPIRE / IN-SPIRE	MDS, ALS, PCA, Clustering	C/S	-	1995	[17]
TOPIC ISLANDS	MDS, Wavelets	S	N/A	1998	[18]
VxInsight	FDP, Laplacian eigenvectors	C	-	1998	[31]
WEBSOM	SOM, Random Projections	C	-	1998	[32]
Starlight	TRUST	C	-	1999	[33]
FeatureLens	FP	C	+	2007	[40]
NewsRiver, LensRiver	FP	C	+	2007	[41]
Projection Explorer (PEX)	PROJCLUS, IDMAP, LSP, PCA	C	-	2007	[42]
SDV	PCA	S	N/A	2007	[14]
Temporal-PEX	IDMAP, LSP, DTW, CDM	C	+	2007	[43]
T-Scroll	GD, Special clustering	C	+	2007	[44]
<i>Benson et al.</i>	Agent-based clustering	SI	~	2008	[11]
FACT-Graph	GD	C	+	2008	[45]
<i>Petrović et al.</i>	CA	C	-	2009	[46]
Document Cards	Rectangle packing	S	N/A	2009	[47]
EventRiver	Clustering, 1D MDS	C	+	2009	[8]
MemeTracker	FP, Phrase clustering	C	+	2009	[16]
STORIES	GD, Term co-occurrence statistics	C	+	2009	[19]

Figure 2.2: Text visualization methods presented by Šilić and Bašić [21]. The table summarizes the methods, their underlying algorithms, the publication year, whether the method includes a temporal presentation or not, and the data type on which the method operates (C: Collection of text, S: Single text, SI: Short intervals). In the 'Temporal' column, if the method conveys time, it has (+), (-) if it does not, and (N/A) if not applicable. Reproduced with permission from the author, Knowledge-based and intelligent information and engineering systems; published by Springer, 2010.

5. Sentiment and affect analysis is used to identify and quantify the emotional aspects of the text.

The survey classifies the text visualization approaches into two categories:

1. Term trend approaches are based on the term frequency in the text. In such methods, feature selection is used to reduce the number of dimensions.
2. Semantic space approaches facilitate semantic methods to extract features of text(s). In most cases, feature vectors representing text are high-dimensional, so more advanced dimensionality reduction algorithms are used to map these features to 2D or 3D space.

The survey provides four exploration methodologies that help the user extract insight from the given data: brushing and linking, panning and zooming, focus-plus-context, and magic lenses.

Wanner et al. [25] take a step towards defining the concept of events within text streams by investigating the existing visual text event detection approaches and providing an event detection and exploration pipeline. An event in a text stream, as defined by Wanner et al. [25], is a valuable, unexpected and unique pattern extracted from the text. They classify 51 papers into different categories based on the event detection and exploration pipeline: text data sources, text processing methods, event detection methods, visualization methods, and supported analysis tasks. The survey also classifies the evaluation techniques applied in each paper.

Wanner et al. [25] derive twelve main data sources. Since 2010, micro-blogging has been the most common data source for visual event detection. In contrast, there is only one paper that detects and visualizes events in online customer feedback. Tables 4 and 6 in [25] show the visualization approaches used in the investigated literature and these approaches along with the event detection techniques applied. It can be observed that all of the clustering based techniques are presented mainly using the river metaphor. Most of the papers in Wanner et al. [25] rely on use cases for evaluation (35 out of 51). On the other hand, only four papers present user studies. They suggest that more involvement from end users is encouraged.

Kucher and Kerren [22] present a visual survey of text visualization techniques. They classify text visualization into five top-level categories, shown in Figure 2.3:

1. Analytic tasks include the techniques that support high-level analytic tasks.
2. Visualization tasks include techniques that support lower-level representation and interaction tasks.
3. Domain describes the techniques that are developed for a specific application.
4. Data consists of two subcategories, source and properties, that describe the data source and the special properties of data used by the techniques.
5. Visualization contains three subcategories to describe the properties of visual representations: dimensionality, representation, and alignment.

Liu et al. [23] present a survey analyzing 263 visualization papers and 4346 data mining papers to extract about 300 concepts in both fields. The survey also analyzes the tasks they

2. SoS TextVis: A Survey of Surveys on Text Visualization



Figure 2.3: The classification of text visualization techniques used in the survey by Kucher and Kerren [22]. Reproduced with permission from the author, IEEE Pacific Visualization Symposium; published by IEEE, 2015.

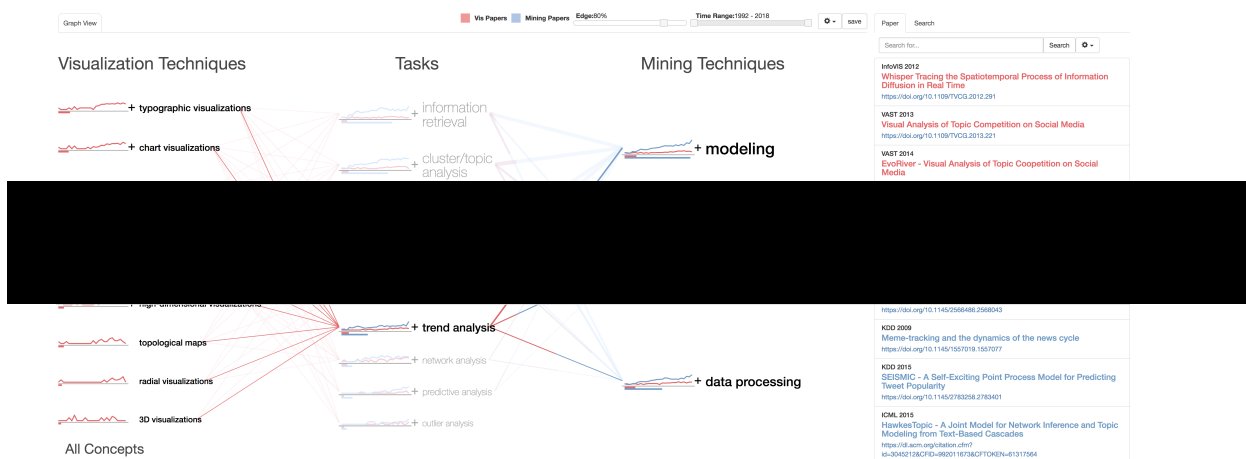


Figure 2.4: A snapshot of the recent web-based visualization tool developed by Liu et al. [23]. The visualization and text mining techniques are connected by their shared analysis tasks. On the right, a list of the corresponding literature that matches user preference. The red and blue colors correspond to the fields of visualization and data mining respectively.

support. Three multi-level taxonomies are provided for text visualization, data mining, and analysis tasks. The paper contributes an interactive web-based visualization of the literature and taxonomies, enabling the user to interactively find the co-occurrence relationship between the concepts and identify potential research gaps (Figure 2.4).

This section also includes a survey by Kucher et al. [75] which generally uses the same taxonomy as [22] with a focus on techniques that visualize sentiment and opinions from text data.

2.2.4 Cross-Disciplinary Text Visualization Surveys

This section presents surveys that support Digital Humanities tasks. There are two surveys which review the literature in the field of visualization that support close and distant reading of textual data by Jänicke et al. [24] and an extended version of this by Jänicke et al. [26]

Jänicke et al. [24] provide an overview of the last ten years of advancements in the field of visualization that support Digital Humanities tasks. They classify the literature based on the representation: whether it supports close reading or distant reading as proposed by Moretti [98]. Close reading attempts to provide direct access to the original textual content in its sequential order, while distant reading does not retain the source text and provides an overview of its global features. The large availability of digital texts introduced by web portals such as Google Books [99] opens up new avenues for close reading techniques and collaborative tools.

Jänicke et al. [24] classify the methods found in their collection based on task supported (close, distant or combined) reading. Furthermore, the review classifies each paper based on the underlying source text (single text, parallel, and corpus) with an extended subdivision in each category. Figure 2.5 shows a summary table of the proposed classification. In the following, we summarize the classification proposed.

Close Reading Techniques: There are a number of techniques applied in the 46 papers included in the research paper collection that provide visual support for close reading visualization:

- Color is used to show a great variety of features, e.g. classification, similarity or importance.
- Font size is also used to convey text features, e.g. word frequency or significance.
- Glyphs are used to present aspects of the text that are difficult to express using other techniques and are mostly used in poems to draw phonetic units.
- Connections help illustrate the relationship between text entities, e.g. to show subsequent words to track variation among various text editions or convey sentence structure.

Distant Reading Techniques: 81 research papers in the collection provide an abstract distant reading view of text. There are several approaches used to visualize summarized information:

2. SoS TextVis: A Survey of Surveys on Text Visualization

			Close Reading					Distant Reading						
			Plain	Color	Font size	Glyphs	Connections	Structure	Heat maps	Tag clouds	Maps	Timelines	Graphs	Miscellaneous
Single Text Analysis	enhanced text views	[Pie10], [CGM*12], [Pie13], [GWH14]				x								
		[PSA*06], [CTA*13], [Ben14], [BJ14]		x										
		[ARLC*13]				x	x							
	both	[WMN*14]		x	x									
		[VCPK09], [BGHJ*14], [KJW*14]		x				x		x				
		[WJ13b], [CMLM14], [KZ14]		x			x							
		[Cay05]		x				x						
		[CDP*07]		x					x					
		[WV08]		x									x	
	abstract text views	[MFM13]		x									x	
		[RSDCD*13]		x										x
		[KO07], [FS11], [CTA*13], [OKK13], [Ben14]								x				
		[Pie05]							x					
Text Analysis	section alignments	[PBD14]											x	
		[WH11], [HKTK14]		x										
		[Cor13], [WJ13b]		x			x							
		[JRS*09]		x				x						
Corpus Analysis	relationships between textual entities	[GZ12], [RFH14]		x									x	
		[MH13]		x					x				x	
		[AKV*14]		x						x				
	social networks	[Cob05], [CSV08], [BDF*10], [RD10], [BHW11], [Kle12], [Bool3], [KOTM13], [Töt13], [Pet14]												x
		[KLB14]		x										x
	space and time	[JHSS12], [JW13], [DNCM14], [GDME*14], [ÖML14]										x	x	
		[Wea08]							x		x	x		
		[BPB10]			x						x	x		
		[DWS*12]		x						x	x	x		
	space	[HACQ14]											x	
		[MBL*06], [DFM*08], [Tra09], [GH11b], [EJ14]												
	time	[KBK11], [ARR*12], [LWW*13]											x	
		[CLF*11], [CLWW14]									x		x	
		[HSC08]		x									x	x
		[DWS*12]		x							x	x	x	
		[ESK14]									x		x	x
				x								x		

Figure 2.5: Hierarchical classification of research papers reviewed by Jänicke et al. [24]. At the top-right, the intended tasks supported by the visual design and the techniques implemented. On the left, the rows show the paper classification organization. Reproduced with permission from the author, Eurographics Conference on Visualization (EuroVis)-STARS; published by The Eurographics Association, 2015.

- Structural overviews illustrate the hierarchy of document or collection of documents.
- Heat maps are usually used to show textual patterns such as similarities.
- Tag clouds encode word occurrence frequency within a text using variable font size.
- Maps display geospatial information contained in a text.

- Timelines are used to visualize text that conveys temporal information. Such a technique could use the text’s metadata and support the temporal analysis of the use of a word over time.
- Graphs usually use nodes and edges to visualize certain structural features of a text corpus.
- Miscellaneous methods are used to explore specific aspects within text interactively.

Techniques for Combining Close and Distant Reading: There are some visual designs that provide both close and distant reading by preserving direct access to the source text. The 26 papers in the collection that use hybrid techniques and serve this purpose are grouped into three categories:

- Top-down approaches implement the information seeking mantra ‘overview first, zoom and filter, details-on-demand’ [96]. Initially, an overview of the textual data is shown, and the user then interacts with the graphics by filtering or zooming, and finally, clicking on the interesting sub-set to obtain details-on-demand.
- Bottom-up methods start with the desired text or part of it and then generate an overview layout which relates to the given section or text.
- Top-down and bottom-up methods provide a mechanism of switching between close (text view) and distant reading (structural overview).

Jänicke et al. extended the survey in 2017 [26]. In terms of classification, they add a categorization of the text analysis techniques which includes 22 more papers than the original. The text analysis taxonomy has five main categories: named entities, topics, similar patterns, text of interest, and corpus analysis. They also extend the discussion of collaboration experiences and future challenges.

2.2.5 Satellite-Themed Text Visualization Surveys

Two surveys review the broader information visualization literature and consider text visualization as a sub-section in their overview classification. These are by Sun et al. [76] and Liu et al. [77]. This contrasts with the focus on text only found in the previous surveys.

Sun et al. [76] review the recent developments in the field of visual analytics and propose a 2D classification they call Analytics Space. The first dimension is an applications category

which includes space & time, multivariate, text, graph and others. The second dimension is motivated by the visual analytics model proposed by Keim et al. [100] which includes visual mapping, model-based analysis, and user interaction. With respect to text classification, Sun et al. provide two categories to organize methods that process text data. The first includes topic-based approaches which mostly leverage algorithms from Natural Language Processing (NLP). In this category, the methods that involve topics or event extraction from the text data are included e.g. TextFlow [101] and EventReader [102]. The second category is feature-based approaches which use text features to visualize text e.g. Wordle [88] and FacetAtlas [103].

Similarly, Liu et al. [77] include a category of application within their information visualization taxonomy that includes four categories: empirical methodologies, interactions, frameworks, and application. In the application category, they [77] include four applications to classify: graph, text, map and multivariate data visual designs. There are two categories assigned to the text visualization collection. The first comprises applications that visualize static textual information. In this category, they discuss and classify techniques that visualize the time-invariant content of the document(s). The second category is the visualization of dynamic textual information, which incorporates designs that visualize temporal changes within a document or collection of documents. In both categories, Liu et al. group the techniques, similarly to Sun et al. [76], into two categories: feature-based and topic-based approaches.

2.2.6 Survey Recommendations

Multi-faceted surveys serve as useful starting points. Wanner et al. [25] and Kucher and Kerren [22] provide well-crafted taxonomies. The former provides a guide for researchers interested in extracting events from text. The taxonomy itself is not complicated and is built on the literature they collected. It also provides a classification of the evaluation techniques that are used in each approach. Wanner et al. identify trends, research directions, and untouched areas in the discussion of their taxonomy which may also be beneficial for readers.

On the other hand, the Kucher and Kerren [22] classification covers many aspects of text visual analytics. We recommend it for researchers who would like to explore or contribute to the field of text visualization. It provides the most comprehensive and up-to-date summary of text visualization [85] of these surveys. The survey's associated text visualization browser enables the user to explore and filter the collection based on classification.

Liu et al. [23] try to bridge the gap between the text mining and visualization approaches. Their web-based, interactive visualization is a useful tool that integrates both text mining and

2.2. Summary and Comparison of Surveys



Figure 2.6: A bi-gram word cloud representation of the surveys by Wanner et al. [25], Kucher and Kerren [22], Jänicke et al. [26], and Liu et al. [23] to illustrate the vocabulary used in each one. We apply the word clouds using the script by Müller [27].

text visualization with the analysis tasks abstraction.

For researchers interested in the digital humanities we recommend Jänicke et al. [26] as they provide a comprehensive overview and discussion of text visualization techniques that serve humanities tasks.

Figure 2.6 illustrates the most frequently occurring bi-grams. In the preprocessing phase, the stopwords are removed and then lemmatization is applied to consolidate inflected words. After that, the word clouds are created using the script by Müller [27]. The figure illustrates the theme of each paper. The survey of Wanner et al. [25] shows (Figure 2.6a) a significant use of words such as detection and event in a context like ‘event detection’, ‘detect event’, and ‘text event’. Wanner et al. was the first survey to consider microblogging as a data source which can be seen in the figure as well. In Kucher and Kerren [22], Figure 2.6b, multiple term pairs are used often such as, ‘text visualization’, ‘visualization technique’, and unsurprisingly ‘survey browser’, and ‘proposed taxonomy’. On the other hand, there is an obvious change of vocabulary in Jänicke et al. [26] (Figure 2.6c) which discusses the approaches within a different context. There is more frequent occurrence of bi-grams that convey digital humanities goals such as ‘distant reading’, ‘close reading’, and ‘digital humanities’. Figure 2.6d shows the bi-grams that most often appear in the survey by Liu et al. [23]. The words ‘mining’, ‘task’, and

‘visualization’ are the most shared among the term pairs; clearly the theme of the survey is to fill the gap between visualization and mining literature via the abstracted analysis tasks.

Figure 2.7 provides a list of the top bi-grams using TF-IDF (Term Frequency - Inverse Document Frequency) to measure the significance of each pair across the collection [104]. We follow the same preprocessing steps to generate the word cloud in Figure 2.6, although we use the weighting factor of each pair in the corresponding survey with respect to the other surveys. The topics ‘document collect’, ‘document visual’, and ‘text collect’ are featured in the three ‘Data Source’ classified surveys, Alencar et al. [82], Gan et al. [83], and Nualart-Vilaplana et al. [84] respectively, excepting the Cau and Cui [85] survey which features the topic ‘document collect’ quite often. The exception is the Cau and Cui [85] survey, which features the topic ‘document similar’, supporting our classification. However, the top words of the second survey in the task analysis group of Federico et al. [86] show a topic pattern that does not align with our classification, e.g. ‘data type’, and ‘node link’. The topic ‘event detect’ is featured highly in the survey by Wanner et al. [25], as expected. Other taxonomy words appear as well, such as ‘text process’ and ‘data source’, which supports the contention that they provide multiple classifications of the approaches, and this applies to the majority of the multi-faceted surveys.

We developed a web-based parallel coordinates plot to interactively explore the vocabulary of the surveys and to further examine correlation and overlap between them [28]. Figure 2.8 shows the most common vocabulary (>1%) between the two surveys from the cross-disciplinary group (Jänicke et al. [24] and Jänicke et al. [26]). It is clear that the two surveys complement each other: they share distinctive words like ‘human’-ity, ‘digit’-al, ‘read’-ing, and ‘close’.

The illustration in Figure 2.9 shows the number of papers reviewed by each survey. A clear increase in reviewed approaches between 2012 and 2016 can be seen, cited mainly by Kucher and Kerren [22] and Liu et al. [23]. The satellite-themed surveys share the same time span of literature (2009 to 2013), and the two cross-disciplinary surveys share almost the same literature, although the later survey covers two additional years (2015 and 2016) which comprise 30 extra papers.

2.3 Discussion of Future Challenges

This section summarizes the future challenges identified in the collection. Table 2.3 lists these challenges along with the surveys. The McNabb and Laramée survey [41] has been added to

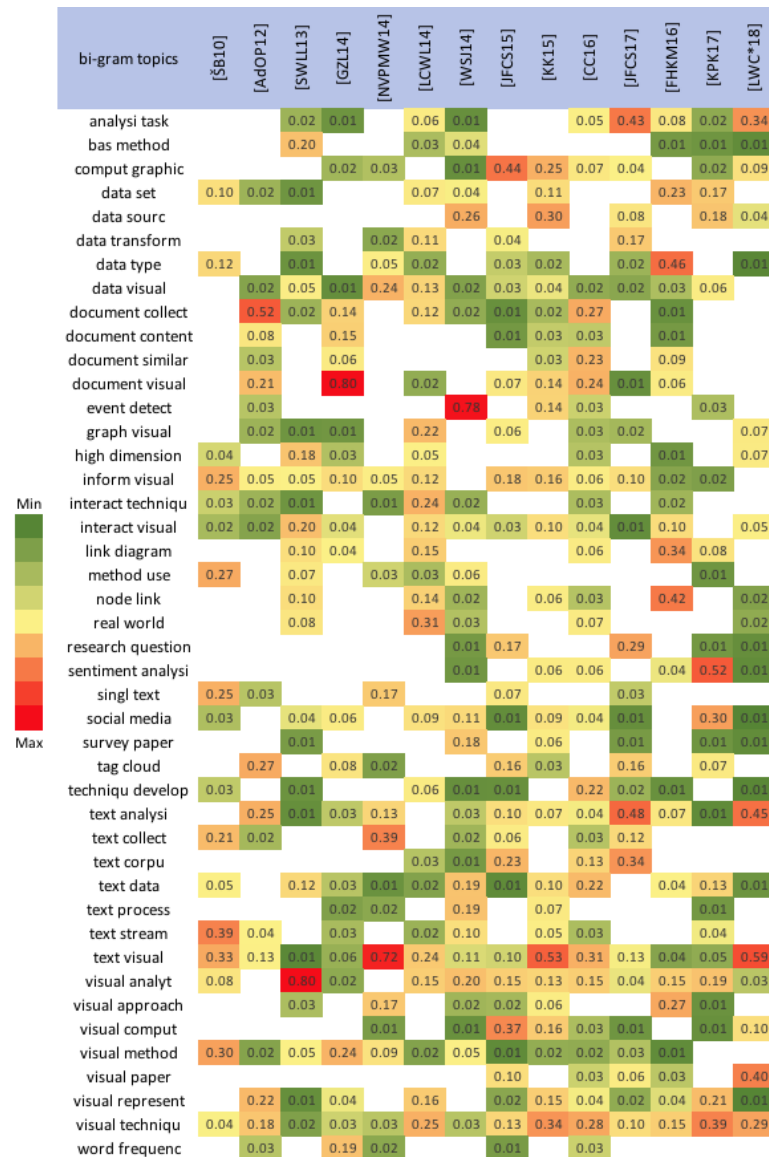


Figure 2.7: A list of the most common bi-grams extracted from the collective surveys. The color mapping of the cells indicates the weights of the corresponding bi-gram.

identify the overlapping challenges reported by them. We identify four unique challenges not reported by McNabb and Laramee since their challenges are derived from a wider perspective. These challenges are adopting advanced text mining techniques, lacking cognitive and/or psychological analysis, lacking clear boundaries of concepts, and the need for a collaboration framework between multidisciplinary scholars.

Nine challenges are common to two or more surveys. Federico et al. [86] identify 10

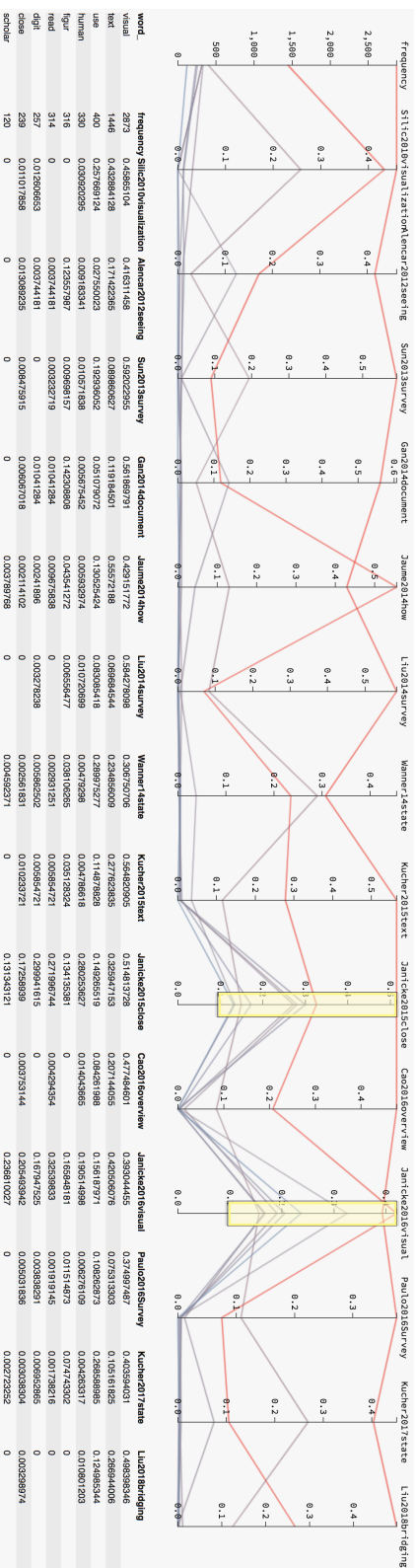


Figure 2.8: A snapshot of the web-based parallel coordinates plot developed to explore the vocabulary of the surveys interactively. Each vertical line (coordinate) represents a survey, except The first coordinate on the left corresponds to the occurrence frequency. Each polyline represents a word and the intersection between the polylines and the vertical coordinates depicts the word weights in that survey. Here, the user selects the most common vocabulary between the two surveys from the cross-disciplinary group [28]

2.3. Discussion of Future Challenges

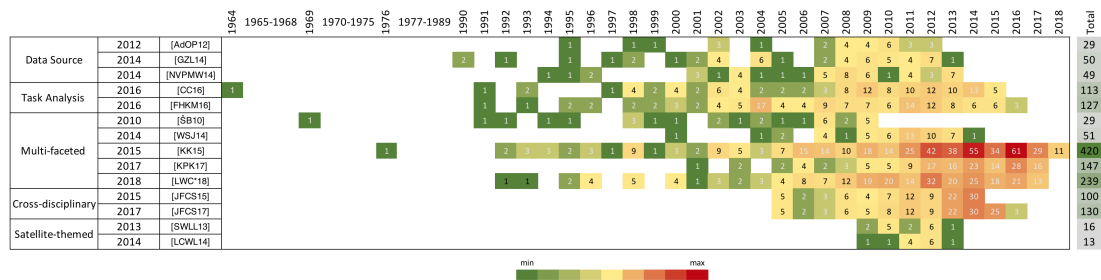


Figure 2.9: Visualization of the number of approaches reviewed by the surveys. Each row represents a survey and each cell represents the number of references in the corresponding year (columns). The color mapping of the cells indicates the number of approaches cited within each survey in the corresponding year. The last column shows the total methods each survey includes.

Table 2.3: Summary of the future challenges reported in our collection of surveys. This list contains the common challenges among the text visualization surveys.

Future Challenges	SoS	[41]	[21]	[82]	[76]	[83]	[84]	[25]	[77]	[22]	[24]	[86]	[85]	[75]	[26]	[23]	Total
Scalability	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	9
Lacking in-depth/effective quantitative or qualitative evaluation	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	9
Adopting advanced linguistics techniques			✓	✓	✓	✓	✓	✓				✓				✓	8
Natural language ambiguity and uncertainty	✓				✓				✓		✓	✓				✓	6
Lacking user interactivity that support the analysis tasks	✓			✓				✓				✓				✓	5
Designing general models for different tasks (versatility)	✓	✓		✓												✓	4
Multidisciplinary framework											✓	✓			✓		3
Lacking cognitive and/or psychological analysis			✓	✓													2
Lacking well-defined general concepts						✓	✓										2

future challenges, three of which are unique. The oldest two surveys by Šilić and Bašić [21] and Alancer et al. [82] and the two multi-faceted surveys Wanneret al. [25] and Kucher and Kerren [22] do not feature unique future challenges. This reflects the fact that these surveys do not focus on a specific discipline or task. On the other hand, Jänicke et al. [24,26] identify two unique future challenges, indicating that they feature a distinctive theme.

One of the most common future challenges is the need for in-depth, effective quantitative or qualitative evaluation, mentioned in nine surveys of the collection. Most of the surveys report a lack of in-depth evaluation of the proposed approaches. Advanced and formal evaluation provide valuable user feedback and facilitate identification of potential system problems [76]. Wanner et al. [25] expect a rise in user study evaluation to verify the strengths and weaknesses of novel visual designs. We believe that further research in the effectiveness of text visualization evaluation would be fruitful.

One of the most widely reported challenges is scalability and handling huge volumes of data. Approaches usually use various aggregations, projections, or multiple views to address this issue. However, further investigation is needed to validate the usefulness and effectiveness of such approaches, especially for scientific literature [86]. This challenge is generally associated with that of adopting advanced text mining and linguistics algorithms.

As natural language often comes with ambiguity, uncertainty, and/or errors, five surveys report this as a challenging task. Many approaches do not consider uncertainty and this could affect the analysis results. Appropriate uncertainty visualization approaches should therefore be developed [86]. The Text Visualization Browser [22] contains 25 articles that include visualization of uncertainty and ambiguity, 12 of which were published in 2016 and 2017. Jänicke et al. [24,26] specifically consider temporal and geospatial uncertainty in literature as an important future challenge. Uncertainty modeling and visualization research is expected to increase.

Another common challenge is the lack of user interaction to support the analysis process. Many approaches represent the outcome of the analysis process visually and fail to provide a means for the user to steer the underlying algorithms to further analyze the data [25,86]. We expect future work in the interactivity of visual analytics.

Jänicke et al. [24,26] and Federico et al. [86] reported multidisciplinary frameworks as a challenging research topic. They suggest a systematic approach that guides and steers the work between scientists and domain experts. The former two surveys by Jänicke et al. summarize the experiences reported regarding collaboration between visualization scientists and humanities scholars.

A lack of cognitive and/or psychological analysis that verifies how users perceive and preserve information and incorporate it into the decision-making process is a challenging task reported in two surveys: Šilić and Bašić [21] and Alencar et al. [82].

Many visual designs are targeted toward a specific point and do not support multiple tasks. Gan et al. [83] believe that it is essential to design general visualization models for different tasks, and Alencar et al. [82] confirm that it is challenging to approach a problem without a domain-specific solution. However, users might have different goals or needs and the visual design should accommodate that.

In the text visualization community, experts have always faced the challenge of an ill-defined concept of ‘event’ and other general elements of textual data [25]. Such a problem may distract efforts for improvement. Nualart–Vilaplana et al. [84] believe that the boundaries of the discipline in data visualization are not yet sufficiently well-defined.

Since the surveys vary in terms of global goals and targets, there are specific challenges reported within a given context. Jänicke et al. note the lack of visualization approaches that represent a transposition of textual entities on all text hierarchy levels using close and distant reading. Federico et al. [86] expect an increase in approaches that integrate citation analysis and other text mining techniques such as sentiment analysis to understand citations and enrich analysis. Nualart–Vilaplana et al. [84] pose a question about the long-term availability of the tools, arguing that if the tool is no longer available and not maintained for use, it may not be effective.

2.4 Chapter Summary

In this text visualization SoS, we presented an extended meta-survey of the reviews of literature in the field of text visualization, classifying the survey collection based on five themes. Each survey classification and its features were summarized to facilitate comparisons between the surveys. Recommendations for researchers and examination of potential future trends in the field of text visualization were provided in the light of the findings, discussed and compared with reference to challenges in the field reported in the collection. This comprehensive review offers an important and unique starting point for both newcomers and experienced researchers in text visualization.

Chapter 3

Background

Contents

3.1 Related Work of Visible NLP	45
3.2 Related Work of AlignVis	49
3.3 Related Work of TransVis	51
3.4 Chapter Summary	57

This chapter introduces the background research of our thesis contributions.

3.1 Related Work of Visible NLP

In this section, we summarize the related literature that corresponds to the second research objective [Ob2]. We include literature that incorporates some visible NLP aspects within their approaches. We also summarize the most common and recent text visualization approaches to re-view the visual design space used to represent text embeddings.

Explainable machine and deep learning models: Several solutions are proposed to solve the lack of transparency in machine and deep learning models. They are often referred to as explainable artificial intelligence (XAI) [118-120]. Hohman et al. [121] present the state-of-the-art of deep learning visualization using an interrogative framework which includes: Why, Who, What, How, When, and Where.

Chatzimparmpas et al. [122] introduce a survey of surveys on the use of visualization for interpreting machine learning models that are designed to clarify and help understanding of the

3. Background

Reference	Derived Data			Visual Pipeline Encoding				
	Vocabulary meas.	Statistical meas.	Syntax meas.	Visible Segmentation	Visible Tokenization	Visible Stopwords	Visible Normalization	Visible Embeddings
Church and Helfman [105]	x							
Wattenberg [106]	x							
Keim and Oelke [9]	x	x	x					x
Wattenberg and Viégas [107]	x							x
Collins et al. (Parallel Tag Clouds) [108]	x	x						x
Collins et al. (DocuBurst) [109]	x	x	x					x
Van Ham et al. [110]	x	x	x					x
Jänicke et al. [111]	x	x						x
Oelke et al. [112]	x	x						x
Jänicke et al. [113]	x	x						x
Geng et al. [64]	x	x						x
Riehmann et al. [114]	x	x						
Abdul-Rahman et al. [115]	x	x		x	x	x		
Jänicke and Wrisley [116]	x	x				x		x
Hu et al. [117]	x	x						x
TransVis [2]	x	x					x	x
AlignVis [1]	x	x				x	x	x

Table 3.1: A summary of related work. Each paper is characterized by a two-level hierarchy [9]: the derived data that each approach generates and represents and the supported visual encodings in the NLP pipeline. Our approach makes the entire NLP pipeline visible.

intermediate process and layers of such techniques. Multiple surveys focus on the interpretation of ML models with the use of visualization, such as [123, 124]. Some of these designs are implemented for educational purposes, such as TensorFlow Playground [125], while others are implemented to cluster, classify and understand reduced-dimensionality vector space, such as [126, 127]. Other approaches incorporate visualization to understand and interpret machine learning models, such as [128-132]. Zhang et al. [133] propose a visual and interactive framework for interpreting, comparing and debugging machine learning models. Ribeiro et al. [134] present a methodology and visual tool that tests individual capabilities of NLP models using different test types. However, most of these advanced techniques are difficult to implement and understand for domain scholars.

The following section first reviews the related visual approaches that visually integrate and facilitate NLP functions and enable user interference in order to update the analysis process. Then, we review the research space for the most common visual representations used to depict text embeddings.

Related visible NLP: Some text visualization approaches incorporate NLP functions to pre-process text data and produce embeddings that are used in visual interfaces.

Abdul-Rahman et al. [115] incorporate tools that enable the user to segment and tokenize text based on multiple presets. They illustrate the results with a dot plot graph to depict text re-use patterns.

Jänicke and Wrisley [116] propose a visual alignment approach to align versions of medieval poetry, providing an overview alignment using a bipartite graph. The user can investigate an alignment from the bipartite graph using an intermediate level they call “meso reading”,

which illustrates the aligned pairs of lines between two text versions and provides a preview that annotates stopwords and encodes the frequency of reused words using saturation.

AlignVis [1] enables the user to manipulate the alignment process via multiple options such as stopwords removal and normalization. AlignVis visually encodes the alignment between the source and target text using a bipartite graph, and encodes the confidence value of the similarity measurement using the color of the text segments and edges.

In contrast to our work, very few previous approaches explicitly demonstrate the processes they undertake and enable the user to explicitly manipulate intermediate NLP steps to observe the effects of changes.

Visual design space: we summarize some of the most common and recent text visualization approaches to represent the visual encodings to depict text embeddings.

Dot plot graph: The dot plot [135] is a 2D matrix plot used to detect similarities and reuse between two sequences. Abdul-Rahman et al. [115] incorporate tools that enable the user to segment and tokenize the text based on multiple presets, integrating a dot plot to illustrate different text alignments patterns. Other approaches utilize dot plot and facilitate colors to encode the embeddings [105,111].

Storylines and stream graph: Storylines and stream graphs are another common visual design to depict interrelationships between text entities and detect patterns above the level of individual terms [136]. TRAViz [113] facilitates a stream graph to enable the user to investigate the variation between texts, with the number of lines and the size of the font indicating the frequency of word reuse. Silvia et al. [136] adopt a storyline design to illustrate interactions between entities in a story and explore how entity relationships evolve over time. The word tree [107] is another kind of directed stream graph used to illustrate the occurrence of terms in a text. Alharbi et al. [2] similarly propose an overview of translations that connects aligned segments using curved lines.

Bipartite graph: A number of approaches incorporate bipartite graphs to illustrate text alignment and communicate the comparison task. Riehmman et al. [114], for example, combine bipartite graph and pixel-based representations to detect plagiarized text passages in PhD theses. Jänicke and Wrisley [116] provide an overview alignment using a bipartite graph. Abdul-Rahman et al. [115] also incorporate an interactive bipartite graph in which the user can define

3. Background

a subset to examine using a dot plot design, and Alharbi et al. [1] adapt a bipartite graph to illustrate the alignment results and color each edge based on the confidence value to guide the domain scholar in refining the alignment.

Heatmap and pixel-based graphs: Geng et al. [64] implement the vector space model to explore patterns of variation between different translations of Shakespeare’s *Othello*, with the term-document matrix visualized using a heatmap where the color encodes the term frequency. Keim and Oelke [9] extract different statistics, such as average word and sentence length, and vocabulary measures, such as word frequency and lexical diversity, and encode them using a pixel-based graph.

Parallel coordinates A parallel coordinates design is considered a useful technique to explore multi-variate data [137]. The embeddings are translated into each dimension and poly-lines connect the corresponding entities. Parallel Tag Clouds [108] integrate the font size and color to visually encode embeddings. Geng et al. [64] implement parallel coordinates of the similarity measures in order to depict the variation between translations. Alharbi et al. [2] incorporate parallel line charts to illustrate the terms embeddings that are generated to convey the variation between different texts.

Word Clouds: Word clouds depict tokens that occur frequently in the source text [94, 138, 139]. Parallel Tag Clouds [108] use the font size mapped to word significance in a document collection which is arranged vertically. Oelke et al. [112] propose a glyph-based visualization to illustrate multivariate properties, integrating a word cloud approach to show the most descriptive terms of each topic cluster, and integrating encodings to visualize the relevancy of each topic to a specific class and determine the extent to which a topic is discriminative for a class.

Network graphs: Several approaches attempt to present the word relations in a text, with embeddings usually encoded using color, edge thickness and font size. DocuBurst [109] integrates WordNet [140] to generate vocabulary measures, depicting the word relations using radial graph layouts. Phrase Nets [110] and SentenTree [117] use a directed node-link graph. The edges encode the strength of the relation between the connected words and the size of the words represents the word frequency. Wattenberg [106] propose the arc diagram which visually connects repeated substrings using translucent arcs.

Reference	Year	Number of Aligned Text simultaneously	Close reading	Distant reading	Source of text studied
Melamed [141]	1989	2	bipartite graph	-	The Bible translations
Cairo [142]	2000	2	juxtapositioned word-to-word corresponds	-	Italian-English corpus
Tufiş [143]	2006	2	bipartite graph	-	Romanian-English parallel corpus
Yawat [144]	2008	2	juxtapositioned word-to-word corresponds	dot plot matrix	German-English corpus
SWIFT Aligner [145]	2014	2	bipartite graph	-	French-English corpus
Jänicke et al. [111]	2014	2	bipartite graph, variant graph	heat-map, dot plot	The Bible translations
iTeal [116]	2017	Multiple	bipartite graph, variant graph	juxtapositioned alignment map	Literature
ViTA [146]	2017	2	bipartite graph	dot plot graph	Literature

Table 3.2: A summary table of related work and paper characteristics included in the computational and visual alignment section. The dashes (-) in the distant reading column indicate that the corresponding reference does not feature distant reading.

Table 3.1 summarizes the representative approaches classified into a two-level hierarchy. The first level classifies the approaches based on the derived data. We adopt the Keim and Oelke [9] literary analysis classification: statistical measures, vocabulary measures, and syntax measures. Vocabulary measures include word frequency and word co-occurrence. Statistical measures include global aggregation such as average word length and occurrence proportion. Syntax measures comprise the utilization of a syntax tree of the texts. The second level summarizes the work based on the supported visual encoding of our adapted pipeline. We categorize the documents based on the existence of interactive means to explicitly modify the task by the user. The representative approaches include common visual designs for texts and designs that support parallel texts.

Our approach is different from a fundamental perspective as it enables the user to explicitly manipulate the parameters of the NLP pipeline process. At each stage, the user can explicitly observe the effect of their changes in each process. The design is applied to a text similarity application to explore the effect on the embeddings that are controlled by user preferences.

3.2 Related Work of AlignVis

This section summarize the literature that corresponds to the third research objective [Ob2]. The related work is divided into four sections. The first section is summarized in Table 3.2.

Computational and Visual Alignment: Jänicke and Wrisley [116] and Abdul-Rahman et al. [146] integrate similarity measurements to detect alignment between texts. Jänicke and Wrisley introduce an interactive visual analytics tool (iTeal) that facilitates computational alignment between multiple text editions, and provide imagery for different hierarchy levels (entire text, lines and words). Abdul-Rahman et al. [146] propose a web-based visual analytics tool (ViTA)

enabling domain experts to interfere in the text alignment pipeline.

Further, translation studies scholars use different off-the-shelf tools [147–149] to segment and align parallel texts in practice. Most of these tools utilize a user interface which enables the user to choose the source and target texts and then view the alignment results in tabular form. They also enable the user to perform certain functions to post-edit the segments and alignments, such as splitting, merging and deletion.

Multiple approaches offer visualizations of the pre-defined alignments and support for manual annotation, such as Melamed [141], Ahrenberg et al. [150], Yawat [144], Tufiş [143], and SWIFT Aligner [145]. Other approaches, such as Cairo [142], align corresponding words and do not allow the user to post-edit the alignments. Most of these approaches provide word-based alignment and use a simple bipartite graph to visualize the links between words.

LF-Aligner [151] is a standard tool commonly used to align translations. It supports multiple languages and parallel translations. A comparison between LF-Aligner and AlignVis is provided in Section 5.5.3, and a comparison between our design and LF-Aligner, ViTA and iTeal is given in Section 5.5.2.

AlignVis is different from these approaches since it combines a visual design that enables multiple alignments simultaneously and enables user interference and modification of the result. It also allows the user to test the result based on various similarity metrics.

Text Re-use and Plagiarism Detection: Several approaches have been developed to detect commonality between texts. Jankowska et al. [152] use n-grams to generate a relative n-grams signature to compare multiple texts against a base text. Jänicke et al. [111] introduce multiple visual designs such as heat-map display to depict text re-use patterns in English Bible translations, using a 2D dot plot to show text re-use patterns between two texts. Abdul-Rahman et al. [146] use a 2D similarity metrics plot to facilitate the discovery of text re-use patterns. The Versioning Machine [153] and JuxtaCommons [154] are digital humanities tools which visualize corresponding text fragments using color highlights and links.

Similarly, visualization techniques have been used to facilitate detection of plagiarism between texts [114, 155–159].

Most of the work presented in text re-use and plagiarism present a visual design that does not incorporate the user’s knowledge in the alignment process and does not enable the user to test different similarity measurement methods. The purpose of these tools is mainly to indicate repeated text in a binary fashion.

References	Source of text studied	Max number of documents viewable simultaneously		Visual design of distant reading view	Languages
		Close Reading	Distant Reading		
Ribler and Abrams [155]	Programming codes	-	Arbitrary	Patterngram	English
Monroy <i>et al.</i> [167]	Literature (Don Quixote)	1	5	Multiple bar charts	Spanish
Schreibman <i>et al.</i> [153]	Poetry	Arbitrary	-	-	English
Jong <i>et al.</i> [168]	Unspecified documents	9	14	Multiple views+pixel-based visualization	English
Collins <i>et al.</i> [108]	U.S. Circuit Court Decisions	1	13	Parallel coordinates+tag cloud	English
Büchler <i>et al.</i> [169]	Ancient Greek text	3	-	-	Greek
Welsh and Hooper [170]	Newton Alchemical corpus	2	-	-	English
Geng <i>et al.</i> [58]	Literature (Othello)	8	8	Parallel coordinates	German
Behrisch <i>et al.</i> [171]	News	2	33	Heatmap matrix	English
Howell <i>et al.</i> [172]	Literature (The Secret Scripture)	2	-	-	English
Jänicke <i>et al.</i> [111]	The Bible translations	2, 7	2	Text Re-use grid, Dot Plot view	English
Jänicke <i>et al.</i> [173]	The Bible translations	-	24	Variant graphs	English
Geng <i>et al.</i> [64]	Literature (Othello)	10	10	Parallel coordinates, heat maps, scatter plots	German
Riehmann <i>et al.</i> [114]	PhD theses and Wikis documents	2	Not-specified	Slope graph, glyph-based visualization	English
Asokarajan <i>et al.</i> [174]	Classical Latin texts	-	22	Multiple views+pixel-based visualization, dot plot	English
Cheesman <i>et al.</i> [56]	Literature (Othello)	1	2, 40	Juxtaposed text versions, stylometrics diagram	German
Silvia <i>et al.</i> [136]	Classical Latin texts	-	12	Storyline visualization	Latin
Jänicke <i>et al.</i> [175]	Medieval texts	2	2	Juxtaposed text versions	French
Abdul-Rahman <i>et al.</i> [146]	18th-century literature	2	2	Parallel Coordinates, dot plot	French

Table 3.3: Summary of visual design characteristics for related work discussed in the sections: [3.3], [3.3] and [3.3]. The “-” sign in the table indicates that an approach does not provide a distant or a close reading view. “Arbitrary” means the number is open based on the author’s claims and “Not-specified” means the number of parallel documents is not mentioned in the paper and is not exemplified. The references are ordered based on publication date.

Version Control Systems: Although software evolution visualization approaches [160,161] differ from our work, they feature some overlap as they try to compare and visualize the similarity and differences between source code. McNabb and Laramée [162] include eight surveys in the software visualization category –e.g. [163–166]. The goal of the tools in this section is to visualize the edit history of source code text.

3.3 Related Work of TransVis

We discuss general comparison as a task and the design that supports it based on Gleicher *et al.* [176] in Section [3.3]. Then, we group the related literature into four groups. The first category in Section [3.3] presents the general design that supports comparison of text in general. The second group in Section [3.3] discusses the designs that support visual comparison between parallel text. The third group in Section [3.3] presents visualization solutions that enable digital humanities tasks. The fourth group in Section [3.3] introduces previous visualizations of Shakespeare’s *Othello*.

In Table 3.1, we summarize the visual comparative approaches that facilitate parallel text comparison tasks. We indicate what type of documents they study, the maximum number of documents viewed in parallel in both close and distant views, the visual designs used in the distant views, as well as the language studied. If the meta-data is not provided explicitly, we extract it from the examples provided.

Visual Comparison as a General Task Comparison is included in most task taxonomies [177–179]. It facilitates the exploration of the data in order to understand the similarities or differences between comparison elements [180,181].

There are many approaches developed to perform comparative tasks. However, users and systems can often perform comparative tasks even if the systems are not developed in the context of comparison [181]. Gleicher *et al.* [176] group comparative visualizations into three representations: juxtaposition which shows objects side-by-side, superposition which overlays objects in the same visual space, and explicit representation of relationships. We can distinguish between the three categories in different ways. The separated object’s design relies on the memory of the user to conduct a complete comparison and there is no correspondence between them. Overlay designs use the same visual coordinate system to layout objects and proximity is needed to represent the connection. Explicit encoding facilitates computational tasks to investigate relationships. Gleicher *et al.* believe that interaction techniques such as brushing and linking can be helpful when applied to facilitate visual comparison. Also, using animation to show, for example, transitions can be helpful in understanding the connection between objects. Animation approaches can be useful to aid comparison between related objects, however, they can be problematic when not implemented carefully [182].

Visual Designs for Visual Comparison of General Text There are approaches designed to support the analysis of a single document which, however, can be extended to facilitate comparison between multiple documents. Fingerprinting approaches [183,184] are used to highlight semantic text properties on different hierarchy levels such as the development of relationships between characters in literature. The Text Variation Explorer [185] provides linguistically-assisted visualization to examine a user-selected text window. VarifocalReader [186] presents an interactive multi-layer visual design based on the hierarchy level of a text, such as chapters, pages, and sentences. It incorporates multiple visual presentations to support the analyst exploring the document, such as bar charts, pictograms, and word clouds.

Many visualization systems are designed to analyze collections of documents without explicit support of inner relationships between documents. For example, Brehmer *et al.* [187] present an open-source platform that analyzes user-uploaded documents. It also allows the user to create custom visual layouts. Also, there are many techniques that visualize results of queries, such as Sparkler [188] which plot the results on a radar-like view and the search-engine similarity (SES) tool [189] which visualizes the results of multiple web search engines using multiple views.

Our work is different from the aforementioned work. We present a specially customized visualization of explicit parallel texts that are strongly related to each other, i.e. versions or editions of the same text. In the next section, we examine the related work in the area of parallel text.

Visual Designs for Comparison of Parallel Texts There are a number of different visual representations that can facilitate visual comparison between multiple texts. The most common layout to visualize parallel texts is juxtaposition. For example ItLv [167] combines a timeline chart with multiple bar charts stacked vertically to represent documents. Another example is the Versioning Machine tool [153] which enables the user to investigate multiple documents side-by-side and integrates linking functionality to highlight corresponding text fragments. Similarly, multiple approaches use a side-by-side layout to represent compared objects, such as Jong *et al.* [168], Welsh and Hooper [170], Behrisch *et al.* [171], Wheelles and Jensen [154], the text view by Geng *et al.* [64], the text reader by Jänicke *et al.* [111], Howell *et al.* [172], Cheesman *et al.* [56], and Jänicke *et al.* [175]. Asokarajan *et al.* [174] visualize the variation in a pixel-base matrix where the x-axis represents the offset in the text and the y-axis represents the variation (witnesses). They also visualize the summary of variation at the pages, lines, and words level. We extend the discussion of some of these approaches in Section 3.3.

Plagiarism detection is an application of visual text comparison as in White and Joy [156]. Also, Riehmann *et al.* [114] combine an overview slope graph and glyph-based detailed representations to explore given text against multiple sources.

Different approaches overlay parallel texts in the same coordinate system in order to communicate comparative objectives. For example, the variant graph in Jänicke *et al.* [190], Storylines in Silvia *et al.* [136], and Geng *et al.* [58,64]. The variant graphs and parallel coordinates visualization represent each object as a line in the visual space. In the variant graphs, the y-axis illustrates the offset in the text or time.

3. Background

There are approaches which extract relationships between parallel documents and explicitly visualize them to support visual comparison. The Stylometric representation of versions in Cheesman *et al.* [56] encodes the similarity between connected texts using the thickness and length of the links. Explicit encoding of relationships can be implemented using a dot plots representation to detect similarity and dissimilarity patterns, such as in Ribler and Abrams [155], in Jänicke *et al.* [111], and in Abdul-Rahman *et al.* [146]. Collins *et al.* (Parallel Tag Clouds) [108] also uses links and word clouds to encode the relationships between documents.

The system we present uses juxtaposition but with up to 38 close as well as distant reading of parallel translations. It also features explicit alignment curves. The exploration and interaction techniques are customized and implemented to satisfy the user requirements and tasks stated in Section 6.3.

Examples of Visualizations from Digital Humanities There are many visualization solutions that enable digital humanities' primary tasks. In this section, we focus on visual approaches that feature alignments between parallel texts. Jong *et al.* [168] present an interactive tool that conveys the structure of parallel texts using color-coded boxes representing words. The reader can switch between the structural view and the textual view to facilitate close reading.

Büchler *et al.* [169] introduce a graph visualization to illustrate citation variation among documents in a corpus. They provide a distant reading view of the citation in the corpus using an interactive bar chart.

Howell *et al.* [172] propose a close reading visual design driven by digital humanities methodologies of the novel *The Secret Scripture* (2008). They visually compare two different encodings of the same novel using different color-coded highlights.

Jänicke *et al.* [111] introduce distinctive contributions to this field. They introduce multiple visual designs to depict text re-use between collections of text. For distant reading, they design a visual matrix to discover the type and amount of text re-use between pairs of texts. Additionally, they introduce the text re-use reader which consists of two panels: a dot plot view and a text reader. The former view depicts the type of text re-use between two texts, e.g. a diagonal pattern indicates sections repetition while vertical and horizontal patterns indicate phrase re-use. The text reader shows two panels aligning two documents, both panels are linked and respond to one another. They also introduce the text variant graph [113] to detect variations between versions at the sentence level. The graph uses color-coded links for each version and

font size to encode the number of occurrences among all versions. They demonstrate a graph applied to five versions of the Bible.

Jänicke *et al.* [173] propose an extended distant view of the variant graph to support analysis on higher text abstractions such as sections or chapters. They exemplify their method with a distant reading of 24 Bible editions.

Jänicke and Wrisley [175] propose a visual analytics environment that supports aligning two versions, or more, computationally. Also, the tool integrates different interactive methods that analyze textual alignments, along with an intermediate view between close and distant reading which they call “Meso reading”. The Meso reading view combines the text and the statistical features together within the same visual field.

The difference between our work and the related work discussed in this section is that our proposed design connects a distant reading view of all 38 translations with the close reading view in a novel way by smooth zooming and panning. Interactive zooming and aligning facilitate comparison of the related speeches across a number of parallel texts (Section 6.4). Previous work separates the close and distant reading views in multiple windows. The user is required to cognitively integrate the two. Also, we encode means to help the user validate alignment or translation of the segments using similarity metrics. The Eddy and Viv metrics are calculated interactively and dynamically to reflect the similarity among the current selection of parallel translations. Among all of the related work, our proposed design deals with a cross-language dataset and presents a macro (distant) view of the entire collection. Previous work does not generally support comparison of over 30 aligned translations and is generally restricted to the English language or a single language.

In Section 3.3, we introduce the related literature that analyses and provides visual designs of Shakespeare’s *Othello* collection.

Previous Work on Shakespeare’s *Othello* In this section, we discuss previous visualizations of Shakespeare’s *Othello*. Geng *et al.* [58] introduce a focus+context parallel coordinates layout for comparing eight translations of Shakespeare’s *Othello*. Their design consists of two main components. The first is a distant view represented by parallel coordinates to show the variation between translations and the use of the most frequent words. A collective concordance of the most frequent words is shown in the column on the far left. Each coordinate represents a word-translation pair, and the thickness of the bar encodes the similarity rank. They support various interaction techniques to aid exploration and investigation, such as brushing,

3. Background

selection, and linking. The second component of the visual design is the close view which shows the actual text and highlights the selected keywords.

Geng *et al.* [64] integrate multiple visual designs to illustrate the similarity between subsets of translations of Shakespeare’s play *Othello*. They provide visual designs to support distant reading, such as heat maps to illustrate segment structure, and parallel coordinates to depict similarity among versions. In the text view, close reading is obtained by showing the text segments in multiple versions of the play.

Cheesman *et al.* [56] present a web-based tool that enables the user to create parallel, segment-aligned multi-version corpora. The main goal of their project is to digitally explore patterns of variation among multiple translations. They present two overview designs which provide a distant reading view of the corpus. A small multiples pairwise alignment map of 35 German translations is used. Each translation is aligned with the base English text. Each speech is represented by a vertical rectangle and the height of the rectangle encodes the length of the speech. The edges between each translation and the base text represent alignments between segments. The viewer can identify different attributes of each text and make comparisons, such as the variation in length between translations and the base text. In the same overview context, they provide more analytical and statistical aggregations of the corpus data represented by a stylometric network. The network diagram shows translation clusters which depict the similarities among versions. The connection edges represent the similarities in particular sets of frequency counts, and the thickness of the edges reflects the degree of similarity.

Additionally, Cheesman *et al.* [56] provide a detailed interface which aligns segments of the base text with translated versions along with similarities metrics (Eddy and Viv) (Explained in Section 6.2).

The difference between the work presented in this section and our work is that Geng *et al.* [58, 64] present only a subset of the collection and is difficult to scale accordingly. On the other hand, Cheesman *et al.*’s [56] alignment map aligns only one translation with the base text and does not encode any similarity features. The Eddy and Viv interface provides only a close reading and does not allow filtering and selection of translations. Our work supports the comparison of the whole collection and incorporates the encoding of similarity metrics. It provides a variety of interaction and exploration techniques that facilitate the analysis and visualization of the collection. There is some overlap of co-authorship between this current work and previous work on Shakespeare’s *Othello*. We exploit the previous studies to guide the current work.

In summary, the current work is unique in that it includes 38 translations along with the source text. The current work supports integrated distant and close reading in the same view and implements them with smooth zooming and panning. Also, it allows the user to explore different regions of interest and stores them for further analysis. The current design allows the user to interactively customize the alignment overview, and subsequently update the similarity metrics based on the user's preference. Although, the TLC (Term Level Comparison) design is not novel, including it in the process of exploring the dataset is novel and proves useful. See Sections: [6.4.1.3](#) and [6.5.2](#) for practical use of the TLC view.

3.4 Chapter Summary

In this chapter, we introduce the previous work related to our technical contribution chapters. We summarized the previous literature and also highlight how our work differs.

Chapter 4

VNLP: Visible Natural Language Processing

“We all need people who will give us feedback. That’s how we improve.”

–Bill Gates (1955-)

Contents

4.1 Introduction and Motivation	60
4.2 Definitions and Terminology	61
4.3 Requirement Analysis	64
4.4 Implementation and Design of Visible NLP pipeline	64
4.5 Evaluation	73
4.6 Chapter Summary	79

This chapter aims to address the second research objective [Ob2]. It presents VNLP, an interactive, customizable, visual framework that enables users to observe and participate in the NLP pipeline processes, explicitly manipulate the parameters of each step, and explore the result visually based on user preferences. The visible NLP (VNLP) design is applied to a text similarity application to demonstrate the utility and advantages of a visible and transparent NLP pipeline in supporting users to understand and justify both the process and results. We also report feedback on our framework from a modern languages expert.

4.1 Introduction and Motivation

Visual computing approaches have been adapted in order to understand and open up machine and deep learning methods, and have been used as an educational means to understand black-box machine learning techniques. For example, TensorFlow Playground [125] is an interactive, web-based tool that enables users to understand neural networks via visualization. Also, Strobel et al. [191] use visualization techniques to analyze the hidden state dynamics of recurrent neural networks (RNNs). Recently, Chatzimpampas et al. [122] present a survey of surveys on the use of visualization for interpreting machine learning models.

However, there remains a lack of such approaches that demonstrate visualization techniques which enable the user to see the results of Natural Language Processing (NLP) processes.

The black-box metaphor is defined by Cambridge dictionary [192] as: “a system or process that uses information to produce a particular set of results, but that works in a way that is secret or difficult to understand.” Merriam-Webster dictionary [193] also defines black-box as: “anything that has mysterious or unknown internal functions or mechanisms.” Guidotti et al. [194] in their survey describe black-box systems as systems that hide their internal logic to the user [194]. This usually applies to machine learning and artificial intelligence models as the user can not interpret their behaviour and predictions. In the context of this chapter, black-box is used to refer to a system that lacks the explanation of how the results are derived and does not enable the user to observe intermediate results and fully understand every stage of the process. For example, a common challenge with standard NLP tools is that they produce results and do not obviously relate to the original text such as in normalization. Furthermore, many standard pre-processing steps involve stop words removal and do not enable users to visually moderate this list.

Additionally, the lack of transparency is considered a challenge when developing interdisciplinary visual analytics tools. Visualization also tends to reduce informational dimensions to produce a focus that shows certain perspectives or interpretations of the data [69]. As a result, intended users struggle to trust such results until they understand how they are derived, which is in most cases very challenging. In this chapter, we address this challenge by making the NLP process visible, transparent, user-steerable, and understandable. To achieve that, our proposed tool leverages both the machine’s computation power and human intelligence. It enables users to set explicit parameters to interactively guide the automation. Complete automation can accelerate the process however that is not the goal of VNLP.

While previous related research is generally guided by the well-established information visualization mantra [96]: “Overview first, zoom and filter, then details-on-demand”, this chapter presents an alternative approach that focuses on the details first: in other words, the process that is used to generate the overview in the first place. Our approach starts with raw text input into the NLP pipeline before developing visible layers of abstraction step-by-step to help the user understand the underlying choices made at each stage of the VNLP pipeline. Figure 4.1 illustrates the visible stages and the corresponding visual encodings. Finally, the overall visible result is explored based on the combined machine + user’s parameter choices and intelligence.

Feldman [195] introduces seven process levels that NLP systems use to understand spoken language or text: the phonetic, morphological, syntactic, semantic, discourse and pragmatic levels. In this context, our design is concerned with the presentation of the input data and the morphological level where the smallest parts of the texts are transformed into their base forms. Our novel design makes this transformation fully visible.

This chapter contributes the following:

- The introduction of the Visible NLP (VNLP) concept;
- A novel interactive design of a generic VNLP pipeline that enables users to explicitly observe the NLP pipeline processes and update the parameters at each processing stage;
- A case study application to text similarity quantification to demonstrate the usefulness and advantages of our approach; and
- Feedback on our framework from a domain expert in modern languages.

The rest of this chapter is organized as follows: Section 4.2 defines the most important and domain-related terminology. Section 4.3 outlines the design requirements. Section 4.4 introduces the VNLP implementation and design. Section 4.5 is dedicated to the evaluation of our visible framework. Section 4.6 introduce the future work possibilities of our research.

4.2 Definitions and Terminology

This section defines the most important and domain-related terminology required for developing a visible NLP pipeline.

4. VNLP: Visible Natural Language Processing

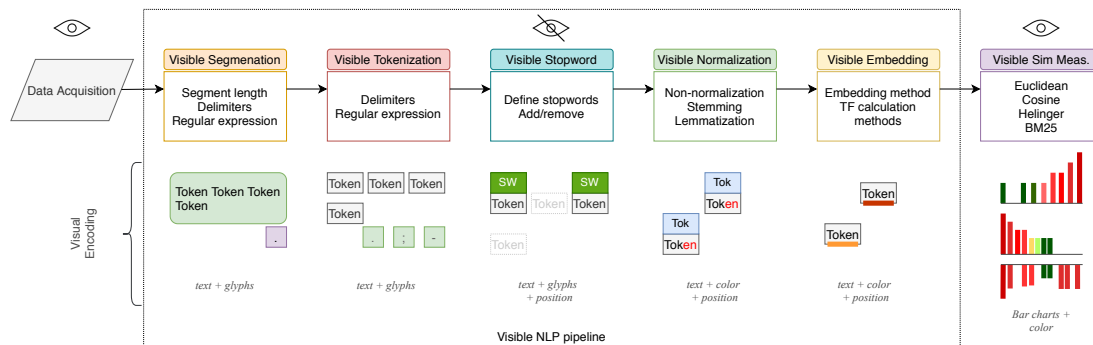


Figure 4.1: Our VNLP pipeline illustrates five main NLP stages: text segmentation, tokenization, stop word removal, normalization, and embeddings. We integrate an application to text similarity quantification to demonstrate the usefulness and advantages of our approach. In the lower half, we show the corresponding visual encodings of the VNLP pipeline stages.

Segmentation includes methods that break a document down into independent and minimal textual components which are usually called segments or tokens [196]. A text segment is defined as a contiguous piece of text that is linked to itself but largely disconnected from the adjacent text [197].

Tokenization is the process of dividing a segment into individual tokens. A token is an instance of a sequence of characters that are semantically grouped together [198]. Some literature, such as Pak and Teh [196], considers tokenization as a sub function of segmentation. To some extent, we agree they overlap and could be used interchangeably, however in the context of this chapter, we refer to tokenization and segmentation as two stages, as defined here.

Stopwords also called function words (as opposed to content words) are defined as commonly used words that are omitted in the process of generating a concordance [199]. Stopwords can include very common terms such as definite and indefinite articles, auxiliary verbs, prepositions and conjunctions, as well as common corpus-related words with no discriminant value within a given domain or corpus. An example of the latter is the word *learning*, which can be a stopword for the domain of education and a content word in the domain of computer science [200]. Stopwords have grammatical functions and can be defined subjectively or objectively. There are multiple studies that focus on the significance of stopword removal as a pre-processing step [201-203].

Normalization includes techniques that are applied to reduce the dimensionality of feature space [204]. It involves applying linguistic models to restore words to their canonical forms in a standard language [205]. Stemming and lemmatization are examples of normalization.

Stemming is the procedure that standardizes and generally truncates all words with the same root to a common base form called *a stem* irrespective of their inflections [206]. For example: *amusing* and *amusement* have the same stem as *amus* [207].

Lemmatization is similar to stemming in terms of function. Lemmatization functions produce lemmas which are dictionary-based words which, unlike stems, are not truncated or ambiguous [208]. For example: *amusing* and *amusement* have the same lemma as *amuse* [207].

Document embeddings, or so-called document representations [209], are the mapping of documents to numerical vector spaces. There are different approaches used to generate document embeddings such as TF-IDF [210] and BM25 [211]. Contextual word embeddings can also be used to vectorize documents [209,212].

Feature is an individual measurable property, characteristic, or behavior observed [213]. In the context of document embeddings, a feature is a unique or unusual term, phrase, or sentence that can characterize a document.

The similarity measurements we use are as follows:

Cosine Distance [d_{\cosine}]: When two documents, T_1 and T_2 , are represented as feature vectors, cosine distance is the angle between the vectors T_1 and T_2 . The cosine distance is the dot product $T_1 \cdot T_2$ [214].

Hellinger Distance [$d_{hellinger}$]: Hellinger distance (or Bhattacharyya distance) is usually used to compute the similarity between two probability distributions. It can be used for both discrete and continuous distributions. In our case, since we are using bag-of-words features, distributions are discrete.

Okapi BM25: BM25 was developed as part of the Okapi information retrieval system implemented at City University, London to retrieve a bibliographic reference database [211]. BM25 stands for "Best Matching" and is one of the variants of the BM best match function, considered to be the most commonly used version [215].

Word Mover's Distance [d_{wmd}]: WMD is based on the results of the contextual word

embeddings produced by word2vec [212]. Word embeddings are semantically meaningful representations of words generated using the local co-occurrences in a pre-defined window-sized neighborhood. The embeddings preserve the semantic relationships between words and enable arithmetic operators such as $\text{vector}(\text{'Berlin'}) - \text{vector}(\text{'Germany'}) + \text{vector}(\text{'France'}) \approx \text{vector}(\text{'Paris'})$ [212]. WMD calculates the distance between segments, d_{wmd} , and assumes that similar words have similar embeddings. WMD is designed to utilize word embedding relations and to overcome word transformations and reforms.

4.3 Requirement Analysis

Throughout our collaboration with domain experts in the digital humanities (DH), there was consistent interest in a transparent design that reveals how the results are derived rather than just presenting the end results. The experts also expressed appreciation of an informative framework that explains intermediate steps and makes them visible. Visual solutions that exhibit black-box behavior do not facilitate interpretation and exploration of the derived results. We discuss this challenge also in Chapter 7. We established and incrementally refined the following requirements based on our discussions with the DH expert:

R1 Provide information about each pipeline stage that includes an explanation of the corresponding stage, what it outputs, and how it affects the intermediate results.

R2 Show explicit results at each stage and enable the user to adjust the parameters to observe the effect at the individual stage level.

R3 Provide a dynamic layout that customizes the pipeline and scales up and down in line with user preferences.

R4 Demonstrate the usefulness and advantages of the design through an NLP application.

The requirements are coupled to the discussion of our design in the following section.

4.4 Implementation and Design of Visible NLP pipeline

Overview of the Visible NLP Pipeline: Our VNLP pipeline illustrated in Figure 4.1 shows the NLP sub-processes that play a major role in NLP results, quality, and correctness. Apart

4.4. Implementation and Design of Visible NLP pipeline

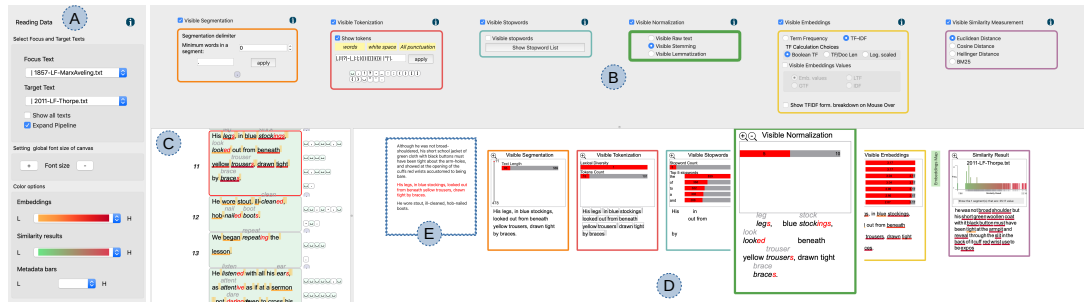


Figure 4.2: An overview of the VNLP pipeline. (A) A dialogue that accommodates options to customize the text, update the font size for accessibility, and the color options. (B) The VNLP pipeline GUI where the user controls the parameters of each stage. (C) The user-chosen focus text. (E) The context of the user-selected segment. (D) The visible results pipeline with each result reflecting the parameters chosen in the corresponding GUI component.

from the raw input data and the end results, the user cannot normally observe the behavior, intermediate results and parameters of the NLP algorithms. The visible NLP pipeline contains the primary stages: visible text segmentation, visible text tokenization, visible stopwords removal, visible text normalization, and visible embeddings generation. In the last phase, visible similarity measurements may be applied to derive alignments. This phase includes a selection of popular, state-of-the-art distance and similarity measurements. Cosine, Hellinger, and Okapi BM25 measurements support the TF and TF-IDF embeddings [216]. Word Mover’s Distance, on the other hand, is hypothesized to be the best that utilizes the quality of word2vec embeddings [209].

The implementation of the VNLP pipeline consists of four main components. The first is a window which accommodates options to customize the focus and target texts (Figure 4.2A) and enables the user to set their preferred font size to make the layout more accessible (R3). It also provides an option that shows the similarity results in the other texts in the collection. The user options include multiple preset color schemes which can be applied to visible embeddings and a similarity histogram graph [217, 218]. This window appears only on demand as it incorporates functions related to the VNLP application process (R3).

The second component is the GUI pipeline where the user interaction is applied in order to modify and steer the underlying visible pipeline stages (Figure 4.2B). The GUI components are ordered based on the pipeline overview discussed in Section 4.4 and shown in Figure 4.1. Each GUI component integrates an information icon which, when clicked, presents detailed

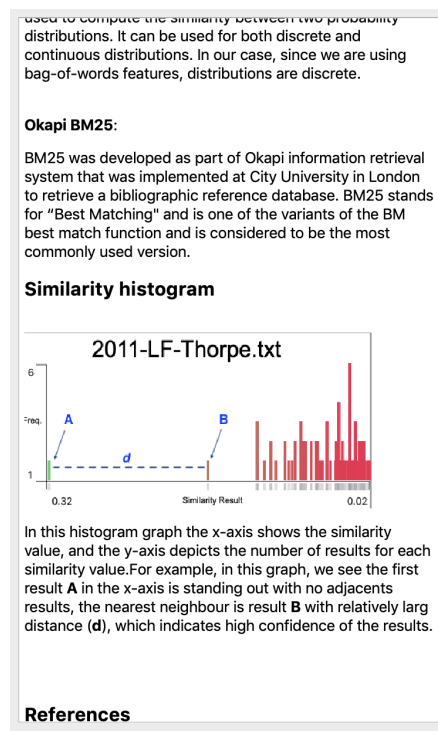


Figure 4.3: An information dialog view that shows a detailed explanation of the corresponding VNLN pipeline stage.

information about the corresponding stage, as shown in Figure 4.3 (R1). Each GUI component can be toggled on or off and the corresponding stage in the visible result pipeline is then updated in the other view below (R4). This is to help the user focus on any stage and display the space efficiently.

The third component is the current visible results pipeline which renders the results and responds to the user's interaction in the GUI pipeline (Figure 4.2D). Each result integrates a graph that provides a metadata analysis related to the corresponding stage. The user can magnify a given stage for closer analysis (R3). In the following sections, each stage is discussed in detail as well as the correspondence between the visible results and the GUI components (R2). The visible results pipeline view includes the current user-chosen segment and context segments (Figure 4.2E), where the user can navigate to the previous or next segment in the same window. The current segment is indicated by a red font color. Next to the visible embeddings result, the window provides a green button (magnified in Figure 4.2D) which leads to the embedding map to illustrate the overlap between the focus and target segment (R4). This ap-

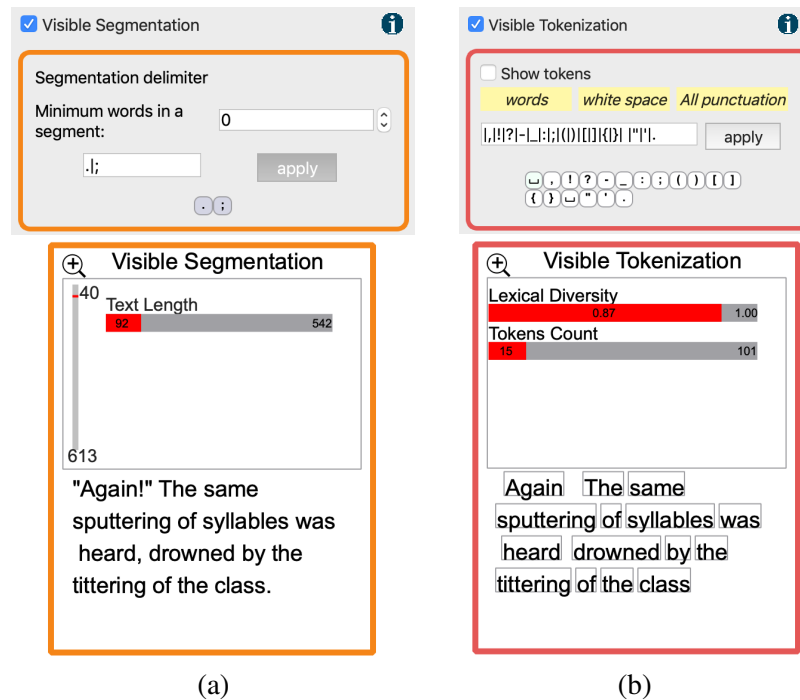


Figure 4.4: Top: the visible tokenization and segmentation GUI components. Bottom: the visible tokenization and segmentation results. In each result, metadata analysis is provided for an overview of the results.

plication is discussed further in Section 4.4.4. The final item in this view, called the “similarity results”, demonstrates the VNLP application and is discussed further in Section 4.4.4 (R4).

The fourth component shows the focus text which is segmented based on the user’s choice (Figure 4.2C). The segments are illustrated top-down as they appear in the original text in order to facilitate the reading task. In this view, the tokenization and segmentation separators are illustrated next to the individual segments to show the position of the separators in each segment.

4.4.1 Visible Segmentation and Tokenization

Segmenting the text into tokens or sentences might sound like a trivial task, but various implicit decisions and different languages can affect the results. Most default segmentation tools do not necessarily provide similar results and each implementation incorporates implicit decisions of which the user may not be aware.

For example, the NLTK function (`sent_tokenize()`) [219] distinguishes between full stops

and periods that are part of a sentence such as “Mr.”, “U.S.A”, while other default implementations do not consider such cases. Cases in which a period is followed by a capital letter are not considered in these default implementations. Other punctuation such as ‘?’ , ‘!’ and ‘;’ are usually ignored in default implementations.

Our design enables the user to explicitly define how to segment and tokenize the text using specific separators or regular expressions. The user-specified delimiters are shown in the GUI component as the user enters them, as can be seen in Figure 4.4. It also incorporates a segmentation threshold to avoid segments that are too short. The visible result is illustrated in Figure 4.4. Our implementation offers a metadata analysis of the segments and tokens in the focus text in order to provide an overview of the results. In Figure 4.4a, the vertical bar indicates the relative position of the user-selected segment and shows the total number of segments in the focus text. The horizontal bar illustrates the length of the current segment and how it compares relative to the other segment lengths. In Figure 4.4b, the metadata indicates the lexical diversity of the selected segment, which measures the number of lexical tokens in the segment, and shows the number of tokens the segment includes compared to other segments in the focus text.

Using the visible framework, the user can observe the segments and how they are derived, as well as identify unwanted behavior that is difficult to discover without a transparent system. For example, Figure 4.5a shows a case where a segmentation delimiter, a period in this case, is placed in the middle of a quote. Also, due to different writing standards, the closing quotation mark is placed after the period in the middle segment, which causes the quotation mark to be pushed to the following segment. Another example is shown in Figure 4.5b, where the period in the word “St. Romain” causes the main segment to be divided into two.

In the case of tokenization, the user can transparently observe and examine the derived tokens. There are many ways in which tokenization implementations can derive undesired results. For instance, in Figure 4.6 the word “o’clock” is divided into two tokens, “o” and “clock”, when using the punctuation-based tokenizer. This can affect the results in different NLP applications. Figure 4.6b shows an example of an interesting tokenization choice where the compound word “well-to-do” is divided into three tokens which all could be stopwords and consequently the phrase is removed in the next NLP stage.

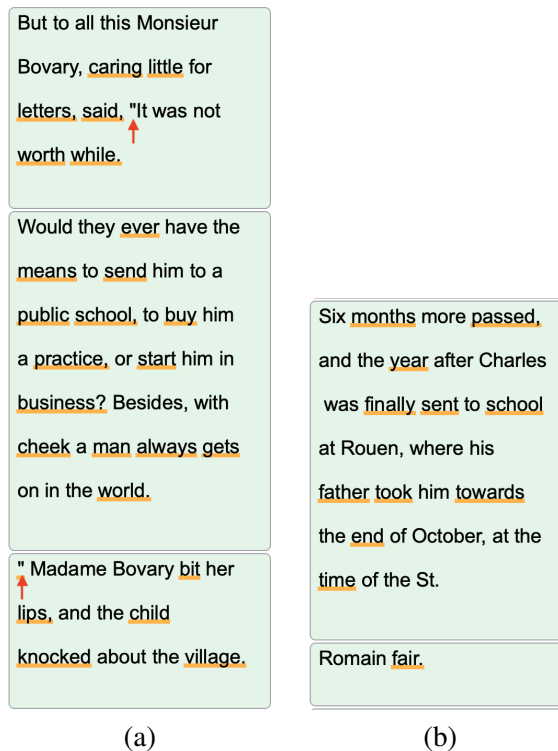


Figure 4.5: Two examples of ambiguous segmentation cases. (a) A period placement in a quote (annotated by arrows) results in a new segment. (b) The second segment was generated due to the period in the word “St. Romain”.

4.4.2 Visible Stopwords

Stopword removal is a common practice in text pre-processing and information retrieval applications. However, other approaches claim that the removal of stopwords may lead to an increase in false alignments [115, 116]. Research indicates that stopwords can be useful in some applications such as authorship attribution as stopwords tend to be included by the author subconsciously [9]. Most approaches facilitate and integrate a fixed set of stopwords and do not incorporate means for the user to explore and manipulate the stopwords list. By contrast, our design enables the user to observe the stopwords in the focus text by annotating them, as shown in Figure 4.8. We also implement interactive means to add or remove stopwords and observe the effect on the results accordingly. As shown in Figure 4.7a, the user can include the stopwords removal function in the visible NLP process, see the current stopwords list and add or remove stopwords. The user can also interactively add or remove stopwords from the visible results of stopwords and normalization stage by right-clicking on the word, as can be

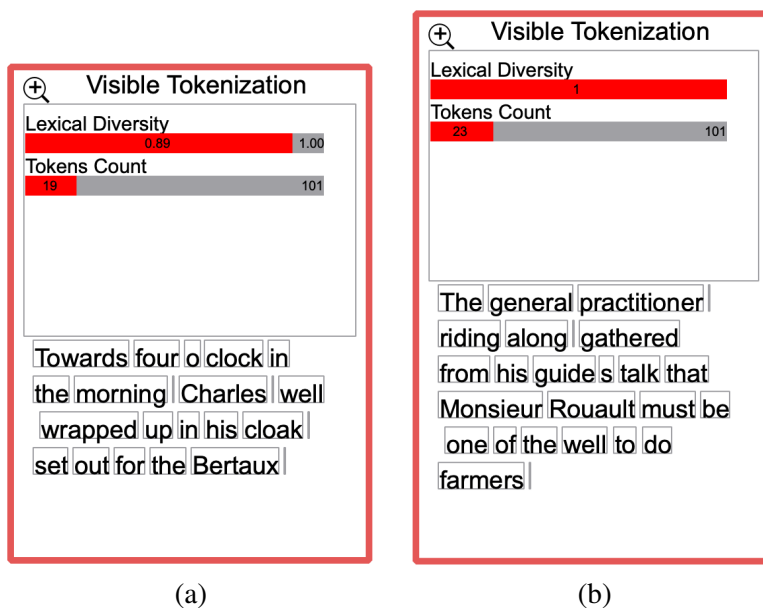


Figure 4.6: Two examples of erroneous tokenization cases. (a) A inaccurate tokenization of the word “o’clock”. (b) The compound word “well-to-do” is divided into three tokens which are considered stopwords and consequently removed in the following NLP stage.

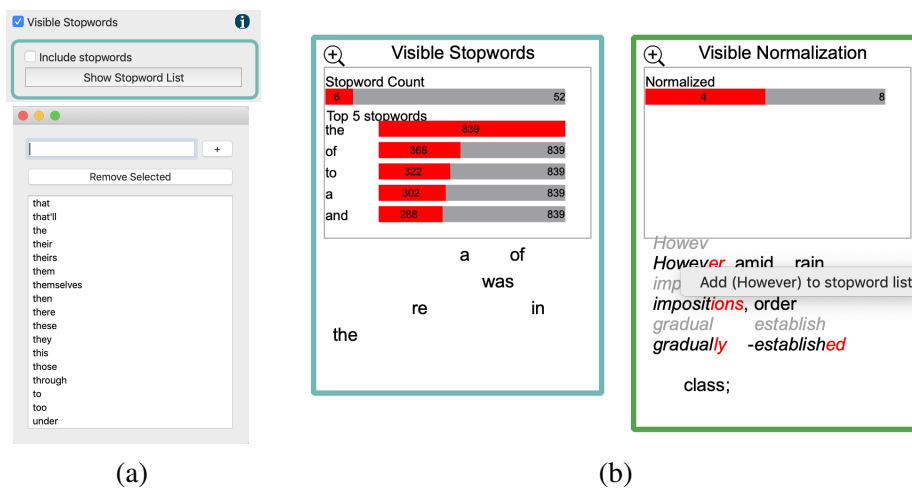


Figure 4.7: (a) in the stopword GUI window, the user can include or remove stopwords. The GUI provides a list of the stopwords where the user can add or remove them. (b) From the visible results of both the stopwords and normalization, the user can observe stopwords and add words from the normalization results to the stopwords list and vice versa.



Figure 4.8: Two cases of visible stopwords. The left shows a case where the entire segment is composed of stopwords. The right shows a segment with five sequential stopwords. The glyphs that accompany each segment illustrate the tokenization delimiters chosen by the user.

seen in Figure 4.7b. In this case, for example, the word “however” is not included in the NLTK stopword list [219]. Another example is shown in Figure 4.9b. The user observes that multiple words could be considered stopwords, such as “whose”, and “like” which are not included in the NLTK stopword list, and so can right-click on these words and add them to the stopword list, as shown in Figure 4.9a.

When applying our design to Shakespeare’s play *Othello* [220], some special cases can be observed. There are cases in which a segment consists only of stopwords, such as the segment “*So kann’s nicht sein*”, or is dominated by stopwords such as the segment “*Ich will mich nicht im Irrtum sicher schätzen*”, as shown in Figure 4.8. The choice of removing the stopwords is not necessarily constructive in such cases. In another ambiguous case, in our tool the word “*sei*” is not included in our list while other similar words are, such as “*sein*”, “*es*” and “*ist*”. Therefore, a transparent design that explicitly shows the results and enables user intervention can be useful.

4.4.3 Visible Normalization

Most of the approaches in our collection do not offer any means for the user to normalize, verify, or explore the result of normalization. In a previous collaboration, prior to this thesis, with the modern languages expert (e.g. [64]), they experienced frequent unsatisfactory results from the normalization implementations provided by GermaLemma [221] and TreeTagger [222]. Although this might be influenced by the nature of our data, we believe the normalization results need to be shown and verified for the user to understand and trust the analytical results. Our design enables users to choose raw text, stemmed text, or lemmatized text to be embedded in the next phase.

Visualizing the results of the normalization process can reveal interesting results for the

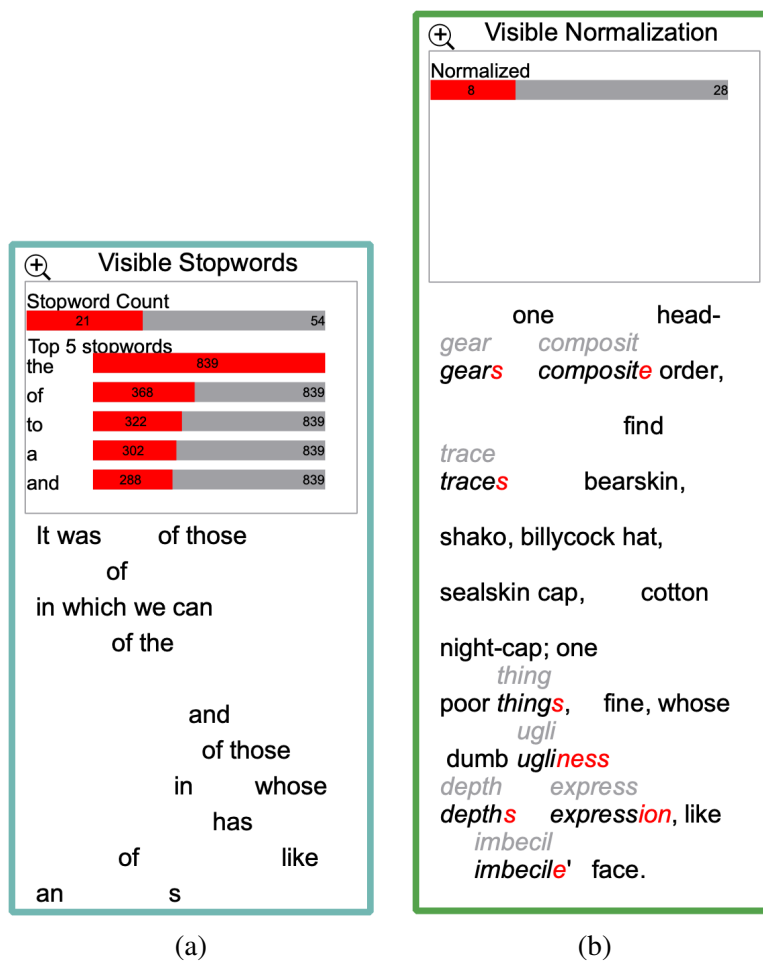


Figure 4.9: An example of stopword exploration. (a) In the visible stopword window, the user can explore the stopwords included in the selected segment. (b) In the visible normalization result the user can identify candidate words and add them to the stopword list.

domain experts that they may not expect or desire. For example, stemming can produce undesirably truncated results, such as the words “*forty, hundred*” stemmed to “*forti, hundr*”. While this is the underlying function of stemming, the domain expert may not appreciate such decisions. Another visible example is shown in Figure 4.9b. The user can see the normalized form on top of each word if it is different from the current form. For example, the word “ugliness” is transformed to “ugli” which leads to understanding the method used to produce the normalized form.

4.4.4 Visible Embeddings

The visible embedding section offers multiple options to manipulate and steer the feature extraction phase. As shown in Figure 4.10, the interactive options in the GUI window enable the user to specify the statistical quantifying approach. The terms frequency (TF) and term frequency-inverse document frequency (TF-IDF) are used to produce fixed-length vectors of word weights. Since the TF-IDF implementations have different definitions of TF [223], this phase enables the user to experiment with three different formulas for deriving the TF values in the TF-IDF function. This stage also enables the user to project different embedding values in the visible result as shown in Figure 4.10. The projected values can be changed to help the user understand the derived embedding results. The user can choose to view the current embedding values, the local term frequency, the global term frequency, or the inverse document frequency. These choices affect the implementation of the TF-IDF and can produce different results accordingly. When the user hovers the cursor over each word, these values can be seen explicitly, as shown in Figure 4.11a, and we also provide a breakdown of the formula if the user chooses, shown in Figure 4.11b.

Showing these values explicitly communicates to the user some of the differences between formulations and how they perform. For example, when the user examines the segment in Figure 4.10a, the words “one” and “cap” are considered the most distinctive words. This does not align with the domain user’s knowledge as these words are common in the current texts as indicated in the projected values on top of each word. In Figure 4.10b, the user changes the embedding generation parameters to use the boolean calculation for TF and observes improved results that correspond with their assumption.

4.5 Evaluation

We evaluate our design by utilizing its features to demonstrate the application of similarity quantification to support and analyze aligned translations in the target text. Following the application, we report feedback from a domain expert in modern languages and translation studies.

4.5.1 Visible Text Similarity Application

The visible similarity GUI, shown in 4.2B, provides a list of similarity measurements from which the user can choose. The visible embedding result incorporates a similarity histogram



Figure 4.10: An example of the exploration of user modifications to the visible embedding generation process. (a) The default embedding generation implementation results in common words such as, “one” and “cap” to become distinctive words. (b) After the user changes the TF calculation method, the words “one” and “cap” are considered non-distinctive and other more important words appear.

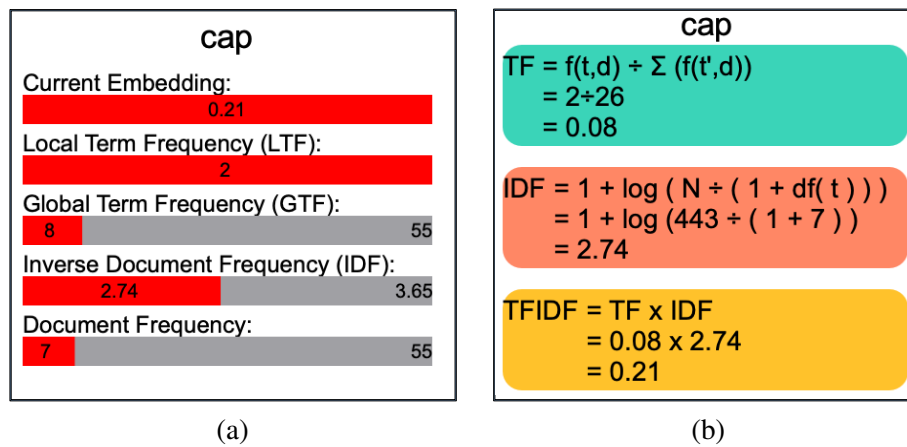


Figure 4.11: The two views that are shown when the user hovers over a word in the visible embedding result. (a) A summary of the embeddings values that are derived for this word. (b) A breakdown of the formula that is used to derive the current embeddings.

and shows the most similar segment text at the bottom of the visible result. The x-axis in the histogram depicts the similarity value and the y-axis shows the number of results in each similarity value. The rationale behind this design is to illustrate the notion of the confidence value presented by Alharbi et al. [1]. For example, Figure 4.12a shows the similarity results along the x-axis. The distance between the first value and the second value along the x-axis is short when compared with the distance between the first and second results in the histogram in Figure 4.12b. The histogram includes a user option to remove the last value column (usually the zero results) as it appears to skew the histogram distribution, as illustrated in Figure 4.12c. The user can also observe the similarity results in the other texts in the collection.

The embeddings map is implemented to help the user investigate the shared embeddings between the chosen focus segment and the segments in the target text. The embeddings map window, as shown in Figure 4.13, provides the user with multiple options by which to sort the embeddings, such as by alphabetical order, by focus or target embeddings values, or by the focus text word order. It also includes a navigation option to move to the next result based on the similarity results derived by the selected measurement. The map includes two bar charts where the x-axis is mapped to the common words of both segments and the y-axis depicts the embedding results.

Here, we demonstrate the usefulness of the VNLP in investigating the embeddings and understanding the process undertaken. When the user selects the segment starting with “ *We could see him working...* ”, the framework correctly shows the aligned segment from the target

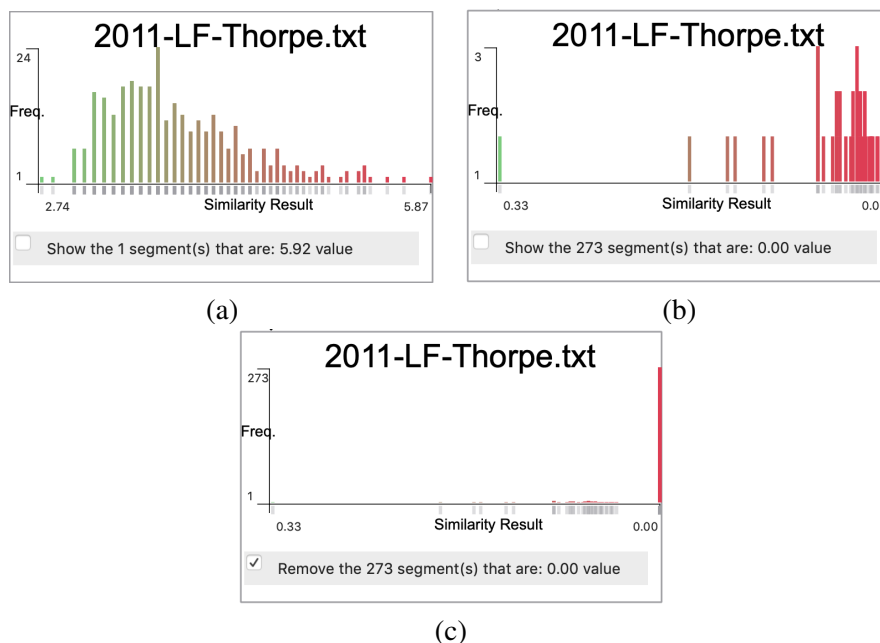


Figure 4.12: Three similarity histograms that show the similarity results along the x-axis. The y-axis indicates the number of results for each similarity value. The histogram in (a) indicates that the distance between the first value and the second value along the x-axis is small while it is relatively greater in the other histogram (b). The histogram in (c) shows the effect of showing the similarity values that equate to zero.

text. When the user examines the embeddings map as shown in Figure 4.14a, the user clearly observes that the words “*work*” and “*working*”, and the words “*look*” and “*looking*” are not combined. The user can also observe that the word “*dictionary*” appears twice due to the period in one of the occurrences. The similarity measurement, Cosine in this case, assigns a value of 0.20 to the result. The user chooses the stemming from the visible normalization options and changes the tokenization delimiters to consider other punctuation. The embeddings map interactively updates based on the user choices, as shown in Figure 4.14b. The user observes in the updated embeddings map that there is one word (stem) that combines both “*work*” and “*working*”, and “*look*” and “*looking*”. As well as this, the map only contains one stem for the word of “*dictionary*” (“*dictionary*”) after handling the punctuation issue. The updated similarity result value is considerably higher due to the changes (0.51). It shows that the word “*working*” is highly distinctive, but not when it is returned to its base form. This helps the user understand the basic notion of weighted terms and the TF-IDF principles. This visibly demonstrates to the user that the features quantity decreases from the map in Figure 4.14a which

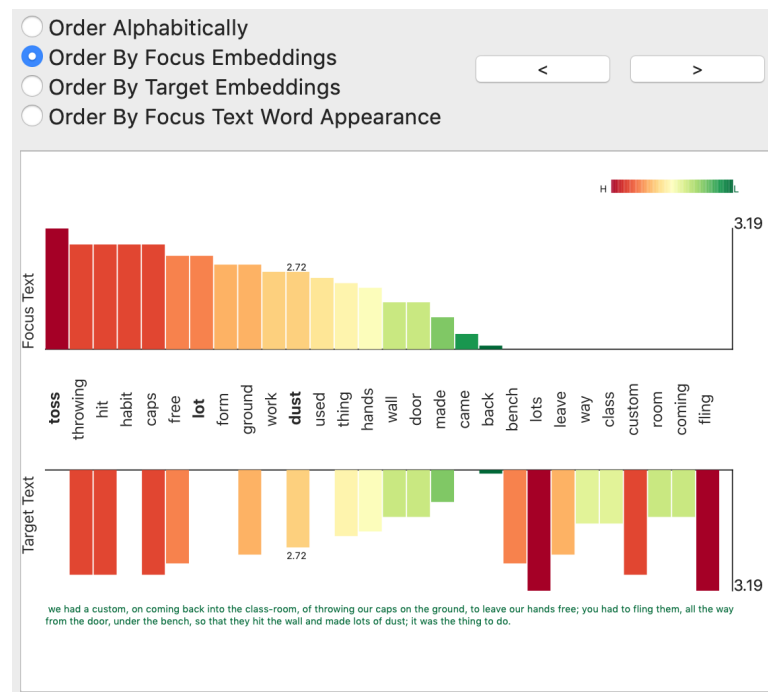


Figure 4.13: The embeddings map window that illustrates the embeddings values in both the focus and target texts. It integrates user options where the user sets the sorting of the embeddings values and navigates to other target segments. The two bar charts represent the embeddings in the focus and target text.

facilitates understanding the idea of features and dimensionality reduction. It also shows that the user can observe the inner features (words) and enhance the embeddings by editing the VNLG GUI options. Users are informed on the different stages of the VNLG and how they collectively influence the end result.

4.5.2 Domain Expert Feedback

When we first demonstrated the framework to the expert, he appreciated the idea of making the NLP pipeline visible, stating: *“This is very interesting and has a lot of potential for introducing NLP to students. I’m pretty sure it’s a unique idea.”*. When we presented the segmentation and tokenization options, he liked the visible options and the glyphs of the separators: *“That is really valuable. It is underestimated, but handling punctuation in text preparation and normalization is very difficult. There are lots of different approaches to use and the decisions you make have massive impacts on subsequent analyses. This is great! I think I could have a lot of fun playing with this.”*.

4. VNLP: Visible Natural Language Processing

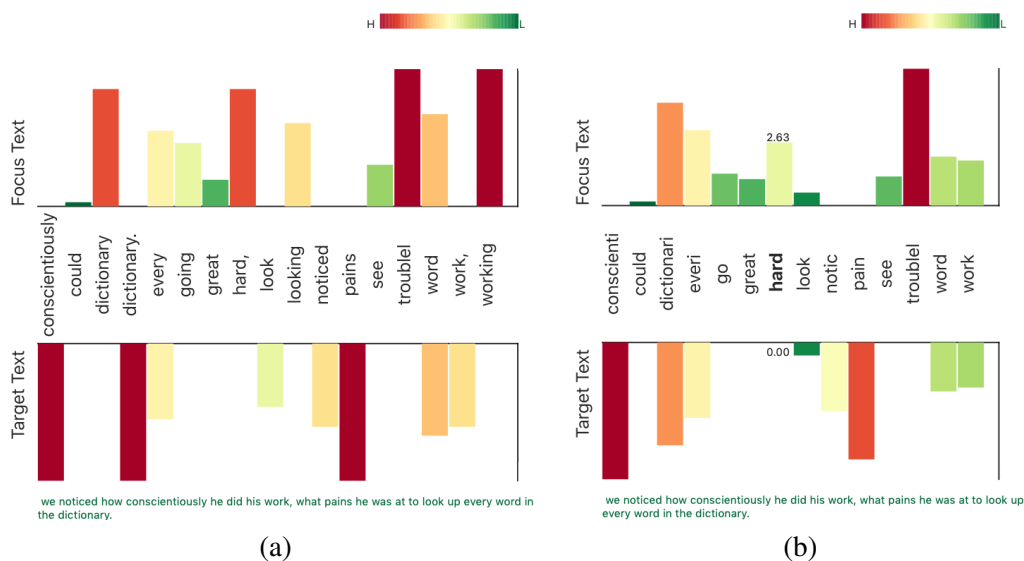


Figure 4.14: Two examples of the embeddings map which demonstrates the effect of the user interaction in the VNLP GUI. The map in (a) shows the default settings for both the tokenization and normalization. The map in (b) shows the reduction of features after applying more delimiters to the tokenization and stemming.

In the case of the visible stopwords and their correspondence with the visible normalization, the expert stated: *“I like that. It’s a perfect demonstration of the value of making the process visible, and giving me visible feedback if I make different choices. We normally present stopwords as one long list. Showing them in the text segments like this makes more immediate sense to users. It helps make the concept clear, and the implications of defining stopwords in different ways. The result is kind of poetic, too. I think a lot of people who are interested in literature will respond to this very well. It’s like a kind of concrete poetry.”* Furthermore, he suggested interacting with both windows in order to add or remove stopwords to make it easier for the user to experiment with the effects of altered lists, and so this feature was implemented.

The visible embedding window was challenging for the expert and this feature evolved most in response to his successive feedback sessions. At the early stages, the framework did not provide information about the different values used to calculate the embeddings and the different calculations. At one point, whilst investigating a case, the expert interrupted: *“Hang on! I’m trying to figure out how these values are derived.”* Making these values and calculations clear and transparent answered his questions and increased his trust in the framework. He stated: *“This is really informative and takes you through the steps, telling you what you need to know.”*

The final discussion with the expert focused on evaluating how useful this framework can be in teaching basic NLP principles. As a closing remark, the expert stated: *“I think it could be a really useful framework precisely in educational contexts, introducing NLP principles and processes to the kind of students we have in languages and translation or linguistics, who usually have limited computational skills and are nervous about NLP interfaces which assume a huge amount of knowledge. This lets them learn a lot by playing around with options which produce different results.”*

4.6 Chapter Summary

In this chapter, we present VNLP, a framework that enables users to observe and participate in the NLP pipeline processes, explicitly interact with the parameters of each step, and observe the effects on the visible VNLP result. The aim of this research is to implement an educational and transparent process of the NLP pipeline. We support this with an application of text similarity to demonstrate the usefulness of the VNLP. This work is a result of a close collaboration with an expert in modern languages and translation studies and evaluated through domain expert feedback.

Chapter 5

AlignVis: Semi-automatic Alignment and Visualization of Parallel Translations

“To succeed in your mission, you must have single-minded devotion to your goal.”

– A. P. J. Abdul Kalam (1931-2015)

Contents

5.1 Introduction and Motivation	82
5.2 Requirement Analysis	83
5.3 Definitions and Terminology	84
5.4 Design of AlignVis	85
5.5 Evaluation	93
5.6 Chapter Summary	98

This chapter aims to address the third research objective [Ob3] by presenting AlignVis [1], a visual tool which provides a semi-automatic alignment framework to align multiple translations. It displays the results of text similarity measurements and enables the user to create, verify, and edit alignments using a novel visual interface. The design consists of three main components: the alignment editor canvas, the post-edit area, and the user options panel. AlignVis exploits both close and distant reading and is designed to help digital humanities and translation scholars enhance the process of text alignment for multiple translations. The design of AlignVis is driven by iterative discussions with the domain expert which resulted in

five benefits: presenting an overview of the aligned translations, support for multiple alignments, enhancement and acceleration of the alignment process, alignment refinement, and testing different similarity measurements. We evaluate AlignVis with domain expert feedback and compare it with a standard alignment tool as well as computational and visual alignment tools.

5.1 Introduction and Motivation

When working with parallel versions or translations, the task of segment alignment – one-to-one, one-to-many, or one-to-nil alignments, in each direction – is a complex operation traditionally performed manually by scholars. When there is considerable uncertainty, orthography reform, or translation instability, manual alignment becomes challenging, tedious, and error-prone. Some view it as a menial task which should be outsourced if possible but for many scholars, the process of performing alignment is an opportunity to gain new knowledge and understanding of how texts relate to one another. In this case, a tool is needed which supports the process and allows the user to intervene.

Therefore, digital humanities and translation scholars spend considerable time using standard text alignment tools such as LF-Aligner [151] to create parallel corpus of translations –text paired with its translation into a second language [224]. Most of these tools require and heavily rely on domain experts to manually segment and validate translated texts one-by-one. We hypothesize that recent advancements in text mining and computational linguistics can address these challenges.

This chapter presents AlignVis, a tool that facilitates the advancement of text alignment techniques and provides interactive visual methods for the domain expert to edit and validate text alignments. AlignVis combines interactive visualization and domain knowledge intervention techniques to support exploration, validation, and refinement of machine recommended alignments. It enables the user to compare and test multiple text representations and similarity measurements to accelerate the alignment process.

Contributions: This chapter contributes the following:

- A novel visual alignment tool to help translation scholars align **multiple texts** simultaneously.
- An interactive visual interface to help **enhance and accelerate the alignment** process and enable modification of the alignments.

- Domain expert feedback, a comparison with a standard alignment tool, and a comparison with visual and computational alignment tools.

The remainder of this chapter is organized as follows: Section 5.2 outlines the design requirements. Section 5.3 explains the parallel translation data and the relevant terminologies. Section 5.4 introduces the design of AlignVis. Section 5.5 is dedicated to evaluation of AlignVis.

5.2 Requirement Analysis

Various tasks were identified during the course of our discussions with the domain expert. Humanities scholars, when studying divergent translations of literature, are interested in combining both distant and close reading. Domain experts also appreciate machine assistance to support and speed up the processes of comparative interpretation. The domain expert we collaborate with has previously tested different similarity measurement and would appreciate the ability to explore and observe the different results each measurement produces.

The requirements were derived and incrementally refined based on multiple meetings with the domain expert, as explained in Section 5.5. We couple the requirements to the discussion of our design.

RO. Provide an overview of the aligned translations: The domain expert is interested in an overview of the aligned translations to analyse and explore the variation among them, and would like to explore the overall relations between translations.

RN. Support for multiple alignments: The domain expert is interested in a design that facilitates and integrates alignments for multiple translations.

RA. Enhance and accelerate the alignment process: Given that the current practice of alignment is time-consuming, error-prone, and performed one-by-one, translation scholars are interested in a tool that enables them to enhance and accelerate the process of alignment.

RE. Allow the user to refine and update the alignments: The result of the automatic alignment is not always accurate due usually to instability and variation in translations. Therefore, the domain expert would like to be involved in the process of alignment and update the semi-automatic alignment process manually. This requirement is closely linked with requirement RA, but RE focuses on incorporating expert user knowledge which results in enhancing and accelerating the alignment process.

RS. Enable the user to apply and test different similarity measurements: The domain

expert would appreciate exploring different similarity measurements and examine the results that each measurement produces. They may observe if the results agree with the domain expert’s own understanding.

5.3 Definitions and Terminology

A description of the parallel translation dataset is provided in Chapter [1](#). The following describes domain-related terminology that corresponds to this chapter:

- **Segment** [s]: text that contains one or more words based on the user’s tokenization preferences (usually a sentence).
- **Alignment** [$a(s_i, s_j)$]: consists of two segments, (s_i, s_j) which are related to each other. The machine-recommended alignment is the result of the pre-processing phase. We use the notion $a(T_1, T_2)$ to refer to an alignment between two translations T_1 and T_2 . The notation $a(s_i, s_j)$ refers to an alignment between two individual segments s_i and s_j .
- **English Text** [T_E]: also called the source text. In our case, the source language is English, so we refer to the source text as the English text (T_E).
- **Focus Translation and Base Translation** [T_F, T_B]: can also be called the target texts. We feature two important target texts. The first is called the Base Translation (T_B) which the user chooses to represent the English text (T_E). The second is called the Focus Translation (T_F) which can be aligned with the T_B .
- **Sequential Alignments**: Sequential alignment is a common practice in the domain expert’s practice. The alignment process is carried out by creating an alignment $a(s_i, s_j)$, where $s_i \in T_1$ and $s_j \in T_2$. A standard process then creates the alignments $a(s_{i+1}, s_{j+1})$, $a(s_{i+2}, s_{j+2})$, etc. In the case of a mismatch, the domain expert corrects it and starts the process over from the corrected mismatch.
- **Distance Value** [$d(s_i, s_j)$]: indicates the distance, d , between $s_i \in T_B$ and $s_j \in T_F$. The distance may vary based on the similarity measurements used.
- **Alignment Confidence Value** [c]: is a gradient operator which measures the difference in distance values, $c = |d_1 - d_2|$, where $d_1 = d(s_i, s_j)$, and $d_2 = d(s_i, s_{j+1})$, and s_j, s_{j+1} are successive in the same translation. It is based on a heuristic used by the domain expert. If the distance between successive aligned segment pairs is high, this indicates a high certainty that the current segment alignment, $a(s_i, s_j)$, is correct.

In the following, we provide brief definitions for close and distant reading:

- **Close reading** defines the process of carefully reading word-for-word and interpreting a passage to develop a deep understanding of the ideas contained in the text [225].
- **distant reading** provides an overview of the text by moving from an in-depth exploration of the individual components of the text to presenting the global features of the text(s) [24].

5.3.1 Alignment Preprocessing

In order to derive similarity measurements and recommend matches between corresponding translation segments, we employ a three-step preprocessing pipeline to convert text to numerical vector spaces. We first normalize the text, remove stopwords and sparse terms (optional), and tokenize the text (1). We then (2) generate various embeddings that are used to compute similarity measurements. Embeddings include term TF-IDFs (Term Frequency–Inverse Document Frequency), term IDFs (Inverse Document Frequency) [210,223], and contextual word embeddings (word2vec) [212]. After generating the embeddings, (3) we implement the similarity measurements to derive the matches between segments. For this, we use a selection of popular state-of-the-art distance and similarity measurements. The first three are most often used with TF-IDF and IDF embeddings [216], while the last one, hypothesized to be the best, utilizes the quality of word2vec embeddings [209]. We have defined the similarity measurement in 4.

5.4 Design of AlignVis

In this section we introduce the design of AlignVis and couple our choices with the requirements in Section 5.2. Our tool utilizes automatic alignments exploiting a preprocessing phase, discussed in Section 5.3.1. The design is composed of three main constituents, shown in Figure 5.1. An editor canvas (1) that enables the user to view and refine the machine-recommended alignments (**RA**, **RE**), (2) a post-edit area that provides an overview of the aligned translations (**RO**), and (3) a user options panel that enables the user to interact with the editor canvas and post-edit area (**RE**). A more detailed discussion of the design of AlignVis follows.

5.4.1 AlignVis Overview

This section provides an overview of our tool’s design components and how they address the requirements outlined previously.

5. AlignVis: Semi-automatic Alignment and Visualization of Parallel Translations

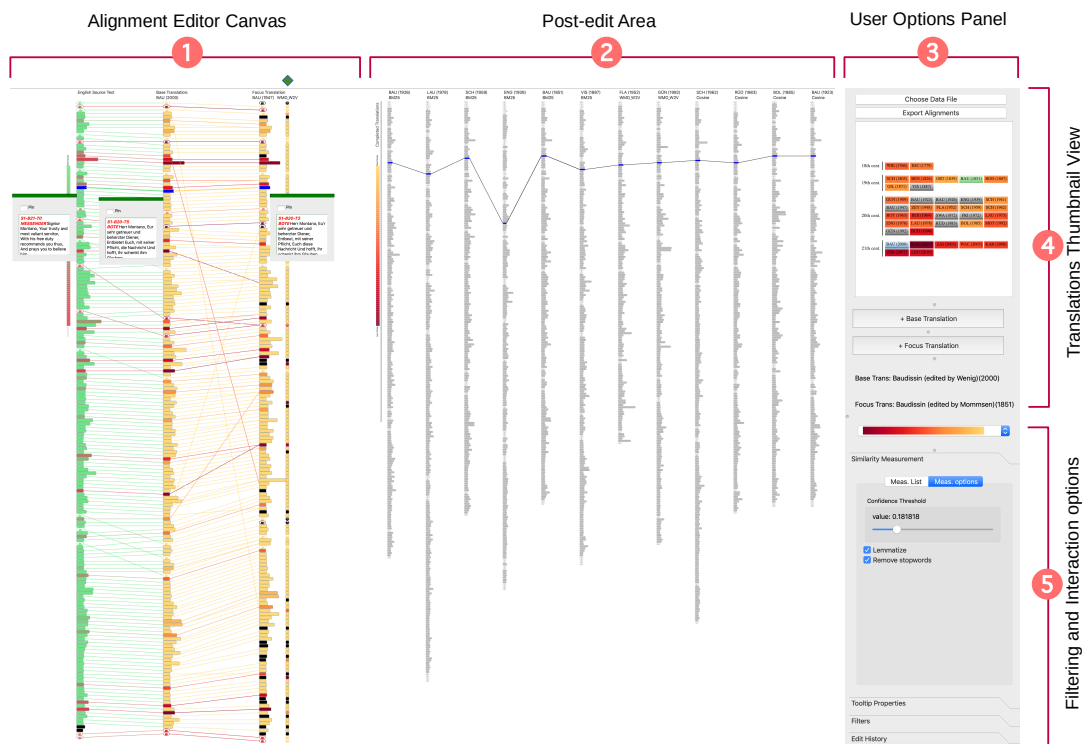


Figure 5.1: An overview of AlignVis. The alignment editor canvas (1) illustrates the original English text (T_E), the base German translation (T_B), and the focus German translation (T_F). This view shows the machine-recommended alignments between the German translations and enables manual refinement. The column indicated by a green diamond glyph shows the secondary measurement feature. The post-edit area (2) shows the processed translations; the user can explore the content and move items back to the editor canvas. The user options panel (3) provides filtering and interaction options and includes the translation thumbnail overview (4). The latter view shows the translations in chronological order of publication and enables the user to add translations to the editor canvas. The user options panel provides options that enable the user to interact with the design and change properties such as similarity measurements, filters, and color schemes.

Visual encodings in the alignment editor canvas: This view informs the current alignment process. The columns of rectangles, as shown in Figure 5.1 ①, depict the English text (T_E), the base translation (T_B) and the focus translation (T_F) from left-to-right respectively. The rectangles represent the text’s segments top-down as they appear in the text. The length of each rectangle encodes the length of the text and the edges illustrate an alignment between two segments. The edges and rectangles are colored to show the confidence of the alignment and can be filtered based on the confidence value. In this view, the user is able to examine the

machine-recommended alignments and refine them as necessary (**RA, RE**).

Design Justification of the alignment editor canvas: AlignVis uses distant reading and juxtaposition with explicit encodings of the translations to facilitate comparison between aligned sections [176] (**RA**). The process of alignment editing is a process involving close reading and it is more natural to the reader to read top-down. Juxtaposition and top-down order are consistent with previous tools. All interactions with the segments in this view are consistent and start with a right-click so as to accelerate the editing process and remain intuitive.

Visual encodings in the post-edit area: The second view is the post-edit area (Figure 5.1 ②). This view stores the processed translations (**RO, RN**). The most current translation is always placed on the left while the remaining translations are shifted to the right. Similar to the alignment editor canvas, the translations are illustrated using rectangles to depict the segments which, to use the space efficiently, are rendered 40% smaller than those in the alignment editor canvas with respect to both width and height. This view is linked with the alignment editor canvas: when the user highlights (using on-mouse-over) a segment, the aligned segments in the T_F and the aligned translations are also highlighted and a tooltip with the underlying text is displayed (**RO, RN**).

Design justification of the post-edit area: AlignVis presents the processed translations and links them with the alignment editor canvas to help and guide the user through the alignment process. This is a key novel feature that enables the user to align multiple translations (**RN**). They are ordered from left to right, with the most recent on the left, making it easier for the user to keep track of the processed translations. The post-edit view presents a distant reading of the processed translations which can guide the domain expert to similar translations while aligning the T_F .

The third component is the user options panel (Figure 5.1 ③). This provides a thumbnail view of all translations (Figure 5.1 ④) (**RO**). The user options panel incorporates interaction and exploration means to customize and update the editor canvas and post-edit area.

Design justification of the translation thumbnail view: This view presents the translations in chronological order to facilitate the search of a translation (**RO**). The translation thumbnails are colored based on the average confidence value of each alignment. The confidence value indicates the certainty of the similarity measurement (**RS**), explained further in Section 5.4.2. This color choice directs the user's attention to the level of alignment certainty for each translation (**RA**).

In the options panel, various filtering and interaction options (Figure 5.1 ⑤) are provided

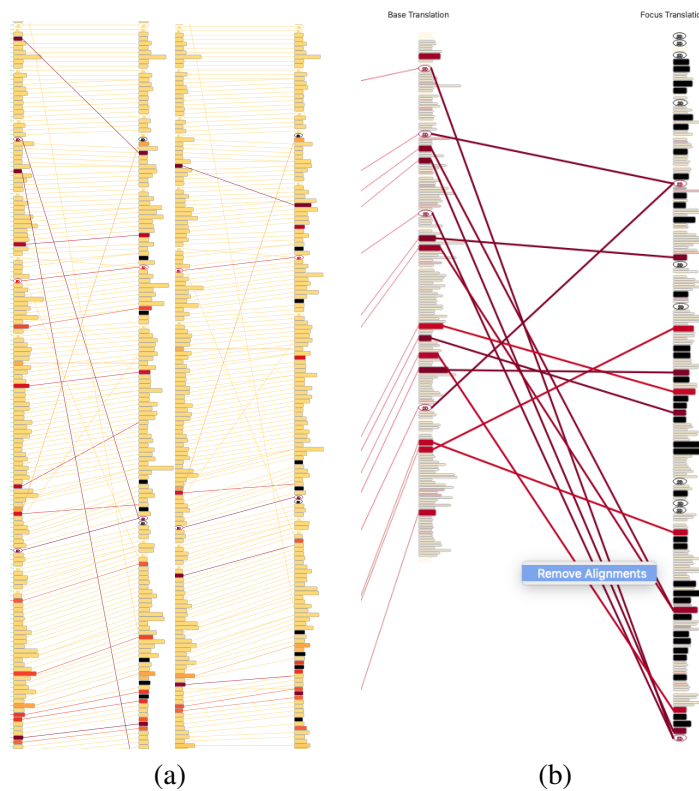


Figure 5.2: (a) On the left, the alignments without applying the confidence value threshold. On the right, the effect of applying a threshold ($\kappa = 0.75$). Some of the diagonal edges are removed. (b) A screenshot of the deletion action when the user selects multiple alignment edges.

that facilitate alignment and exploration (**RA**). The options are organized and grouped based on their objectives. For example, all options related to changing and updating the similarity metrics are placed in one group called “similarity metrics”, all options which filter the data or design items are in the “Filters” tab. From the options panel, the user can export the alignments into XML format that comply with the VVV’s project format. The options panel also provides the user with multiple preset color schemes [218]. The options presented in this panel were guided by our discussion and feedback with the domain expert.

5.4.2 Semi-automatic Alignment Exploration and Verification

Exploration and verification of the alignments contribute to (**RA**) and (**RE**) and are particularly important since the user of AlignVis does not necessarily have experience with what AlignVis offers and the text similarity measurements.

AlignVis color maps the edges between the T_B and T_F based on confidence values. The user can verify the performance of the similarity measurement via the overall view in the alignment editor canvas. AlignVis incorporates a feature that allows the user to set a confidence value threshold, κ . If the alignment confidence value is below κ , AlignVis chooses the index shortest distance to the T_B segment from the best three segment candidates produced by the similarity measurement algorithm. The index distance is the difference between the index of the T_B 's segment and the index of the T_F 's segment. Figure 5.2a illustrates the effect of applying the confidence value threshold κ . In the alignments on the left, we see there are some diagonal edges which are probably not correct, while the alignments on the right illustrate that these edges are reduced when applying a threshold of $\kappa = 0.75$.

Exploration and Verification in the Alignment Editor Canvas The alignment editor canvas also implements an action that facilitates the reading of the T_F as well as verifying the T_B alignments (**RA**). This is performed by selecting any segment, s of the T_F or T_B and then using the keyboard arrows to navigate to the next or previous segment. The segments and edges are highlighted while the user close-reads the translation, as shown in Figure 5.1.

The user can choose to add secondary similarity measurements to compare with the current measurement (Figure 5.1 ♦). This feature helps the user discover the best similarity measurement alignment if the first measurement fails (**RS**). This feature was added after exploring the variance between the similarity measurements. A secondary measurement could recommend and improve the alignment and accelerate the alignment process (**RA, RS**).

Exploration and Verification in the Post-edit Area The post-edit area can be used to verify the correctness of the machine-recommended alignment (**RN**). When the user selects a segment in the T_F or T_B , the post-edit area highlights the processed alignments and displays the original text if the user chooses. An edge is rendered between segments $s_1 \rightarrow s_2$ to show $a(s_1, s_2)$ and help capture a sense of the segment placements in the processed translations. For example, if the segment is aligned with two segments and the post-edit context view does not show this split alignment, this may indicate an incorrect alignment.

Sequential Alignments There are cases where the alignment between two translations is difficult even for domain scholars. Some translations are not stable and may be unfaithful with respect to the original T_E . They do not expect the machine-recommended alignments to

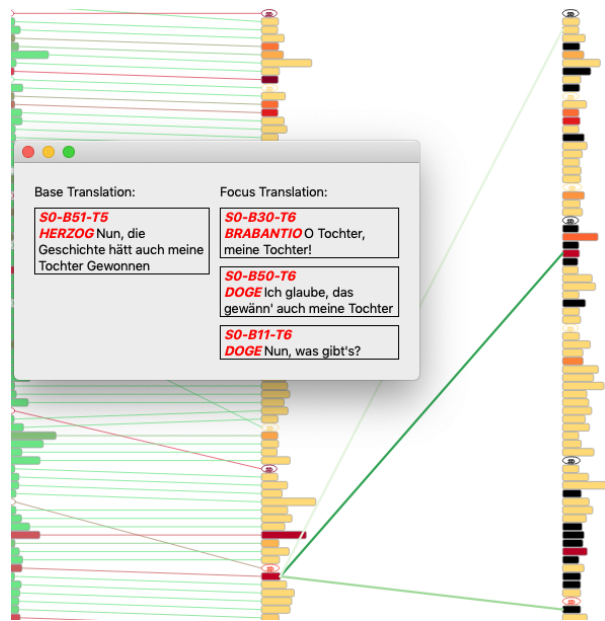


Figure 5.3: A screenshot of the top three candidate segments for a user-chosen T_B segment (**RE**). The candidate segment edges are rendered in green and the saturation of the color represents the rank of the candidate segments. The order of the segments is based on the ranking of each segment.

detect many correct alignments. Therefore, we support the domain expert to perform sequential alignments for such cases. See Section 5.3 for detailed explanation of sequential alignment.

In AlignVis, we implement this for aligning both the T_E with the T_B and T_B with the T_F . This follows the same convention: the user right-clicks on a T_B segment and from the menu selects “Create Sequential Alignments”. AlignVis follows the process described previously and applies a domain constraint to match the segment types when aligning. The constraint enables AlignVis to only align a speech segment with a speech segment and a stage direction segment with a stage direction segment.

5.4.3 Domain Expert Refinement

AlignVis enables the domain expert to refine and update the machine-recommended alignments (**RN**) and offers a selection of editing tasks.

Alignment addition: AlignVis enables the user to add a new alignment edge. The user can select a T_B segment, then right-click and choose “Begin manual alignment”. The alignment editor canvas then changes to edit mode. When the user chooses any segment in the T_F , a

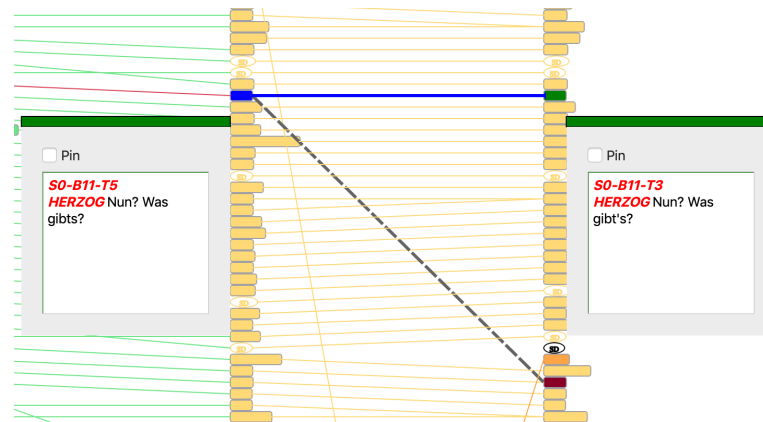


Figure 5.4: The alignment update mode: when the user chooses to update an existing alignment (**RE**). The original alignment is represented using a dashed gray edge, and a dynamic blue guide edge is rendered to indicate the new alignment.

dynamic edge is rendered and both segments are highlighted to facilitate the alignment process. The user can confirm the alignment by right-clicking and then selecting “Confirm alignment”. The user can exit edit mode by choosing “Clear alignment” or hitting the “ESC” key.

To accelerate the alignment process, the user can view the top three candidate segments based on the current similarity measurement by right-clicking on the base segment and choosing “Show candidates for alignment”. Then, as shown in Figure 5.3, a separate window is presented to close-read the best three candidates for the segment, ranked by distance. The saturation of the alignment edges represents the rank of each individual segment distance. After the user examines the original, the segment can be re-aligned with the best match. The best three candidates for the segment are the first three focus segments with the shortest distance to the base segment.

Alignment update: The user can update an existing alignment and change both the T_F or T_B segment correspondence. A dynamic blue guide edge is rendered to indicate the new alignment. In order to help the user update the connections and not lose track of the original segment, the original segment is highlighted using a dashed gray edge as shown in Figure 5.4.

Alignment deletion: AlignVis incorporates the action of alignment deletion. Consistent with the previous actions, the user may right-click on an edge and choose “Delete an alignment edge”. AlignVis also enables the deletion of multiple connections: the user can select multiple alignments and then “Delete alignment edges” to delete the selected alignments, as shown in Figure 5.2b. All editing actions are stored and can be undone by the user.

5.4.4 Selection and Filtering

AlignVis incorporates a number of interaction features to customize the alignment editor canvas and the post-edit area (RA).

Selection in the Alignment Editor Canvas AlignVis enables the user to interact with the visual design in various ways. When the user chooses a segment, all of the aligned segments in the post-edit area are highlighted and linked with edges while the other segments are rendered as context, shown in Figure 5.1. Customized tooltips have been designed to provide close reading of the segments, which the user can slide anywhere in the editor and pin in a specified location (analogous to a Post-it note) for further exploration, as shown in Figure 5.1.

Selection in the Translation Thumbnail View The translation thumbnail view in the user options panel (Figure 5.1 4) enables the user to add T_F s to the alignment editor canvas (RN). The user can select a translation by right-clicking and choosing “Add focus translation” or double-clicking on the translation thumbnail. This results in interactively sliding the current T_F to the post-edit area and replacing it with the new user-chosen T_F . The T_F can be removed from both the alignment editor canvas and the post-edit area. The user is able to change the T_B translation by selecting the translation thumbnail and clicking on “+ Base Translation” to add the T_B translation to the alignment editor canvas. When the user selects a new T_B , the current T_B is replaced.

Filtering of Translation Segments The user can apply filtering options to reduce the visual complexity (RA) caused by many segments. AlignVis offers two filters, the first of which excludes stage directions. A stage direction is a sentence which instructs the director and actors in a play. Researchers are not always interested in studying stage directions, so removing them can reduce clutter from the scene. Stage directions are clearly distinguished from normal speech, as can be seen in Figure 5.1 1. They are illustrated by ellipses and colored borders.

The second filter renders the alignment edges based on the confidence value. This option offers a range slider to enable the user to set the confidence value threshold κ . The filters option provides the user with dynamic feedback on the number of preserved and filtered segments.

When applying this filter, the alignment editor canvas highlights the segments and edges within the filter’s range. Edges and segments which are not within the range are not rendered in focus, as shown in Figure 5.2b.

5.5 Evaluation

AlignVis has been designed in close collaboration with a domain expert to design a solution that addresses the requirements outlined in Section 5.2. The following sections describe the domain expert feedback and provide a comparison with a standard alignment tool and computational and visual alignment tools.

5.5.1 Domain Expert Feedback

This section reports some of the domain expert feedback on the features of AlignVis.

Alignment Exploration and Verification Feedback The domain expert found the features AlignVis provides to enable the expert to explore and read helpful, stating that, *“The process of establishing and checking alignment is a process of reading the text, and AlignVis could help with that a good deal.”* Further, the domain expert stated that *“The alignment editor canvas provides a quick way to read and check the alignments.”* Exploration and reading is facilitated using the alignment editor canvas and the post-edit area which integrates close and distant reading in the same view. While experimenting with the two designs, the domain expert stated *“I like the way that looks right away.”*

The domain expert uses the edge color as an indicator to validate the alignment. In this way the user can explore and validate the alignment by the overall view of the alignment results (RO). The domain expert agreed that coloring the edges to indicate the similarity measurement certainty is a good idea, and that the overall view illustrates how much a manual alignment is needed for a specific T_F .

The domain expert emphasized that presenting the top suggested segments saves time (RA) as he needs to read the translation to find the other candidates. In addition, automatically adjusting the alignments to choose the shortest edges using the confidence value threshold is helpful and saves time exploring the alignments (RA). *“This feature sensibly prioritises the top few best candidates rather than presenting all possible candidates, speeding up my decision-making,”* stated the domain expert.

The secondary measurement is an effective way to see multiple measurements in the same view (RN). For example, the domain expert investigated the alignment between the segment *“Saal im herzoglichen Palast”* in the T_B Baudissin (2000) and the segment *“Ein Beratungszimmer”* in the T_F Baudissin (1962). d_{\cosine} failed at detecting this alignment, but the d_{wmd} detected

the correct one because it utilizes semantic word embeddings. Both of the words “*Saal*” and “*Zimmer*” have similar meaning that indicates a room.

The Post-edit Area Feedback One of the most important advantages of the post-edit is that it presents distant reading of the processed translations. The domain expert considered distant reading in both the alignment editor canvas and the post-edit to be beneficial as it helps in the comparison task between translations at a global level (RN). Highlighting the corresponding alignment across the post-edit area is also useful as the user explores and validates the alignments. “*Doing the alignment process is not just preparation, however, as you do the alignment you discover things about the texts that you want to investigate more. This view facilitates this as I am interested in the other texts and I want to be able to retrieve them,*” the domain expert stated (RA).

Domain Expert Refinement Feedback The domain expert found it simple to refine the machine-recommended alignments. He appreciated the ability to perform a one-to-many and many-to-one alignment for the first time, and that all of the alignments actions are in one place when a segment is right-clicked. He stated: “*From a design point of view it is very good compared to other alignment tools that I have had to deal with, this is quick, painless and easy to do*” (RA).

The domain expert expressed approval of the update function when AlignVis particularly encodes the original alignment while choosing another alignment (Figure 5.4). The dashed gray edge that represents the original alignment has the advantage that the user can see what is to be changed.

Selection and Filtering Feedback The stage direction filter can be used to reduce design complexity. This accelerates work as it saves time reading the speeches without the stage directions interrupting the reading and alignment refinement process (RA). In addition, the two-range confidence value filter is beneficial as it reduces the edges between the T_B and T_F , especially when the T_F is not stable.

The domain expert stated that “*The option to filter alignments by level of confidence is useful in cases where the user has found that the low-confidence suggestions are of no use (they are all false), so the user can save time and attention by excluding/deleting them. On the other hand, there can be cases where low-confidence suggestions are worth examining specifically, e.g. where the overall confidence level is low.*” (RE).

Tasks supported	LF-Aligner [151]	ViTA [146]	iTeal [116]	AlignVis
Close reading	×	×	×	×
Distant reading		×	×	×
Multiple alignment (n>2)	×		×	×
Post-edit interaction	×		×	×
Incorporating similarity measurement		×	×	×
Testing different similarity measurement				×

Table 5.1: A comparison between AlignVis and the related work.

The domain expert thought that the selection of translations is intuitive as the translation thumbnail view presents them in chronological order of publication and is consistent with the other views, using right-click to add and remove a translation (**RA**).

Moveable and Pinable Tooltips The domain expert admired the design of the tooltip as it helps him pin the tooltip and move it around, which facilitates the comparison tasks with other segments. The domain expert uses this a lot as he reads and explores the translations (**RA**, **RN**). “‘*Excerpting*’ is a fundamental technique in humanities research – meaning, we read something, and select a quote/excerpt from it, or paraphrase the selection/excerpt in our own words, and we make a note including the excerpt and a reference. You can call this ‘manual text mining’. In exploring and comparing translations, it’s very helpful to be able to do this inside the application,” the domain expert stated.

5.5.2 Comparison With the Computational and Visual Alignment Tools

Table 5.1 provides a comparison summary of the tasks and related work, incorporating only related work that visually or computationally generates alignments. Our comparison is based on six supported tasks derived from the requirement analysis discussed in Section 5.2. The first two tasks are (T1) close and (T2) distant reading. Close reading involves the process of careful word-for-word reading and interpreting a passage to develop a deep understanding of the ideas contained in the text [225]. Humanities scholars appreciate access to the raw text [226] and so this increases trust in the implemented approach [24]. Distant reading, conversely, illustrates the global features of the texts using computationally and analytically abstracted visualization [24].

Most of the tools in Table 5.1 implement close reading solutions. However, LF-Aligner does

not provide distant reading of the aligned texts.

Humanities scholars spend a great deal of time aligning multiple versions or translations as most tools present only one-to-one alignment solution. Table 5.1 shows that most of the compared related work incorporates (T3) alignment of multiple texts, but ViTA is limited to only two texts.

Post-editing of the results is a feature valued by Humanities scholars. The scholars' knowledge intervention is always an important add-on when preparing and aligning texts. Most of the compared related work provides interaction to support (T4) post-editing of the results, with the exception of ViTA which does not support human post-editing.

LF-Aligner uses different measures to sequentially align texts and does not (T5) incorporate similarity measurements. The other related work computationally aligns multiple texts using similarity measurement algorithms. AlignVis enables the user to (T6) test multiple results from varying similarity measurements and visualize them to support exploration and analysis. The other related work does not integrate the ability to use different similarity measurements beyond the implemented algorithm.

5.5.3 Comparison With a Standard Alignment Tool

LF-Aligner We offer a comparison with LF-Aligner because this is what the domain expert uses in our case. The domain expert has tried other tools and found LF-Aligner the most useful for the purpose of aligning related texts. However, the limitations of LF-Aligner and other tools that Humanities scholars commonly use inspired this project.

In LF-Aligner, the user uploads a corpus and the software performs an initial automatic sentence segmentation and alignment using an algorithm which primarily inspects sentence lengths in sequence. The success of this process is varied, particularly when dealing with our German Shakespeare corpus. In this case, LF-Aligner's results are very unreliable due to structural differences and considerable variation between sentence length sequences in different versions.

LF-Aligner's manual alignment correction interface is a tabular display which fills the screen: parallel full texts are displayed as columns and segments are displayed as rows. The software's initial segmentation and alignment can be modified manually by the user using a small set of keyboard controls to split or merge, insert or delete cells.

The simplicity of LF-Aligner's interface is a benefit to most Humanities scholars as nothing distracts from the view of the texts, which is the scholar's main focus of interest. LF-Aligner is

therefore suited to close reading. The user combines the segmentation and alignment process with detailed inspection of the texts. The alignment correction process is slow, but offers the scholar new information and subsequently knowledge which contributes to the interpretation of the texts.

One disadvantage of LF-Aligner is that it cannot cope with transposition: cases where a segment sequence $\{s_1, s_2\} \in T_1$ aligns with a sequence $\{s_2, s_1\} \in T_2$. The user must manually reorder the sequence in one of the texts, but this creates an inaccurate representation of the original text – a loss of significant information.

The primary disadvantage of the LF-Aligner interface is that it offers no distant reading, although the segmented and aligned manually edited corpus can easily be exported into other systems which do offer distant reading. However, the segmentation and alignment process would be more efficient if the user could shift back and forth between close, full text view and a distant overview of text structures and alignment patterns. In the LF-Aligner interface, depending on screen settings, the user sees only the equivalent of one to two printed pages on the screen so it is impossible to obtain an answer to questions requiring an overview, such as: Which passages exhibit a continuous one-to-one alignment? Which passages have no alignment or multiple alignments? Which passages align differently in different versions? This sets limitations on the amount of information the user can gain from the alignment process.

AlignVis implements the feature of distant viewing in both the alignment editor canvas and the post-edit area. LF-Aligner, on the other hand, does not support support distant view of the texts. The default view of a text is a distant view, representing a sequence of segments as a narrow column of blocks. Visual features of the blocks represent segment types (e.g. speech text or stage direction) and computed features of the segment (length, etc) and its alignment(s). This system of representation enables the equivalent of multiple printed pages to be represented on a single screen, affording a rapid overview of corpus characteristics of interest to the researcher: lengths, segmentation structures, patterns of alignment, and passages of interest for editing purposes.

The focus of interest here is not the automation but the display and manual correction options. The automated results are far better than the results obtained by LF-Aligner, leading to significant time saving in manual correction.

LF-Aligner's auto-alignment implements the hypothesis that a sequence $\{s_1, s_2, s_3, s_4\} \in T_1$ will normally align with $\{s_1, s_2, s_3, s_4\} \in T_2$. This factor could have more influence on AlignVis's metrics. The steep diagonals alignments are certainly false alignments, but the

ability to accommodate transposed alignments is an advantage.

A crucial issue for the Humanities user is not so much the performance of automated alignment but rather the ease of inspecting and correcting alignments. Here AlignVis holds several major advantages over LF-Aligner. These include distant viewing of the full corpus in the options panel thumbnails, with color coding indicating confidence levels, guiding the setting of editing priorities; distant viewing of texts, segments, and alignments for base text and focus text; distant viewing of aligned texts in the post-edit area, with easy switching of the T_F ; visual representation of confidence values for alignments to guide editing priorities; ease of obtaining a close view of segment text for exact reading; and ease of manual correction.

The user can rapidly scroll through T_B and T_F simultaneously, speed-reading successive segments while visually checking the defined alignments. This resembles the user experience in LF-Aligner, except that by default, only one pair of aligned segments is in view, alignments are represented by an edge rather than as contents of a row; and the edge can be selected for rapid editing. The visual encoding of confidence values enables the user to scroll very rapidly where confidence is high, whereas in LF-Aligner each aligned pair must be visually checked.

Keyboard shortcuts could be implemented for frequently used commands. The AlignVis user focuses on one segment pair at a time: compared with LF-Aligner, this speeds up the process in ‘cruise mode’ where many successive alignments are one-to-one. It is simple to ‘skip’ down or back up columns. Where alignment problems occur, the option to ‘pin’ segments of interest is helpful for close reading, which often requires close comparison of segments across different passages of a text. This helps ensure that the alignment task is a knowledge-gaining process for the scholar.

Overall, AlignVis is a promising design, combining text mining and language processing affordances with a practical solution to supporting the labor-intensive task of exact segment alignment. AlignVis makes this task much more efficient than existing tools while simultaneously supporting the potential for alignment checking and correction to be an integral part of the scholarly process of understanding and interpreting texts.

5.6 Chapter Summary

In this chapter, we presented AlignVis, a tool that combines interactive visualization with domain knowledge intervention to facilitate the alignment of parallel translations. AlignVis was designed in close collaboration with a domain expert to implement five requirements (Section 5.2). AlignVis was evaluated through domain expert feedback and comparison with a standard,

widely-used alignment tool as well as computational and visual alignment tools. Future work and limitations are discussed in Chapter [8](#).

Chapter 6

TransVis: Integrated Distant and Close Reading of Othello Translations

“... whereas what we really need is a little pact with the devil: we know how to read texts, now let’s learn how not to read them”

–Franco Moretti (1950-)

Contents

6.1 Introduction and Motivation	102
6.2 Definitions and Background	103
6.3 Design Requirements and Tasks	105
6.4 TransVis’s Design	107
6.5 Evaluation	119
6.6 Chapter Summary	129

This chapter aims to address the fourth research objective [Ob4]. In this chapter, we introduce TransVis [2], a novel integrated visual application to support distant and close reading of a collection of *Othello* translations. We present a new interactive application that provides an alignment overview of all the translations and their correspondences in parallel with smooth zooming and panning capability to integrate distant and close reading within the same view. We provide a range of filtering and selection options to customize the alignment overview as well as focus on specific subsets. Selection and filtering are responsive to expert user preferences and update the analytical text metrics interactively. Also, we introduce a customized view for close reading which preserves the history of selections and the alignment overview

state and enables backtracing and re-examining them. Finally, we present a new Term-Level Comparisons view (TLC) to compare and convey relative term weighting in the context of an alignment. Our visual design is guided by, used and evaluated by a domain expert specialist in German translations of Shakespeare.

6.1 Introduction and Motivation

Text visualization is a popular subfield of information visualization due to the rapid increase in digital text data over the last two decades [3,22]. In this project, researchers with an expert background in the Arts and Humanities prepared 38 translations of Shakespeare’s play *The Tragedy of Othello, the Moor of Venice* (1604). The translations were originally written over a span of 244 years from the Christoph Martin Wieland translation [227] in 1766 to the Christian Leonard translation [228] in 2010. Our data set contains the 38 translations as well as the English base text of a sample of the full play –Act 1 Scene 3.

Based on the notion of close and distant reading of texts [225,229], we attempt to create a novel interactive visual design that combines distant and close views of the parallel translations of Shakespeare’s *Othello*. Guided by the visual information seeking mantra [96], inspiration from previous work on this topic, and close collaboration with the domain expert we derive six requirements and five tasks that our application must support (discussed in Section 6.3). The proposed visual design is guided closely and reviewed by a domain expert from Arts and Humanities.

Contributions: In this chapter, we contribute the following:

- We support integrated distant and close reading in the same view and implement them with smooth zooming and panning.
- A novel visual design that supports comparison of an arbitrary number of parallel translations.
- Customized mechanisms for rapid and interactive filtering and selection of a large number of German translations of Shakespeare.
- Interactive and dynamic analysis of similarity metrics to support comparisons and analysis of customized parallel translations.
- Examples, detailed observations, a case study, and domain expert feedback from a specialist in German translations of Shakespeare.

The rest of this chapter is organized as follows: Section 2 introduces and defines the most important terms used in this chapter and introduces the parallel data as well as the similarity metrics. Section 3 discusses previous work related to our approach and the challenge domain. Section 4 outlines the design requirements and tasks that our approach supports. Section 5 introduces our proposed application components. Section 6 provides the domain expert feedback and case studies. We finish with conclusions and future work directions.

6.2 Definitions and Background

This section provides background information on terminology, the text data, and translation meta-data.

• **Definitions:** In this section, we explain the notions of close and distant reading. **Close reading** generally defines the process of carefully reading word-for-word and interpreting a passage to develop a deep understanding of the ideas contained in the text [225]. Close reading signifies the critical analysis of small and specific components of the text over the general theme. Close reading may involve annotation and highlighting techniques to increase the comprehension process. In a literary context, close reading is defined by Nancy Boyles [225] as “*reading to uncover layers of meaning that lead to deep comprehension.*” The practice of close reading is inherently subject to the reader, and interpretations of the text vary according to readers and context [230].

On the other hand, **distant reading** aims to provide an overview of the text by moving from an in-depth exploration of the individual components of the text to presenting the global features of the text(s) [24]. In contrast to close reading, distant reading is technically a more objective process because most of the context is hidden and the reader is left with the result of computationally and analytically abstracted visualization.

Throughout the paper, some special terms are used. A **segment** is a meta-data object. A segment can be any continuous sequence of words within a text, but generally, it is a meaningful unit. The text of *Othello* is a play for theatrical performance. Like most play texts, it contains three kinds of segment: ‘speeches’ (words to be spoken by actors), ‘speaker identifiers’ (words indicating the character in the play who speaks), and ‘stage directions’ (words which instruct the director and actors). A speech is always preceded by a speaker identifier. A speech can consist of one to many sentences, and one to many words. In our dataset, all three kinds of text are predefined as segments.

An **alignment** is a meta-data object which links a given translation segment with its corresponding text in the English base text. In our dataset, alignments have been created (in a machine-assisted manual process) between each segment in the English base text, and each segment in all the translations which has a meaning which corresponds to the base text segment. Some translations omit, transpose, and add new words, sentences, and even speeches, so aligning is a complex task. However, the majority of translations in our dataset are ‘faithful’, ‘close’ and complete translations. With those, making speech-by-speech alignments is relatively straightforward.

- **Similarity Metrics “Eddy and Viv”:** The Eddy and Viv metrics were introduced by Cheesman *et al.* [56] to quantify how a given base segment (English in our case) is interpreted and translated between parallel texts (German in our case). Eddy characterizes the translated segments in terms of distinctiveness. A higher Eddy value indicates higher dissimilarity from other translations.

The word forms are important at this stage. In the previous work, the formulations of Eddy and Viv do not consider advanced linguistics algorithms to reduce inflectional forms of words such as lemmatization or stemming [208]. This is due to the nature of the German language which is considered inflected. Special challenges appear with German Shakespeare texts due to the use of antiquated language and poetic orthography. However, we build up a lemmatization dictionary for our corpus using Cascaded Analysis Broker “CAB” [231] which is developed for the German Text Archive (Deutsches Text Archiv, DTA) [232]. CAB is an HTTP-based web service morphological normalization tool developed for historical German text especially for the 18th and 19th centuries.

As a dimensionality reduction algorithm, each segment is represented by a fixed length of vector of word weights TF-IDF (term frequency–inverse document frequency) [210, 223]. Then, the similarity coefficient between segments vectors can be obtained by the Euclidean Distance between each pair of segments. Euclidean distance is usually the default metric used to measure the distance between two points or vectors. It is the default distance metric used with the K-means algorithm [214].

After obtaining the similarity values between each pair of aligned segments, a weight value “Eddy value” is computed by averaging the sum of similarity values between each pair, such that:

$$Eddy(S_i^j) = \frac{\sum_{k=1}^n \|S_i^j - S_i^k\|}{n} \quad (6.1)$$

where S_i^j denotes segment i in translation j and n denotes the number of translations.

Viv, on the other hand, is the average pairwise distance between every two segments projected on the base segment, also known as the diameter of a cluster [233]. Viv represents the stability of the base segment. A high Viv value indicates low stability (high variability) which means the segment's translations vary considerably between authors. We compute the Viv value for a given base segment i as:

$$Viv(i) = \frac{\sum_{k=1}^n Eddy(S_k^i)}{n} \quad (6.2)$$

High Eddy and Viv values are interesting to arts and humanities researchers because if a translation has high Eddy values, it indicates that the translator is working in a more unusual, possibly a more creative way relative to others, maybe interpreting the text in a new way. This might be due to circumstances such as historical changes in the language and/or the culture, as a result of political, economic and social change for example. It may be due to an individual translator developing their own new approach to the translation task. Or it may have to do with a new market developing for a new kind of text –in this case, new kind of theatrical drama. High Viv values indicate which base text segments are associated with variation among translations, which enables research into the textual factors (such as complexity, ambiguity, polysemy, semantic salience or affective intensity) which may provoke translators to deviate from one another.

Eddy and Viv metrics can be computed interactively and dynamically, that means every user customization derives new similarity metrics.

6.3 Design Requirements and Tasks

The original question that was posed for the application to address is: how can the variation between any number of parallel translations of a given source text be represented visually so as to enable users (a) to identify overlaps, absences, additions, and variation between parallel translations and (b) to study the findings of various kinds of algorithmic, comparative text analyses. So, the rationale behind our visual design is to enable users to interactively explore the translation collection to answer this question. To achieve this, we established and incrementally refined a list of requirements. The requirements that our proposed implementation fulfills are as follows:

R1 An application that enables comparison of translated parallel text.

- R2** A visual design that supports both close and distant reading.
- R3** A layout that supports close reading for further detailed analysis.
- R4** A visual design that considers stable versus unstable translations.
- R5** Interaction that enables customization of the translated texts.
- R6** Interaction that supports general exploration and analysis.

To action this list of requirements, we established a list of associated tasks for implementation. We derived five main tasks based on the typology of visualization tasks by Brehmer and Munzner [234] in order to achieve the aforementioned requirements and motivate our visual design:

- T1** In this *discovery* process, the user should be enabled to *explore* the alignment overview in order to *identify* a region of interest (**R1**). [discover→explore→identify]
- T2** After identifying a region of interest (**T1**), the user may *navigate* the space leveraging smooth zooming and panning. As the user navigates, multiple levels of details are *aggregated* and rendered (**R2, R3, R6**). [T1→navigate→aggregate]
- T3** As the user performs **T1** and/or **T2**, the user may apply different *filtering* and *selection* tasks to assess the exploration task (**R5**). [T1/T2→filter/select]
- T4** As the user performs **T2**, the user may *select* a segment to obtain details-on-demand (i.e. a close reading view) (**R3**). [T2→select]
- T5** As the user performs **T1, T2** and/or **T4**, the user may perform interactive *comparisons* of the parallel translations exploiting meta-data based on the alignment of speeches and text similarity metrics (**R4, R6**). [T1/T2/T4→compare]

We relate to these tasks in the discussion of our proposed design (Section 6.4). Our design is influenced by the visual information seeking mantra by Shneiderman [96] that suggests providing an overview first, then zooming and filtering options, and finally details-on-demand. The design is also influenced by previous work on this topic which aims to align texts side-by-side to support comparison tasks. Additionally, the design is also guided by careful collaboration with the domain expert.

6.4 TransVis's Design



Figure 6.1: The alignment overview (A) shows the parallel alignment of translations with the original base text. The highlighted path in (A) shows the distant alignments of a segment of the “*Othello*” speech starting with “*I ran it through...*”. In the left-bottom, a zoomed-in view magnifies the curved edges. Window (B) shows the options panel that facilitate exploration of the collection and comparison of translations. View (C) is a close reading view (the detailed view) that corresponds to a user-selected speech and each aligned speech. Window (D) shows the Term-Level Comparison (TLC) view. The zoomed-in rectangle is part of the figure, not of the visualization itself.

In this section, we introduce our proposed interactive visual design of parallel translations and relate our choices to the tasks from Section 6.3. Our system is composed of four main constituents starting with an overview.

The first window offers the **alignment overview** of parallel translations of Shakespeare’s *Othello* (T1, T5). It provides a general context for understanding the collection and conveys the

whole dataset in one visual layout. Furthermore, it leverages interactive capabilities to enable the user to explore and find interesting patterns and features within the collection. The alignment overview allows users to examine significant, larger patterns in the translations which are not readily viewable from narrow or detailed views. Window (A) in Figure 6.1 shows the distant reading view of 38 parallel translations aligned with the base text. The curved edges between translations depict alignments between speeches. The zoomed-in portion in Figure 6.1 shows a close view of the curved edges and segments.

Design justification of the alignment overview: Our data is high-dimensional. We present 38 texts and each text contains multiple levels of abstraction (term, segment, speech, and books). Thus, we present a parallel view of translations and in each translation we present the encapsulated structure. Juxtaposition supports visual comparison of alignments intuitively, when comparing different manuscripts the user places them spatially next to each other and performs comparison. Also, we incorporate a number of exploration techniques to help the user customize the alignment overview to their preference such as by zooming, filtering, or selecting.

The second main component is the **options panel**, shown in Figure 6.1 (B). It provides the user with a range of layout functions in order to facilitate exploration of the collection and comparison of translations (**T2**, **T5**). The user can perform a query-based search and the results are visualized using focus+context in the main window (A). It features a number of tabs that support different tasks. The first tab is for color properties which enables the user to modify the color mapping schemes and color-map different properties of the speeches, such as individual speech length and language similarity values. The second tab is the options tab where the user may alter the properties of the visual design, such as the order of the translations and length of time tooltips are shown over speeches. The third tab is the filter tab. Different filters are provided for the user to reduce the complexity of the visual design, such as stage direction and speaker filters. These filters are discussed in Section 6.4.2.3. The last tab is the time-oriented thumbnail view. In this section, all of the translations are depicted using color-coded thumbnails. If the translation is shown in the main window (A), the thumbnail is green otherwise it is red. These filtering and selection options are the result of our feedback sessions with the domain expert. This is described in more detail in Section 6.5.1.

Design justification of time-oriented thumbnail view: Most of the user options are implemented in close collaboration with the domain expert. The thumbnail view clusters the translations chronologically which makes it intuitive and quick to explore and navigate. Adding

and removing translations is simply performed either by toggling a translation on or off or dragging and dropping. It is important to let the user customize the starting point when there are so many translations. The view aids the user by visually informing him of the selected translations (green buttons) and deselected ones (red buttons).

The third main component of the system is the **detailed view** which is the focus subset that the user is interested in after performing filtering and selection (**T4**). As shown in Figure 6.1 (C), the detailed view shows a close reading of the user-selected speech along with the aligned speech translations. It stores both the interesting aligned segments (path) and the alignment overview state, so the user is able to revisit any previously selected paths for further analysis (**T5**). Also, the text in this view is accessible to user and can be copied to be used beyond our system. The detailed view is discussed in Section 6.4.1.2.

Design justification of the detailed view: It is recommended to allow the user to have access to the actual text particularly when developing visualization for literary scholars [26,168]. This close reading gives complete access to the text for further analysis with other software. This view aligns the segments in a compact and simple context for further analysis. It incorporates a list of previously selected paths to facilitate comparison between paths and also to enhance the user experience by saving the actions history (**T5**). We provide this addition of close reading to support close and distant reading simultaneously without losing the user's first (or previous) choice of speech for close reading.

The fourth component is the **Term-Level Comparison (TLC)** view (Figure 6.1 (D)). The TLC is an interactive analytical tool that assigns weights to each word in each user-chosen segment aligned with a base English segment. We use TF-IDF as a term-weighting to measure the significance of each original word or lemma in a segment with respect to the word occurrence in the whole corpus (the segments aligned with the base English text in our case). The view clearly justifies the Eddy metrics and signifies the terms (original or lemmas) that define segments. Also, it aids the user in finding terms and translation variance (**T5**). To show this plot, the user can drag an interesting path, usually because of an Eddy value distinctiveness pattern, then drops it onto the TLC view area to see the word's contribution in this path. This view is also motivated by the on-going discussion with the domain expert requesting close analysis and details.

Design justification of the TLC view: As TLC view could be considered as parallel coordinates, it presents strength when exploring and processing multi-variate data [235]. Such techniques are useful to explore anomalies in the data even without an extended outlier-detecting

mechanism [236]. As a result of the TF-IDF, the commonly used terms are assigned lower values and vice versa. Thus, the distinctive terms will stand out very clearly in the TLC view which is the main motivation behind this design. To overcome the limitation of the view when there are cluttered lines, the view provides a list of words in alphabetical order. When the user selects any word, the TLC highlights the line corresponding to this word. The user also can enable brushing to highlight multiple lines. The color opacity of the unselected terms is reduced in order to remain as context. The terms list updates to reflect the brushed terms. The terms list also assigns the same color of the term's line to the list item to visually identify the line's correspondence as shown in Figure 6.3. The colors of the lines are automatically generated to uniquely assign a color to each term.

6.4.1 Visual Design Factors

We separate the primarily visual and interactive design features to facilitate reading.

6.4.1.1 Filtering based on Derived Alignments

The dataset contains alignment information between each speech in each translation and the English base text. However, it is useful to have an alignment between any two arbitrary translations. We derive meta-data that aligns two arbitrary translations with respect to the base text. We align two speeches from different translations if they correspond to the same base English speech.

With the alignment meta-data, we have the alignment between the source text segments and the translations' segments. The derived alignment is the alignment between a translation and another translation. If the translation is not adjacent to the source text, we derive the alignment between two segments (a and b) of two translations (A and B) if b has an actual alignment and a has an actual alignment, we connect them. The path of alignment disconnects if one of these conditions break. For instance, if b has an actual alignment with the source segment, and a which belongs to translation A that occurs before translation B does not have an actual alignment, the path of alignment disconnects. This mechanism is implemented as a user option, and as explained it results in filtered alignments as the path moves left-to-right. However, the user-chosen order of translations affects the results in the alignment overview and the user needs to choose the sorting function carefully.

This mechanism can be useful, particularly when comparing translations and as a filtering option (T3, T5). However, in some cases, it might not yield results when the translations are

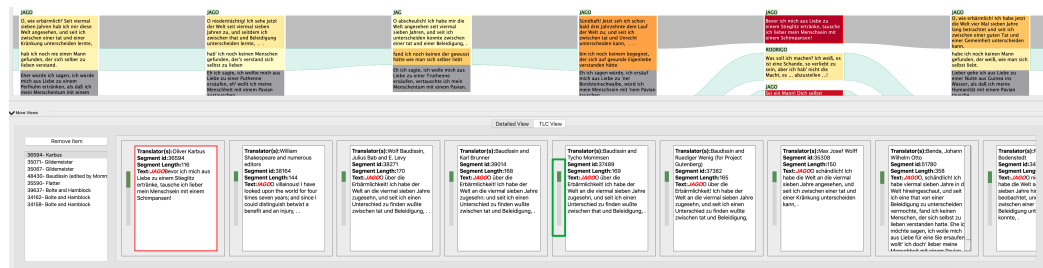


Figure 6.2: The detailed view shows the user-selected speech highlighted using a red border and the aligned speeches ordered consistent with the translations appearing in the alignment overview. Each speech is paired with with a colored bar (annotated using a green border) to indicate the similarity distance. In the bottom-left corner, a list of all previously selected speeches. If the user selects any speech from the list, the corresponding text is highlighted in the alignment overview and the edges of alignments are presented above.

not related, such as when translations stem from a different era or author.

6.4.1.2 Detailed View For Close Reading

In addition to smooth zooming in the alignment overview that enables the user compare specific speeches, the user is able to select any speech to analyze and compare it along with the aligned speeches in a dedicated detailed view as shown in Figure 6.2 (T4, T5). The detailed view provides another close reading option for the speeches such as translators names, speech identifier and Eddy value (discussed in Section 6.2). The close view shows the user-selected segment using a highlighted red border and each aligned speech in a scrollable window that is easy to read and investigate (T5). Each speech is paired with a colored bar showing the similarity distance to indicate the distinctiveness between aligned speeches. The longer the bar, the more distinctive the speech is with respect to the other translations.

In order to improve the user experience and to ease the exploration and analysis task for the researchers, we maintain a history of all previously investigated speeches in a list as shown in Figure 6.2 (bottom-left corner). When the user revisits any of the previously examined speeches in the list, the corresponding speech and the edges of alignments are highlighted in the alignment overview. The interactive history list identifies each user-selected alignment using the segment identification and the translation name such that the user may remember their own user provenance with respect to the alignment selection.

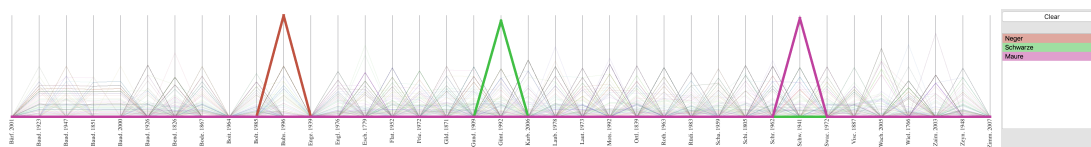


Figure 6.3: An example of aligned segments of the base English segment starting with “*Thou art sure of me...*” depicted by term-weighting matrices using the TLC view. In this example, the user brushes the three peaks that reflect the distinctive translations of the word “*Moor*” in the context: “*I hate the Moor*”. The highlighted translations are “*Neger*” in Buhss (1996) [29], “*Schwarze*” in Günther (1992) [30], and “*Maure*” in Schwarz (1941) [31]. The terms list reflects the brushing result and assigns the same colors to the terms. The user-chosen terms are rendered in the focus while the rest are rendered as context.

6.4.1.3 Term-Level Comparisons (TLC) View

The TLC View illustrates the weighting of each term using line charts. The y axis shows the normalized weighting of the terms. Along the x axis, we render vertical lines to indicate the translations. In this view, we can explore each term weight across all translations. The TLC view (Figure 6.3) facilitates comparisons and analysis tasks (T5). For the term-weighting, as discussed in Section 6.2, we use the TF-IDF weight for each word.

Alignment overview with TLC view: The rationale of this view is to observe and explore the aligned segments in the term level and highlight outliers and uncommon terms. The TLC shows the terms that contribute to the pattern discovered in the alignment view when using the similarity metrics.

To explore aligned segments, the user can drag-and-drop any segment of the path from the alignment overview into the TLC view. A list of the terms is also presented on the right. If the user selects any word in the list, the corresponding line is highlighted, and vice-versa.

In Figure 6.3, three distinctive terms are highlighted in the TLC view: “*Neger*”, “*Schwarze*”, and “*Maure*”. These translations correspond to the word “*Moor*” in the context: “*I hate the Moor*.” These terms are illustrated in the three brushed peaks shown in the figure. The terms list shows the only selected terms and the color facilitates identifying the correspondence between the lines and the terms. This selected terms are rendered in focus and the rest are rendered in context. See the domain expert feedback in Section 6.5.1 for further discussion on this.

Lemmatization (as discussed in Section 6.2) is a normalization algorithm used to reduce inflected words and identify the base word [208]. We use CAB [231] to create a dictionary of the lemmas for our corpus. Processing such an antiquated language is challenging due to many

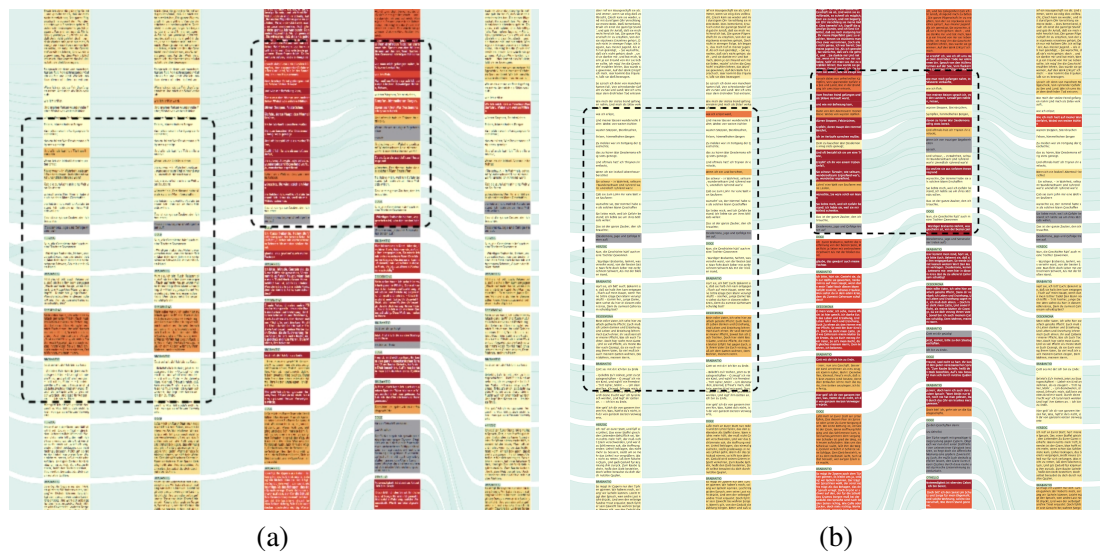


Figure 6.4: (a) An example of two corresponding sub-sets of six editions of Baudissin (1832) [32]. The variation of colors in the dashed rectangle in the original version (a) is clearer than the corresponding rectangles in the lemmatized version (b).

missing terms.

Example–lemmatized versus original text: We compare the results between a lemmatized and non-lemmatized version of the translations. Our corpus includes several editions of the classic, canonical Baudissin translation, first published in 1832 and often re-published (100s of times so far): printed editions from 1851, 1923, 1926, and 1947, and a digital edition from 2000. In Figure 6.4, we can see in the original version (a) that the variation of colors in the dashed rectangles is more distinctive than the corresponding areas in the rectangles in the lemmatized version (b). The variation in (a) is clearly a result of inflected words that share the same lemma.

Although, technically, TLC view is not a parallel coordinates view, it inherits some of its limitations. The TLC view can be difficult to interpret and explore in the case of long segments due to over-plotting. Much research focuses on interaction techniques to overcome the visual clutter challenge in parallel coordinates plots by reducing the dataset or by reducing or modifying the order of the dimensions [237]. In our case, the combination of TF-IDF as a term-weighting metric causes the common terms to clutter in the areas of low x values. However, as this can be problematic when searching for common terms, it helps to identify rare vocabulary in each translation.

The alphabetically-ordered word list that accompanies the view can also help the user find

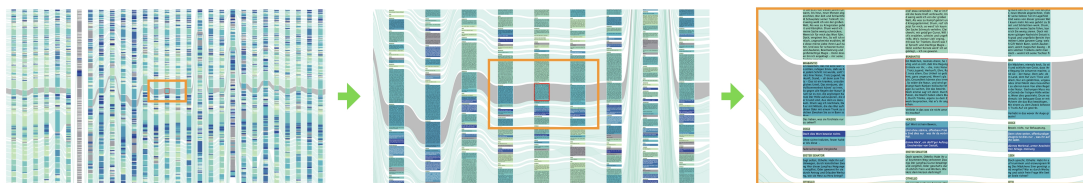


Figure 6.5: An example of three levels of integrated zooming and the smooth changes to the level of detail. (left) Distant reading without zooming. (middle and right) close reading after the user zooms in. The textual content of the speeches fades in smoothly at an increasing level of detail. The grey curved path indicates a user-selected alignment.

term line by clicking on a word and highlighting the corresponding line. The stop words removal, or lemmatization can make the plot less cluttered.

6.4.2 Interaction Design Factors

In this section, the primarily interaction design factors that our system encompasses are elaborated. The design is heavily based on user preference and interaction to customize the output. In general, the interaction starts with the alignment overview, the system by default shows the entire collection. The user can interact with the system to modify the alignment overview using the options panel, shown in Figure 6.1 (B). For example, the user can change the order of translations based on a range of sorting presets, filter the alignment overview based on a speaker, or change the presented translation using the time-oriented thumbnail view (Section 6.4.2.3). Then, the user can use the smooth zooming and panning (Section 6.4.2.1) to explore and identify passages of interests. Once the user finds an interesting pattern guided by the predefined customization and the similarity-based color-coded segments, the user can use the detailed view (Section 6.4.1.2) to explore or save the path of alignment by clicking on it, or dragging the path and dropping it on the TLC view (Section 6.4.1.3) to examine the terms contained in the path. When the user clicks on individual segments, the system saves them and allows the user to revisit them and retrieve the alignment overview. Both of the case studies (Section 6.5.2) demonstrate the interactive overall process that facilitates the finding.

6.4.2.1 Smooth Zooming of Translations

In our design, we start with the alignment overview of the whole collection of translations. The user is able to zoom in smoothly and fluidly to explore the dataset and investigate (T1, T2).

When zooming in smoothly, more details fade in gradually, such as the actual text content of each speech to support close reading. See Figure [6.5](#).

With zooming as an option, the user is able to investigate a region of interest in the same view and has the ability to transition between close and distant reading without switching between multiple windows. When zoomed in, the user is able to pan smoothly for comparison of translations (**T4**, **T5**).

During the zooming, we maintain certain levels of detail. The first level illustrates the speech and depicts the length of the segments. Also, the relative thickness of the border highlighting the user-selected segment is increased relatively. We decrease the thickness of the highlighting border as the user zooms in. In the second level, more segment details are revealed, such as local colors and the length of the text. Finally, at the third level, the text is readable and the thickness of the highlighting border is decreased considerably. This functionality is implemented in the alignment overview and it zooms along both x and y axes with respect to the mouse position. Zooming in on only y axis could cause only a single word to appear on each line which is difficult to read. Zooming in on x axis could result in long, difficult to read lines and difficulty comparing alignments.

Multiple works link different abstractions of single text using zoomable layouts. VarifocalReader [\[186\]](#) uses a combination of x and y axes zooming to show the multiple layers of abstraction. However, it becomes challenging when comparing more documents, and the number of abstractions layers needs to be decreased. Gold *et al.* [\[238\]](#) integrates the close and distant reading of a single text using zooming along the y -axis. On the other hand, Asokarajan *et al.* [\[174\]](#) use the zooming along the x axis to examine the view closely and do not provide a level of abstraction based on the zooming.

There are other approaches that utilize interactive lenses [\[239,240\]](#). We believe that magic lens and interactive magnification techniques that allow the user to obtain a close picture of the interested area could be useful. However, they also introduce new drawbacks such as either distortion or discontinuity between the focus and context areas. We discuss these techniques with the domain expert and believe such alternatives could be explored in future through task- and design-driven studies. We did not incorporate them because the user requirements are fulfilled with our current design choices.

We compute the average time (milliseconds) of rendering the scene while performing the zooming. For each number of translations, we record the average time for the process. Our system takes about 5.6 ms to render the scene. To demonstrate the scalability and the linearity

6. TransVis: Integrated Distant and Close Reading of Othello Translations



Figure 6.6: Two snapshots of the same region of the collection. In (a) the segments aligned with the user-selected segment are out of view and not visible. In (b) the segments are horizontally aligned with the user-selected segment.

Between the number of translations and performance time, we modified the dataset to increase the number of texts to 76 translations and 1 base English text. We perform this experiment using a machine with the following specification: Processor: 3.3 GHz Intel Core i5, operating system: Mac OS version 10.14, memory: 8 GB DDR3, and an AMD Radeon graphics card with 2 GB of memory.

6.4.2.2 Filtering, Selection, and Positioning in the Alignment Overview

Filtering and selection aim to reduce the complexity of the scene by abstracting some elements of the translations to help the user focus and find regions of interest. The user can click on an individual speech and see the corresponding translated and aligned speeches interactively. Within each translation, horizontal bars represent a speech or a segment depending on the user preference. The vertical order of bars implicates the position of the corresponding speech or segment in the text. The thickness illustrates the length of the speech or segment calculated in words. However, when adjacent translations are distant from one another vertically, it may become difficult to compare the aligned speeches, specifically in zoomed-in views, as illustrated in Figure 6.6 (a). Thus, the user can choose to interactively align them vertically side-by-side (T5), as shown in Figure 6.6 (b).

We connect the corresponding segments using curved edges as illustrated in Figure 6.6. The user-selected alignment is highlighted to be distinct from other alignments. The user may also smoothly re-arrange the order of the translations manually through a drag-and-drop mouse movement. In our feedback and evaluation sessions with domain expert from the Arts

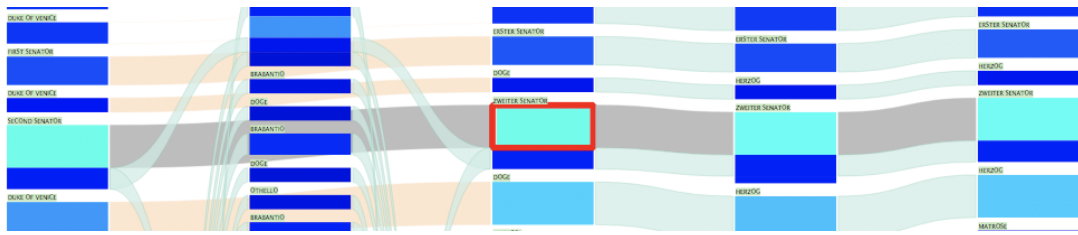


Figure 6.7: The placement of the unstable translation by Bärffuss (2001) [33] results in multiple disconnected edges between non-adjacent translation. The differently colored edges illustrate the alignment between the two non-adjacent translations.

and Humanities, this feature was used extensively.

Example–Juxtapositioning Aligned Speeches: We implement a mechanism to smoothly translate corresponding speeches vertically as seen in Figure 6.6. In Figure 6.6 (a), the user-chosen segment and the aligned segments are not in the alignment overview as a result of their original placement within the translation sequence. In Figure 6.6 (b) the translations slide vertically to line up and with the user-chosen segment. Thus, the user is able to compare and explore the segments in the same close view (T5).

A configuration can occur, if two adjacent translations are not aligned. This might lead to confusion. To address this limitation, we implement a user interaction technique to render such edges that connect non-adjacent translations using a different color and rendering order as shown in Figure 6.7 which illustrates the problem when examining an unstable translation such as Bärffuss (2001) [33]. However, this is still can be challenging particularly when dealing with unstable translations.

6.4.2.3 Filtering and Selection of Speeches and Translations in Options Panel

In the options panel (Figure 6.1 (B)), the user is able to filter and compare the translations in a variety of different ways (T3, T5). We found that this option was always the first user-option chosen by the domain expert (see case studies in Section 6.5.2).

One option is filtering the data based on a custom query. The user is able to search for a given speaker, a specific segment ID or word. Also, the collection can be filtered based on a speaker name if the user is interested in a specific character. The speaker search is non-trivial since the speakers are translated differently from one translation to another, e.g. the speaker “*Duke of Venice*” is translated for example into: “*Der Doge*”, “*Herzog*”, and “*Doge*” and the speaker “*First Senator*” is translated into: “*1 Senator*”, “*Senator*” and more commonly “*Erster*

6. TransVis: Integrated Distant and Close Reading of Othello Translations

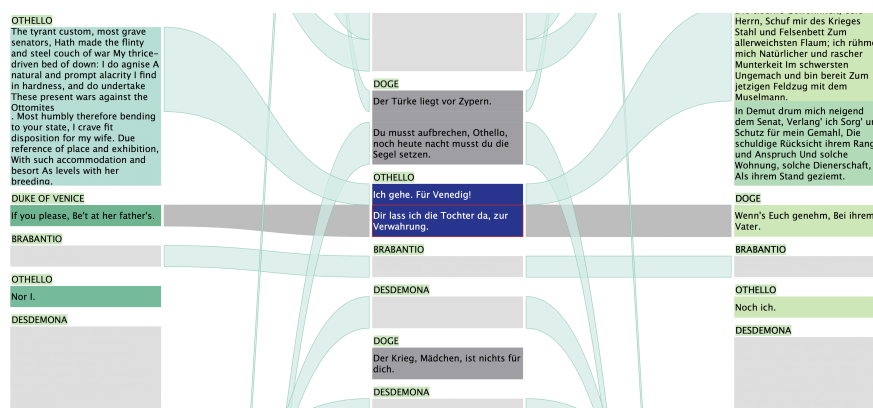


Figure 6.8: A subset of a filtered focus+context rendering of alignment overview. The view is filtered by the speaker “Duke of Venice”. The high number of matches is due to mistaken correspondence during the original segment alignment process which is easily discovered by the visualization result. In this figure, a segment of speech by “Othello” was mistakenly aligned with a segment of speech by “Duke of Venice”.

Senator”. However, to address this issue and to reveal more relevant results across both the base English text and the translations, we populate the speaker list with all of the characters in the original text along with those not aligned with the original text. When the user looks up a speaker, we can eliminate all characters not aligned with it. Thus, we are able to depict all translated speakers in the translations. See Section [6.5.1](#) for the domain expert feedback.

Example–Filtering Based on Speaker: In some segments, we are able to see transformation in speaker names, e.g. the speaker in the speech: “Duke of Venice: Dead?” is transformed differently into “Erster Senator: Tot?” as in Flatter (1952) [\[241\]](#), “Die Senatoren: Tot?” as in Zeynek (1948) [\[242\]](#) and “Senator: Tot?” as in Wachsmann (2005) [\[243\]](#). In some cases, speaker transformation is accepted since the source text might indicate different speakers for the same speech and translators interpret this with a variation.

However, in other cases, the process of the alignment is not accurate and the speaker filter can facilitate the discovery of errors during the original, labor-intensive alignment process. For example, in Figure [6.8](#), the “Duke of Venice” speaker is selected to show only the corresponding segments. However, a great volume of matching translated segments are connected to our surprise (≈ 1500 segments). After investigating, we find that the aligner has mistakenly linked a speech by “Othello” to the speaker “Duke of Venice” which causes this volume of unexpected results.

In all of the above filters, the results are shown using focus+context. The results are visually

highlighted while the context is preserved in greyscale.

Some of the segments in the collection are not spoken in the play but provide directions to the characters. These segments are called stage directions and can be toggled on or off (**T3**, **T5**). Most of the time, researchers are not interested in studying them and removing them can reduce clutter from the scene.

The researcher is able to use the thumbnail view tab to **add or remove translations**. Adding and removing translations is achieved using a smooth drag and drop interaction or simply by selecting and deselecting translations (**T5**). The translations in the options panel as seen in Figure 6.1 (B) are color-coded green or red according to the translation presence or absence in the alignment overview (A) respectively. See case studies Section 6.5.2 for an example of this used in practice.

Additionally, we implement various **translation ordering and sorting** options to aid researchers in finding the best layout that supports comparison of translations (**T5**). The options provide an initial layout that suits the researcher's needs. Some of the options have been suggested by the domain expert, such as sorting the translations chronologically. Other options sort based on the aggregation of the similarity metrics, e.g. average and total of Eddy values. See case studies Section 6.5.2 for an example of this used in practice.

The proposed visual design implements different **color mapping schemes** to assist researchers in investigating the parallel translations. There are different primary color mapping schemes as shown in the accompanying video. In the center, there are three translations attributes that we color code the text based on (segment length, speech length, and Eddy and Viv value). In the bottom section, we provide the user with a range of color mapping scheme presets for the Viv values. The user is able to customize specific translation attributes which are multiple-aligned segments and non-aligned segments. See Section 6.5.1 for a case study demonstrating the utility of these color-mapping schemes. The sequential color schemes are generated using Color Brewer [217]. The rainbow-style color scheme is generated using Telea's algorithm [244].

6.5 Evaluation

We work closely with the domain expert and the project is driven by a real-world historical investigation. With the close observation by the domain expert, we designed the application to satisfy the user requirements in Section 6.3 and facilitate exploration and interaction to enhance the user experience.

In Section 6.5.2, we present two case studies conducted to validate our visual design and the integrated similarity metrics.

6.5.1 Domain Expert Feedback

The Alignment Overview Feedback The alignment overview enables the researcher to capture global patterns and direct attention to a region of interest for a more detailed investigation. When demonstrating the camera-positioning features in the alignment overview, the domain expert states, *“In a birds eye view, the visual design shows how versions differ in length, and the number of alignments, and the Eddy color mapping neatly highlights (a) versions which are generally a high-Eddy and (b) segments and passages in the run of text which are high-Eddy. This is great.”*

The color mapping used to illustrate different translation attributes in combination with sorting options may reveal new insight to the domain expert. As seen in Figure 6.9, the domain expert notices that the variation between translations increases distinctively after the second world war. Also, he stated that, *“modern translators increasingly diverge from the norms of theatrical Shakespeare language established in the 19th century and early 20th versions.”*

Search Feedback The focus+context rendering of alignment overview of search results can help the user to discover patterns or uniqueness in the collection. For example, the domain expert was interested in the word *“Lust”*. The results are appealing and uncover two main stream of paths that use the word. Most of the translations use the word once (13 translations), four translations did not use this term. As shown in Figure 6.10, above the top and below bottom main patterns we can see outlier usage of the word. *“That is very good because I can see straight away some patterns and most of the translators are using the word in the same segments and some translators are using the word unexpectedly,”* the domain expert stated.

Filtering Feedback The domain expert filters out the outlier translations, and stage direction segments then chooses five different editions of Baudissin (1832). The domain expert discovers that Wolf (1926) [245] and Brunner (1947) [246] show high-Eddy values. The domain expert stated, *“These editors ‘intervened’ quite often, altering the text they had received from earlier editions –usually to improve it, i.e. remove bits of poor writing. Both make changes which are not ‘significant’ but just make the text more readable (and actable).”* From this focus and similar context, we can depict translation variation easily. For example, in a speech *Iago* says

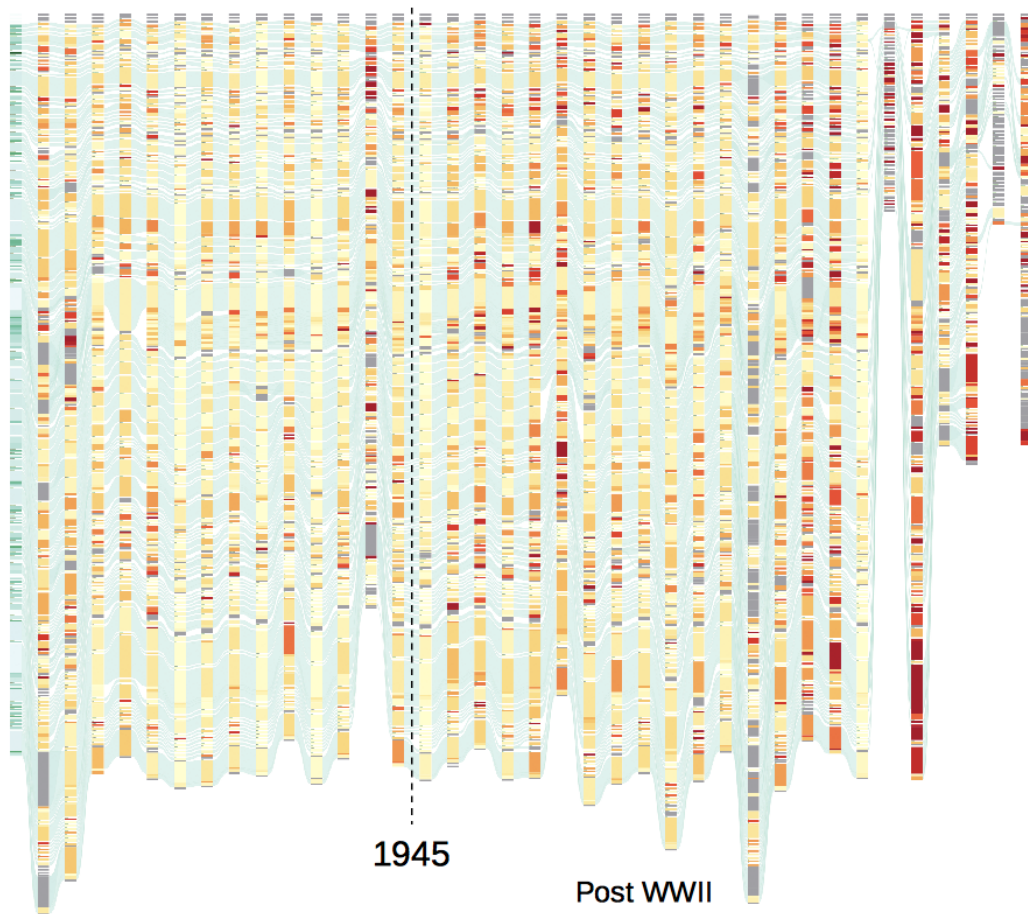


Figure 6.9: The alignment overview of the translation collection reveals an increase of variation between translations particularly after the second world war, with the exception of Engel (1939) [34].

to Roderigo: “*If thou dost, I shall never love thee after. Why, thou silly gentleman!*” All other Baudissin use the word “*Freundschaft*” which means ‘friendship’ and Wolff (1926) uses the word “*Liebe*” which means ‘love’ (Figure 6.11). “*Again, intensifying and bringing back the hint of homo-eroticism in the original, which the classic text censored.*”

Detailed View Feedback The domain expert finds this view useful because it fulfills tasks T4, T5. He experimented with this feature and shortly afterwards he started searching for an alignment path he examined previously. He states: “*How do I find it now?*” The provenance list that stores previously selected paths enables the user to trace back all of the user’s actions whilst exploring the alignment overview. The domain expert finds this helpful to retrace his

previous selections. The technical limit to the number of archived user-interactions is the same as that of a web browser with bookmarks. The domain expert might find it cognitively difficult to remember the location of an alignment in the list when there too many archived user-interactions. This cognitive limit can be addressed in future work by enabling the user to personally rename the archived user-interaction labels. Also, the domain expert suggested a potential feature that enables them to keep notes in each alignment.

Term-Level Comparisons View Feedback The aligned translations of the base English speech starting with “*Thou art sure of me...*” show different segments with noticeable higher Eddy values. We easily plot the path of alignments interactively using a drag-and-drop of any segment from the path onto the TLC view. As seen in Figure 6.3 we discover that there are three translations that stand out from the rest. The three translations have three recognizable words with high values which can be easily observed in the TLC view. These words are translations of the English word “*Moor*” in the context: “*I hate the Moor ...*”. The translations are “*Maure*” in Schwarz (1941) [31], “*Schwarze*” in Günther (1992) [30], and “*Neger*” in Buhss (1996) [29]. Most of the translations translate the word into “*Mohren*”. “*That is a good result,*

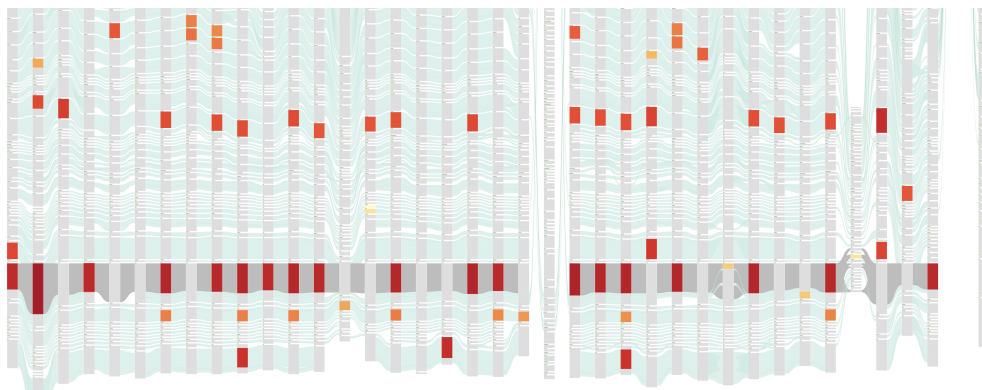


Figure 6.10: Focus+context rendering of alignment overview of the results of the search for the word “*Lust*”.

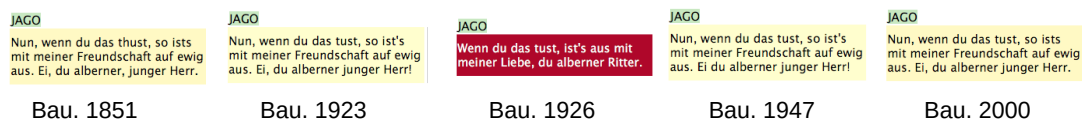


Figure 6.11: An example of a translation variation between five editions of Baudissin (1832). Wolff (1926) stands out to be the most distinctive translation among the editions.

that is interesting," states the domain expert as he looks at the resulting images. He validates the discovery and reasons that as, *"That word is translated in different ways in different times with different political implications."*

6.5.2 Case Studies Using the Design and the Integrated Similarity metrics

We can select a particular set of translations (a sub-corpus) within our corpus on the basis of known features such as type of translation, or date. The Eddy color mapping identifies segments which are of interest because they diverge from others which are more similar. Eddy is calculated on the basis of words, not semantics, so different values do not necessarily predict differences of understanding or interpretation. But Eddy color mapping in this interface encourages an exploratory kind of reading which shifts between scales and between following the course of a single text (vertically) and comparing between texts (horizontally). It also encourages exploring what comparative juxtapositions produce interesting results for a humanistic reader. The case studies are written collaboratively with the domain expert.

Do apparently identical texts differ? Discovering the work of editors/rewriters: As mentioned earlier, our corpus includes different editions of the Baudissin translation, first published in 1832 and often re-published (100s of times so far): printed editions from 1851, 1923, 1926, and 1947, and a digital edition from 2000. Using the interactive drag-and-drop thumbnail view, we can select those five translations. Looking at a distance at this set of five texts, the Eddy value coloring immediately conveys that the 1926 edition is very different from the other four: high Eddy in almost every segment, while the other four show low Eddy values, in nearly all segments as shown in Figure 6.12(a). This reveals that the 1926 so-called ‘edition’ (‘revised’ by Max Wolff) [245] is not really an ‘edition’ of the Baudissin translation at all: it’s almost a completely new translation, created by editing Baudissin. Interactively zooming in for close inspection shows that Wolff consistently modernizes (rewrites Baudissin in 1920s language, including some slang) and frequently intensifies meanings for dramatic effect, including making the play’s homo-erotic subtext more prominent. This reflects a relaxing of sexual inhibitions in the Weimar Republic.

Next, we can interactively de-select Wolff’s 1926 translation from the sub-corpus under investigation, leaving four actual editions of the Baudissin translation. Now, the updated Eddy coloring shows variation among them –as shown in Figure 6.12(b), particularly in Brunner (1947) [246] but also others, in the majority of segments. Some of this variation is due to historical spelling changes (lemmatization aims to filter these out, but can still have trouble

6. *TransVis: Integrated Distant and Close Reading of Othello Translations*

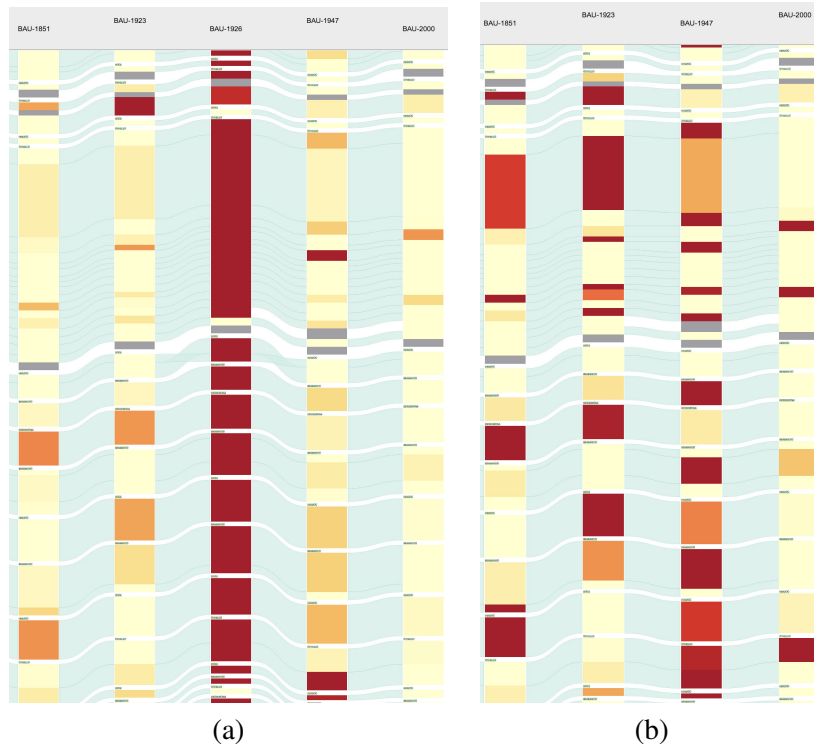


Figure 6.12: (a) Five classic canonical Baudissin translations. We can see the Wolff translation stands out due to the high values of Eddy. (b) After de-selecting the 1926 translation, we can see more variation among the four remaining translations.

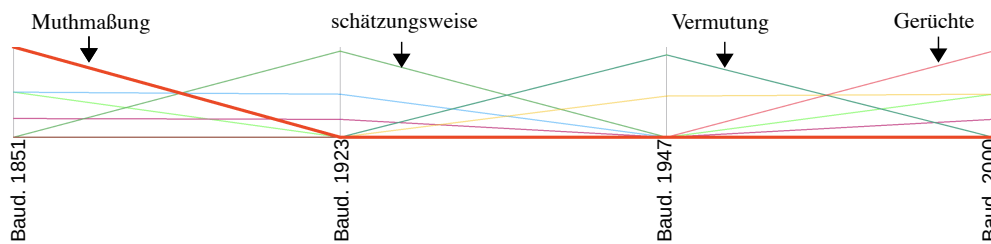


Figure 6.13: The TLC view of the path containing the phrase “*the aim reports*”. The colored lines represent terms and each vertical line represents a translation. The annotated words and arrows illustrate the corresponding terms and lines.

with some antiquated and poetic, unusual forms of spelling). Some changes are more significant. They are caused by editors aiming to ‘improve’ the Baudissin text they have received, making it easier for readers, actors, and audiences. We can explore variation between these translations at the term level using the TLC view, for example Shakespeare’s phrase “*the aim reports*” was translated by Baudissin as: “*Muthmaßung berichtet*” (conjecture reports) (1851, in accord with the 1832 text). This is a very compressed expression (meaning: ‘people making conjectures report’), not easy to follow and also difficult to speak. Some later editors changed this: “*schätzungsweise man berichtet*” (estimating, people report) (1923), “*Vermutung meldet*” (supposition reports) (1947), and “*Gerüchte melden*” (rumours report) (2000). The TLC view (Figure 6.13) illustrates the translation variation between these four editions. The changes in 1923 and 2000 make the text more like conversational German, less ‘literary’. The change in 1947 substitutes a much commoner word. Such small changes are referred to as ‘silent emendations’ in the history of edition-making. Identifying them is a hideously tedious task for traditional scholarship.

The individual distinctiveness of translators: In the same way, the interface makes it possible to explore comparatively sets of independent translations. Our corpus includes a majority of full-length, poetic translations for theatrical performance –26 of these (including the variant Baudissin texts); also four ‘prose’ translations (for reading, not for performance) and eight theatrical adaptations (shortened, and much freer in the ways they translate). From the interactive thumbnail view, we select the 26, and view them at a distance in chronological order. As seen in Figure 6.14, the Eddy value colorings clearly show three periods of generally high Eddy value: the early 19th century (Schiller (1805) [247], Benda (1826) [248], Ortslepp (1839) [249]), the 1950s-60s (Schröder (1962) [250], Rothe (1963) [251]) and the 1990s (Günther (1992) [30], Motschach (1992) [252], Buhss (1996) [29]). The translations created before Baudissin (1832) were experimenting with varied ways of translating Shakespeare’s plays. Baudissin’s version of *Othello* (for the ‘Schlegel-Tieck’ edition of Shakespeare’s plays) soon became canonical - the standard, the one which ‘everybody knows’, even today. Until the 1950s, there were many other German translations of Shakespeare but all were heavily influenced by the ‘Schlegel-Tieck’ style, and Eddy values are low. After the Second World War, new ways of translating Shakespeare began to be tried, as part of a general breaking away from tradition. But this was a complicated process. Innovation and tradition co-existed, tradition becoming stronger again in the 1970s-80s. Then in the 1990s, experimentalism took over.

In the interface, de-selecting the earlier 19th-century translations heightens the visibility

6. *TransVis: Integrated Distant and Close Reading of Othello Translations*

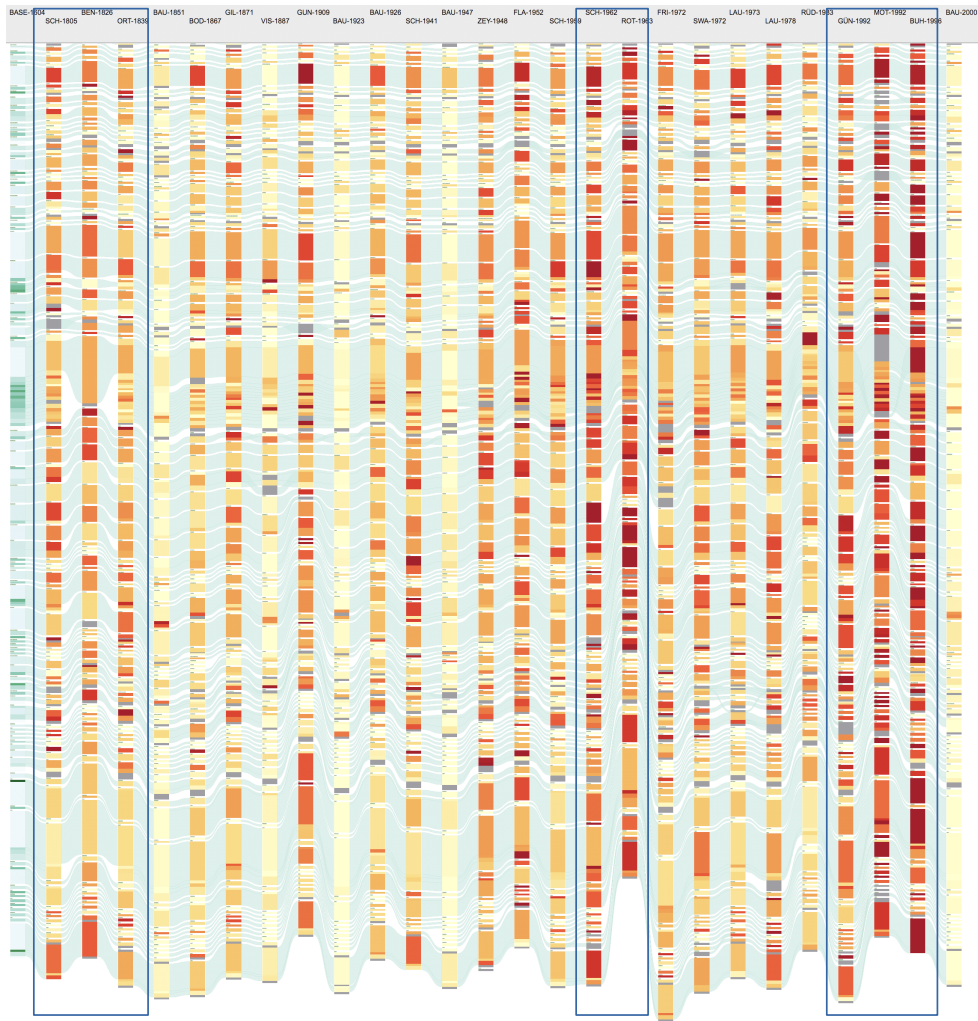


Figure 6.14: A sub-set of the translation selected based on the domain expert knowledge shows three periods of generally high Eddy values. The three periods are highlighted by borders.

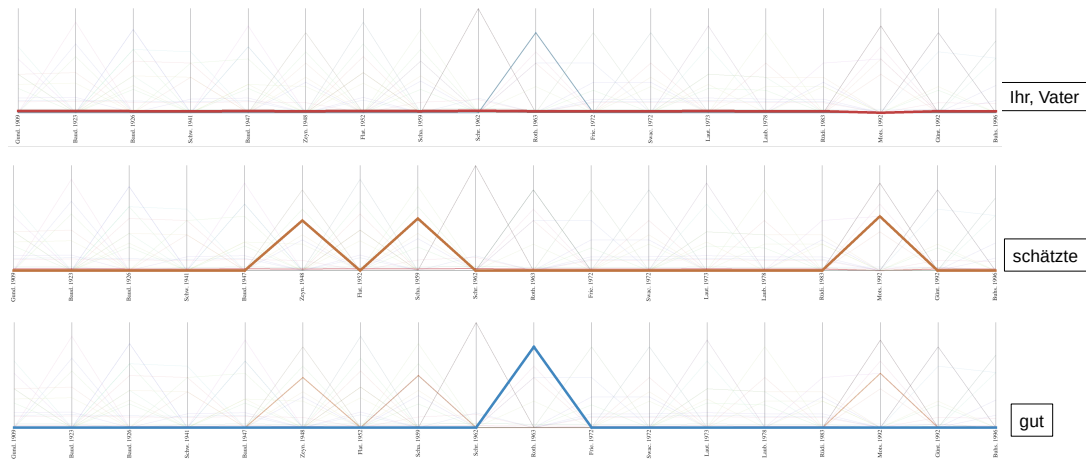


Figure 6.15: Three uses of the TLC view to help the domain expert find variation between different translation. The top shows a common use of the words ‘*Ihr*’ and ‘*Vater*’. The middle and bottom views show that the words: ‘*schätzte*’ and ‘*gut*’ which are used only in specific translations.

of the distinctiveness in the 1950s–1960s and 1990s in terms of Eddy coloring. Inspecting segments with high Eddy in the 1960s and 90s reveals interesting patterns in textual detail. After zooming in for a closer view and with the assistance of TLC view as shown in Figure 6.15, we can see: *Othello*’s line beginning ‘*Her father loved me*’ is translated by all as ‘*Ihr Vater liebte mich*’ (‘Her father loved me’) except: ‘*schätzte mich*’ (thought highly of me) (Zeynek (1948) [242], Schaller (1959) [253], Motschach (1992)), ‘*war mir gut*’ (was fond of me) (Rothe (1963)), ‘*mochte mich*’ (liked me) (Günther 1992, Buhss (1996)), ‘*schien mich zu mögen*’ (seemed to like me) (Leonard (2010) [228])). The periods of more different, distinctive translation are evident here. Love and hate, liking and disliking, and their ambiguities, between men and women and between men, are major themes in *Othello*. Desdemona’s father, Brabantio, hates Othello, his son-in-law. When translators change the kind and intensity of emotions in this way, it matters a great deal. The interface makes it relatively easy to discover patterns in translators’ treatment of emotions and other themes. Exactly why particular translators make particular choices at particular times requires a lot more discussion.

Most of the translators are unknown –their work has never been studied. One is of particular interest because she is the only woman who has translated a number of Shakespeare’s plays into German: Hedwig Schwarz. Her version of *Othello* is from 1941 [31]. By selecting a subset including other full-length performance versions from her lifetime (1898-1985), we can highlight segments where her version is distinctive. There are not many. In several short segments,

6. TransVis: Integrated Distant and Close Reading of *Othello* Translations

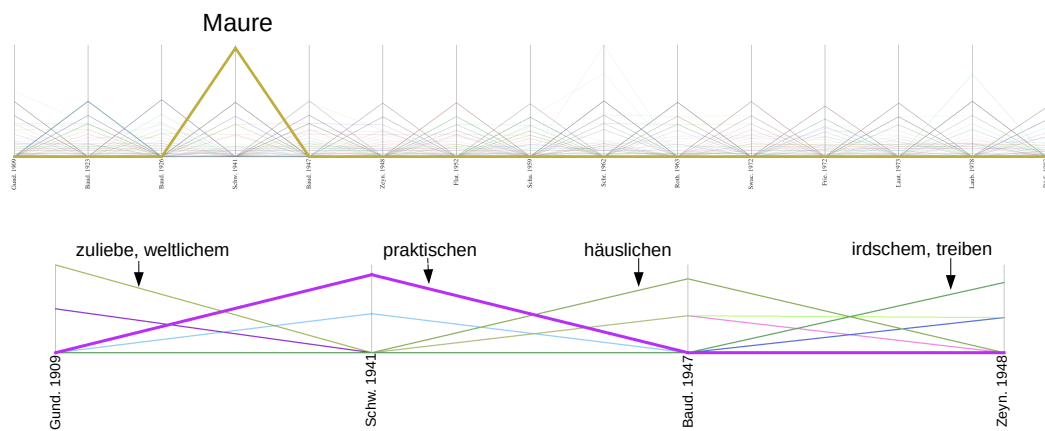


Figure 6.16: Two TLC views that show words used distinctively in the translations. In top the word “*Maure*” was firstly used by Schwarz (1941). In bottom, we can see the distinctive words in the translations of Shakespeare’s odd phrase: “*Of love, of worldly matters and direction*”.

she is unusually concise and informal. A feature which stands out is her unusual use of the term ‘*Maure*’ for ‘*Moor*’. *Othello* is a ‘*Moor*’ – a controversial term, meaning African, variously understood as Black African or North African/Arab/Berber (as in the ‘*Moors*’ of Moslem Spain). German has two traditional words - ‘*Mohr*’ meaning Black African and ‘*Maure*’ meaning North African, with contrary connotations of ‘barbarian’ versus ‘civilized’. All translations before Schwarz used ‘*Mohr*’. She was the first translator to use ‘*Maure*’ for ‘*Moor*’, in the play’s subtitle and in the text. She uses ‘*Maure*’ when the speaker is respectful, e.g. *Othello*’s wife, *Desdemona*. Racist characters (*Brabantio*, *Iago*, *Rodrigo*) use ‘*Mohr*’ but sometimes ‘*Maure*’. *Rothe* (1963) used the same double-word tactic. No other translators have used ‘*Maure*’ since. Figure 6.16 (top) shows the TLC view with the word ‘*Maure*’ highlighted. Some more recent translators use provocatively offensive terms, or neutral terms intended to minimize the racism theme: different responses to ongoing developments in the politicisation of race language and awareness of its emotional force: one of the themes of *Othello*. Schwarz was a pioneer in using the variety of available German terms in order to dramatize race attitudes within in the play. Her use of ‘*Maure*’ is clearly intended to dignify *Othello* as a black man, which is perhaps surprising given when and where she was writing. Interactively reading up, down and diagonally in the corpus enables us to see this.

A further reduced sub-set, using the thumbnail view, can be investigated to compare Schwarz with *Baudissin* (a translation she knew), *Gundolf* (1909) [254], one which she may

have known, and Zeynek (1947), who also translated during the Nazi dictatorship. Her version is shown to be generally lowest on distinctiveness. She is generally conservative. But Eddy is high for her segment including Shakespeare's odd phrase '*Of love, of worldly matters and direction*'. We can justify the distinctiveness pattern using the TLC view as shown in Figure 6.16(bottom). Baudissin had: '*Der Lieb und unsrem häuslichen Geschäfte*' (love and our household business); Gundolf: '*Zuliebe weltlichem Geschäft*' (for the sake of worldly business) (a typical poetically condensed translation by him); Zeynek: '*zu Liebe, irdschem Tun und Treiben*' (for love, earthly activities); Schwarz: '*für die Liebe, für die praktischen Geschäfte*' (for love, for practical business). The example illustrates her rather practical, easily comprehensible, modern language style. Zeynek, on the other hand, uses antiquated and ornate language, and so scores generally very high Eddy values.

6.6 Chapter Summary

In this chapter, we present a unique, integrated visual design to support distant and close reading of the collection of parallel translations of *Othello*. The visual design aims to present a smooth interactive experience for digital humanities scholars. We identify five main tasks that our proposed application addresses. The application consists of four components. The first one is the context view (alignment overview) of the collection which leverages a range of exploration and interaction techniques. It facilitates smooth zooming which can integrate distant and close reading within the same window. The second component is the options panel which enables the user to customize the alignment overview. The third is the detailed view which interacts with the main window and displays a close reading of the alignments of the selected segment. Also, it saves the history of user-selected segments and reveals previous alignments. Finally, we present the TLC view, which is a novel and interactive technique to examine the word level variation within the translations. Our proposed application is driven and evaluated by experts from the Arts and Humanities. We provide the domain feedback on the application features. Future work and limitation are discussed in Chapter 8.

Chapter 7

Collaborating with Digital Humanities: A Methodology

“You are more likely to find big breakthroughs if you work on a real problem.”

— Ben Shneiderman (1947-)

Contents

7.1 Introduction and Motivation	132
7.2 Background	132
7.3 Collaborative Workflow	133
7.4 Collaboration Outcome and Reflection	139
7.5 Chapter Summary	141

In previous chapters, we presented software designed in close collaboration with digital humanities scholars. In this chapter, we aim to address the fifth research objective [Ob5]. We reflect and summarize the collaborative research and propose a methodological workflow to guide such interdisciplinary projects. Through discussion of this workflow, we report challenges that arise from such projects and how the proposed workflow addresses them. The workflow contains three spaces that illustrate the outputs, three channels that help deliver them, and three quality criteria that need to be fulfilled in order to obtain useful outcomes. Finally, we discuss our collaboration’s outcome utilising guidance provided by Sedlmair et al [11].

7.1 Introduction and Motivation

Digital humanities increasingly adapt visualization approaches to enrich their research. They find that visualizations create new modes of knowledge and facilitate more effective discovery of new observations [26, 67, 68]. Hinrichs and Forlini [68] claim that visualization should be considered not just a means to an end but as a research process in its own right, which has led to the development of multiple interdisciplinary collaborations between the digital humanities and visualization communities. These collaborations have also been studied and discussed in both communities in order to identify means to enhance the collaborations and discuss the challenges encountered [26, 255, 256].

We believe that collaboration should follow a conceptual workflow that considers all aspects of collaboration if possible. Without this, collaboration may develop in undesirable directions due to the different perspectives each stakeholder brings to the project.

The rest of this chapter is organized as follows: In Section 7.2 we present previous research related to digital humanities collaborative research. Section 7.3 relates our proposed conceptual workflow to our collaborative experience. Section 7.4 reports challenges that encountered our collaborative work and design guidelines that are derived based on the design of our chapter 6.

7.2 Background

Collaboration between the visualization team and digital humanities for interdisciplinary visualization projects has been the subject of significant discussion. Recent developments in interdisciplinary research highlight challenges in digital humanities projects and encourage research to propose a collaborative framework to address these challenges [69, 257, 258]. Munzner [259] proposes a general nested model that guides the process of design and evaluation of visualization projects, while Kath et al. [260] propose a methodological framework supporting knowledge generation of collaborative projects using visualizations. Simon et al. [261] suggest the liaison role shares knowledge and language with both domains to foster collaborative communication. El-Assady et al. [255] present a conceptual workflow of the problem-solving process and collaboration in digital humanities projects with visual text analytics. Jänicke et al. [26] discuss collaboration themes, including the initial start of projects, development iterations, and evaluation methods. Roberts et al. [262] discuss a similar process on the collaboration between academia and industry in visualisation projects, discussing the nature of such projects and how knowledge transfers between the two parties throughout an

interview study. Most recently, Schetinger et al. [263] introduce a re-purposed framework of the Data-Users-Tasks triangle [35] to overcome limitations in the context of digital humanities.

In this chapter, we provide a methodological workflow based on our previous collaboration with digital humanities. The approach combines the three most important aspects: domain, tasks, and design spaces. It also integrates quality criteria to ensure useful outcomes.

The rest of this chapter presents our methodological collaborative workflow based on the accumulative experience of collaboration.

7.3 Collaborative Workflow

The proposed workflow is a conceptual workflow (Figure 7.1) to inform the collaboration and design of interdisciplinary visualization projects. It features three spaces, three channels, and three criteria. The following section discusses the workflow components and how they complement one another.

7.3.1 Three Spaces and Three Channels

This section consists of a discussion of our workflow. The word “*channel*” is used to illustrate the connective phases between spaces as they usually involve communication between the two users in the workflow.

Problem Space: The problem space is the starting point of the workflow. It essentially resembles the domain users, the data, and more likely a set of challenges. For example, in the TransVis project [2], a collection of German translations of Shakespeare’s *Othello* was curated by the domain expert in order to be analyzed and visualized. Exploring and examining the collection without computational and visual aids is a laborious and challenging process for digital humanities scholars. The domain users are usually interested in studying how existing approaches can solve their problems, and they generate hypotheses to be confirmed and evaluated based on their data. In this space, the domain problems are clearly identified. Each problem statement needs to be unambiguous, focused, concise, complex, and arguable [255].

Communication Channel: This channel plays a vital role when collaborating on interdisciplinary projects. In our collaboration, we think of this communication as an educational experience for both domain scholars and visualization teams. The domain scholars strive to

7. Collaborating with Digital Humanities: A Methodology

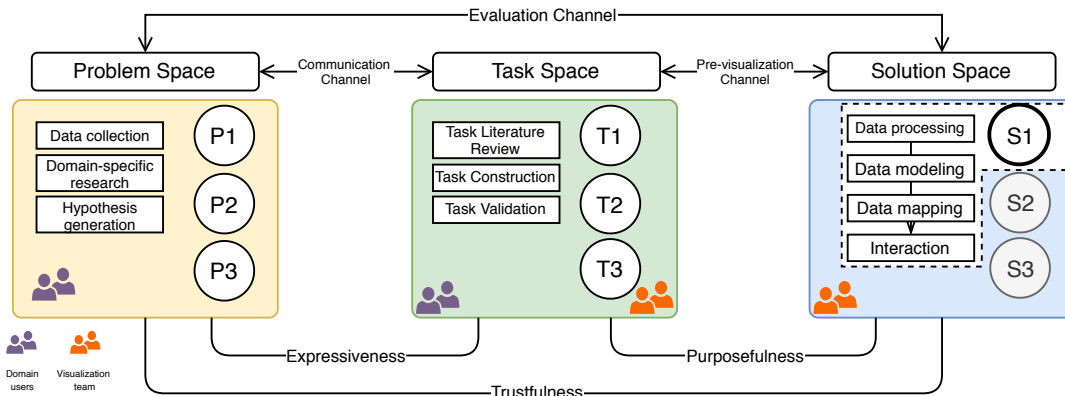


Figure 7.1: Our proposed methodological and interdisciplinary workflow. The workflow consists of three main components: the domain, task, and solution spaces. The tasks are informed by a communication channel between the two users' groups. A pre-visualization channel is attempted between the task and solution spaces to prepare for implementation. In the solution space, one or multiple solutions are implemented to address the predefined tasks. The terms expressiveness, purposefulness, and trustfulness indicate the quality criteria that need to be fulfilled to obtain useful outcomes (Section [7.3.2](#)).

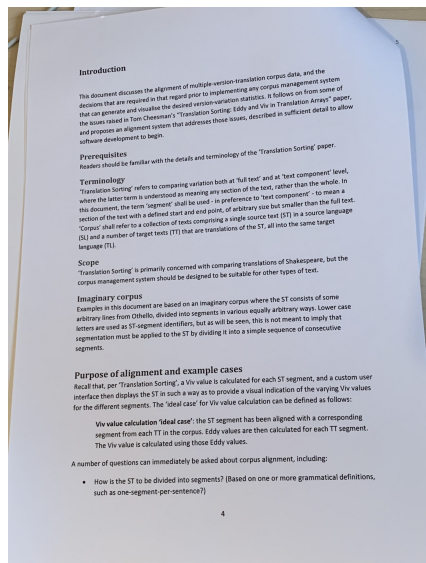
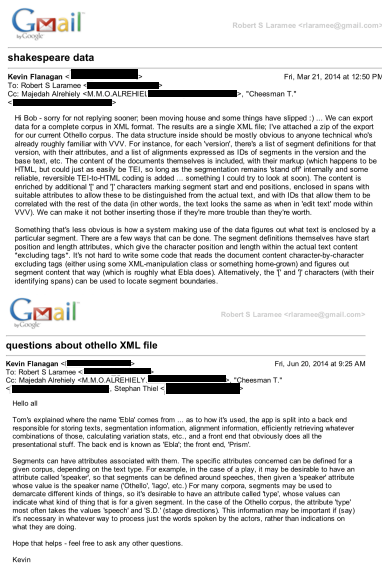


Figure 7.2: On the left, email that was exchanged in the early stages of the collaboration to enable sufficient understanding of domain-specific terminology. On the right, a screenshot of a formal document that explains the dataset components and terminology.

understand computational and visual tools as much as possible. This understanding and awareness increase the chances of designing helpful solutions. This means that the visualization

team explains the fundamentals and does not assume that they are self-explanatory. The visualization team also strives to understand the domain data and problems, and is encouraged to participate in domain readings and discussions which can help discover relevant mutual problems [67]. The development of such common knowledge can be complex, and we suggest constructive regular meetings at the early stages of collaboration to bridge the differences between the two domains. Simon et al. [261] propose a liaison role in the workflow who shares sufficient knowledge in both disciplines to foster more effective interdisciplinary communication and contributes to the project by capturing the problem complexity or mental model. We also suggest documenting a glossary of terminology that define the key terms in the domain area. In the early stages of the collaboration between the two teams, various email was exchanged and sessions were held to create sufficient understanding of the dataset element and its associated terminology (Figure 7.2).

In this communication phase, flexibility is an essential skill as the discussions strive for balance between the two disciplines. If the visualization team focuses more on the implementation and computational side, it might result in failure to deliver useful solutions. Additionally, what each discipline considers a contribution may vary and this could take the project in an undesired direction if the initial communications and discussions are not balanced [263].

Task Space: In this space, the tasks are formulated based on the research problems (gaps) and interdisciplinary discussions, and are expressed differently between domains. One problem could result in one or more tasks to be solved. The domain scholar might have broad, high-level tasks, such as close or distant reading, while the visualization team is responsible for transferring these tasks into more technical, well-expressed tasks. The tasks are complete, discriminative, objective, and measurable [264]. Although this is not always achievable, it is nevertheless attempted. In a previous collaboration [2], we adapted the detailed Brehmer and Munzner [234] typology of visualization tasks, which can be communicated to the domain scholar in order to abstract user tasks.

Pre-visualization Channel: This channel is usually where the visualization team parts ways with the domain scholars. The main goal of this channel is to study user tasks and data and begin implementing visual solutions. Here, the visualization team surveys the design space of existing approaches in order to explore potential design solutions and carefully study their advantages and limitations [264]. It is also beneficial to study the domain specific tools because the use of visualization is becoming an essential element of research [68]. The main properties

	TransVis [2]	AlignVis [1]	VNLP [10]
Data processing	data cleaning, integration, tokenization, normalization, feature extractions		
Data modeling	Eddy and Viv analysis	Similarity computation	Embedding analysis, similarity computation
Data mapping	Segments colors	Segments and edges colors	Histogram, bar charts, etc
Interaction	multiple sorting options, Filtering and selection	Confidence threshold, Filtering and selection	Overview similarity results, Customizable pipeline items

Table 7.1: Example representatives of the results of the implementation stages in the solution space that correspond to our contributions, TransVis [2], AlignVis [1], and VNLP [10].

of the activities in this channel are that they involve iterative sketching and trials, and it is crucial to communicate the results to the domain scholar and validate appropriateness against the tasks specified.

Solution Space: Implementing a visual solution starts with data transformation. Often, the data that comes from the domain suffers from a number of problems and may come from a variety of sources with different formats or conventions. Therefore, the data must be preprocessed in order to be cleaned, integrated, and prepared for the next stage. In data modeling, the data is analyzed and interesting meta-data derived, such as Eddy and Viv [2], alignment detection [1], and embeddings generation (VNLP). In data mapping, the abstracted data is mapped to visual encodings. Lastly, user interaction is implemented to aid exploratory analysis. Table 7.1 shows example representatives of the results of the implementation stages that correspond to our contributions, TransVis [2], AlignVis [1], and VNLP [10].

We recommend implementing prototypes iteratively with a subset of the data and presenting the results to the domain scholars. Such frequent presentations and discussions help satisfy the user tasks, obtain intuitive results, and increase the domain scholar’s engagement [265].

Evaluation Channel: Evaluating the efficacy and usability of the visual solution is an essential goal of any interdisciplinary project. However, many visualization approaches lack an in-depth, effective quantitative or qualitative evaluation [3]. Furthermore, humanities scholars tend to doubt and question computational, qualitative evaluation. A lack of ground truth is one of the most common challenges in digital humanities [69]. Jänicke et al. [26] report that there are more visual approaches for text analysis tasks published in digital humanities than in the visualization communities due to the usual demand of quantitative evaluations which are challenging to incorporate as a result of the limited number of collaborators from the humanities. Munzner [259] provides guidance on evaluation methods for different design choices. Lam et

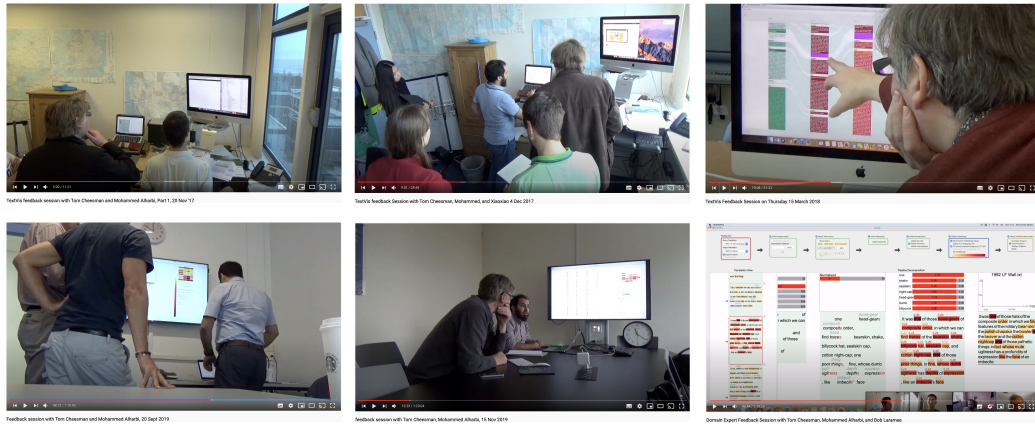


Figure 7.3: Samples of different recordings of our domain expert feedback sessions.

al. [266] provide a scenario-based method to study evaluation for information visualization. They introduce seven scenarios derived through an extensive literature review of over 800 visualization publications. There has also been work on evaluating visualization which guides users on how to carry out an evaluation for information visualization [267-270].

In our collaboration with the domain scholar, we evaluate our project usability obtaining domain expert feedback and conducting use cases. The domain expert feedback is based on regular sessions to demonstrate the design features. All of the sessions are video-recorded for post-analysis and archiving. Figure 7.3 shows a selection of feedback session recordings of our collaboration with the domain expert. Table 1.1 shows a list of these sessions along with a link to the video recording. Semi-structured interview questions are planned and guided by Hogan et al [70]. The early sessions usually consist of mock-ups, sketches, or software demonstrations to guide the development of features, and gradually become active hands-on use of the software by the domain expert. During the sessions, the software evolves due to feature demands. During the face-to-face feedback sessions, patterns can be observed, such as the discovery of software bugs and data-level errors.

7.3.2 Quality Criteria

The design triangle (data–users–tasks)(Figure 7.4) methodological approach to inform the design of interactive visualizations suggests three quality criteria that need to be fulfilled in order to obtain useful outcomes [35]. Expressiveness refers to the requirement of conveying the information contained in the data, effectiveness concerns the degree to which the visualization addresses the cognitive capabilities of the human visual system and the context of the user, and

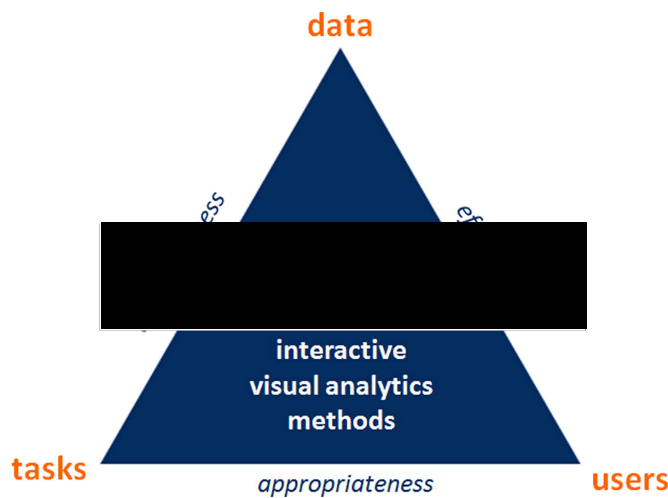


Figure 7.4: The design triangle by Miksch and Aigner [35]. They include factors to be considered during the design and implementation of interactive visualizations. Image courtesy of Miksch and Aigner [35].

appropriateness quantifies the cost-value ratio of the benefit of the visualization process with respect to achieving the intended task. Schetinger et al. [263] repurpose the design triangle and propose three quality criteria that take into consideration the context of digital humanities. Trustfulness reflects the degree to which it can provide guarantees of faithfulness within the epistemological framework of its domain, purposefulness is similar to appropriateness and represents the relation between users and tasks, and meaningfulness expresses the potential value of the custom-made visualization software in terms of generating new insight from the data.

Our workflow consists of domain, task and solution spaces. We adopt similar quality criteria that need to be satisfied in order to obtain the most useful results: expressiveness, purposefulness, and trustfulness (Figure 7.1).

Expressiveness refers to the relation between the problem space and task space. It has general and specific aspects. The general aspect is that interdisciplinary exchange and communication can be challenging [255, 261], so a glossary of terminology can be adopted and all researchers involved are clearly established. In the specific aspect, the tasks must be well-expressed. Different well-established typologies of task abstractions [177, 271] can be utilized to establish a well-defined and expressed task and requirement analysis.

Purposefulness refers to evaluating the visualization against the given tasks. If the requirement and task analysis are optimally defined, this criterion can be quantified. It also important also to consider alternative solutions and how they would achieve the same tasks if they could.

Trustfulness refers to the relation between the solution and the expert user and to what extent they trust the result. Visualizations designed for digital humanities tend to exhibit black-box behavior (not transparent). Rieder and Röhle [272] define transparency as “our ability to understand the method, to see how it works, which assumptions it is built on, to reproduce it, and to criticize it”. Based on this, overcoming the lack of transparency is a challenge. The results of modeling and machine learning algorithms are often difficult to interpret and back-trace. Additionally, visualization tends to reduce informational dimensions to produce a focus that shows certain perspectives or interpretations of the data [69]. Based on our collaboration, the domain users do not appreciate this and struggle to trust such results until they understand how they are derived, which is in most cases very difficult. In our collaboration, we keep a close connection with the domain user in the early stages and validate the visual approach with a subset of the data that they know. The user evaluates the result based on the input data. If this visual approach is deemed faithful to the data and domain knowledge, we test the tool with a larger subset of the data.

7.4 Collaboration Outcome and Reflection

In this section, we discuss our collaboration outcome utilising guidance provided by Sedlmair et al [11]. Our collaborative tools have not been fully integrated with the domain expert’s previously used framework (VVV) due to limited resources. This challenge is reported by Sedlmair et al. [11] as pitfall (PF-5: Insufficient time available from potential collaborators). As the domain expert is familiar with visualization projects related to the domain, the projects encountered another pitfall (PF-17: Experts focusing on visualization design vs. domain problem) where the domain expert sometimes focuses on the visualization design problems rather than the domain problem. The projects also encountered another pitfall (PF-30: Too much domain background in paper) in the earlier stage where we presented unbalanced background towards the domain side. Although we think the domain expert feedback and the case studies are balanced, we observe more emphasis on positive feedback which could indicate to another pitfall (PF-26: Liking necessary but not sufficient for validation). Table 7.2 compares and reports our experiences and reflections on the pitfalls reported by Sedlmair et al. Another possible pitfall which is related to the presentation of the work is that attempting to publish the work too soon before serious and thoughtful refinement (PF-32: Premature end: win race vs. practice music for debut).

We also encountered other challenges which are not always a pitfall and might relate to

some of the pitfalls reported in Table 7.2. For example, feature creep [273] (PF-33) is a challenge we encountered, where the domain expert requests features beyond the previously defined requirements and are difficult to incorporate in the current design because it might need another design approach. Another pitfall we encountered (PF-34) is the domain expert expectations of the development cost when implementing new features which may be too high. We believe that this applies to many cross-disciplinary projects where the domain experts lack the knowledge of the development cost.

In order to communicate some of the lessons learned throughout TransVis project lifespan (Chapter 6), we derive some general visualization guidelines to facilitate transferring the experience to the general visualization audience:

- Implement support for customization of the texts. Giving the user control of adding or removing translation, and to sort based on a variety of options is always appreciated.
- Provide a mechanism to save the user actions. We think this supports the user better while exploring the application.
- Users appreciate smooth zooming due to the context-and-detail provided in the same window.
- Close reading is always preferable to be available [26, 168]. Also providing users with complete access to the text is beneficial to enable them to analyse and explore text beyond the visual tool.
- Implement a keyword or sentence search functionality. We found this feature useful during the feedback session.
- Support close and distant reading for a large number of texts. We encourage further research that supports distant and close reading for an even larger number of texts.
- Establish good communication with the domain expert(s) in order to guide the project and to provide useful feedback.

We also offer some caution that working on text analysis for multiple languages is very challenging because different languages have different analysis tools associated with them requiring a variety of specialized language-specific knowledge.

Pitfall #	Pitfall	How?
PF-5	Insufficient time available from potential collaborators	No time/support for full deployment
PF-17	Experts focusing on visualization design vs. domain problem	Domain expert focuses in design issues
PF-25	Usage scenario not case study: non-real task/data/user	Some reported discoveries made by developers
PF-26	Liking necessary but not sufficient for validation	Domain experts were linked closely during design, however, some criticisms were still reported
PF-30	Too much domain background in paper	At earlier stage, the background was dominant
PF-32	Premature end: win race vs. practice music for debut	Attempts to publish early submissions too soon before serious and thoughtful refinement
PF-33	Feature creep	The domain expert requests features beyond the defined requirements
PF-34	Domain expert high/low expectation	The domain expert might lack the knowledge of the development cost

Table 7.2: A table of the encountered pitfalls identified by Sedlmair et al. [11] and their relevance to this project.

7.5 Chapter Summary

This chapter proposes a methodological workflow for collaborative research with digital humanities, introducing three spaces, three channels, and three criteria to guide the collaboration in order to produce visualization solutions. The spaces characterize the domain, task, and solution aspects of the project. The channels illustrate the three communicative means between spaces: the communication channel between the problem space and the task space, the pre-visualization channel between the task space and the solution space, and the evaluation space between the solution space and the problem space. The three criteria (expressiveness, purposefulness and trustfulness) are essential to obtain useful outcomes between each space. Future work is discussed in Chapter 8.

Chapter 8

Conclusions

“Try to name a true success story that involves someone giving up before the job was done.”

–Wess Roberts (1946–)

Contents

8.1 Outcomes	143
8.2 Future Work	145

The objective of this thesis is to propose visual solutions that address challenges of visualizing and analyzing parallel translations and texts. This chapter reviews how each chapter contributes a solution to a domain challenge and discusses future directions based on the discussions and collaboration reported.

8.1 Outcomes

This thesis is based on real-world data and challenges. In the following, we discuss how each chapter contributes to research challenges, followed by a more general conclusion.

Chapter 2 addressed the first research objective [Ob1: A review of surveys of text visualization]. An extended meta-survey of literature reviews in the field of text visualization was presented, classifying and discussing the features of the existing surveys based on five themes: (1) document-centered, (2) user task analysis, (3) cross-disciplinary, (4) multi-faceted, and (5) satellite-themed. The chapter also provided survey recommendations for further research in

the field of text visualization based on each aspect discussed, gave details about challenges in the field and examined potential future trends.

Chapter 4 addressed the second research objective [Ob2: A transparent and informative visual design that justifies Natural Language Processing (NLP) results]. A visual framework (VNLP) was presented which enables users to observe and participate in the NLP pipeline processes, explicitly interact with the parameters of each step, and observe the effects on the visible VNLP result. The aim was to address the challenge of making the algorithmic results of NLP preprocessing pipeline more transparent and informative. Our visual framework was combined with an application to text similarity and demonstrate usefulness of the VNLP. This work is a result of close collaboration with an expert in modern languages and translation studies and guided by domain expert feedback.

Chapter 5 turned to the third research objective [Ob3: A semi-automatic visual alignment approach]. This chapter presented a visual tool (AlignVis) that combines interactive visualization with domain knowledge intervention to facilitate the alignment of parallel translations. AlignVis was designed in close collaboration with the domain expert to implement five domain requirements and guided by domain expert feedback. It is comparable to widely-used standard alignment tools as well as computational and visual alignment tools.

Chapter 6 examined the fourth research objective [Ob4: Integrated close and distant reading of the alignments]. Here, a unique, integrated visual design to support distant and close reading of the collection of parallel translations of *Othello* was presented. The visual design aims to present a smooth interactive experience for digital humanities scholars. Five main tasks that our application addresses were identified, and the application consists of four components. The first is the context view (alignment overview) of the collection, which leverages a range of exploration and interaction techniques and facilitates smooth zooming which can integrate distant and close reading within the same window. The second component is the options panel, which enables the user to customize the alignment overview. The third is the detailed view, which interacts with the main window and displays a close reading of the alignments of the selected segment as well as saving the history of user-selected segments and revealing previous alignments. The final component is the TLC view, which is a novel and interactive technique to examine word level variation within the translations. Our proposed application is driven and evaluated by experts from the Arts and Humanities and the domain feedback on the application features was reported.

Chapter 7 addressed the fifth research objective [Ob5: A methodological and collaborative

workflow]. Our experience during collaborative work was reported and a methodological and collaborative workflow proposed to ensure the fulfillment of the predefined domain problems.

8.2 Future Work

This section summarizes the possible future directions highlighted by this work. Chapter [2](#) summarizes the global challenges reported in our collection (Table [2.3](#)) some of which we faced in our work, such as scalability, effective evaluation methods, and engaging in a multidisciplinary framework. Future research challenges and work are as follows.

Literature-based challenge in the field As the information visualization landscape evolves, the chances of working on a problem that has already been solved increase. Also identifying new domain challenges and unsolved problems is becoming more difficult. There are some areas that can benefit from a thorough literature review as the number of techniques and resources is increasing. For instance, a survey of the NLP tools that can facilitate humanities tasks, or a study of how existing visualization techniques incorporate comparison techniques in order to help humanities scholars.

Scalability Producing an interactive visualization can be challenging even when applying our techniques to a limited number of parallel texts. As textual data is usually associated with high dimensionality and scalability, more in-depth research is suggested to address this challenge, especially when dealing with large parallel collections.

Generalizability Our designs were applied to specific text collections and languages, so generalizing them for different corpora and languages is a challenge because each corpus has different encoding schemes and annotations. Our tools were designed to incorporate the annotation of the VVV project website [\[55\]](#) Research into more generalizable designs is therefore encouraged.

Specialized data from printed sources As discussed in Chapter [1](#) collection and curation is a challenging task for the humanities due to the nature of their original data. Years can be

spent collecting, scanning, cleaning, and curating in order to generate a corpus, and much of the work is conducted and checked manually. Further research into methods that could address this challenge could prove fruitful.

Adopting advanced linguistics techniques In each chapter, existing techniques were applied and the space is open for more research to experiment with and apply different techniques. In Chapter 4, we applied our design to the basic NLP functions and encouraged extending it to include more NLP functions such as part-of-speech tags (POS), named-entity recognition (NER), and syntax trees. Another promising direction is contextual embedding in the application of document similarity and translation variations.

Usability studies The most common evaluation methods used in interdisciplinary, collaborative projects are usage scenarios and case studies. Although formal usability studies are challenging due to the careful design of specific targeted tasks, they could be a source of fruitful future work.

Other future work The following is a review of the specific future possibilities reported in Chapter 4, Chapter 5, and Chapter 6.

Chapter 4 opens up various future directions. It can serve as a starting point for this broad subject and we encourage applying it to more techniques and NLP functions such as named-entity recognition and syntax trees. While this design is limited to one application, adding different applications to increase the usability of the framework such as visible sentiment analysis or visible text classification would be of great value.

In Chapter 5, it was reported that the domain expert proposes integrating the similarity measurements to vote for the correct alignment and highlighting alignment patterns such as one-to-many and one-to-nil. The ability to use a small screen, visualize multiple documents and provide close and distant reading are challenging tasks that could be addressed in future research.

Chapter 6 suggests, as a future enhancement, the incorporation of semantic clustering and structural similarity between translations. The rendering could be updated to reflect this when

zooming in or out of the scene. This chapter opens up many avenues for future work in terms of the scale of the corpus, the use of more sophisticated linguistics or evaluation methods, and more languages. Future possibilities include the application of translation training. This also includes the development of novel techniques for close reading, as most existing techniques adapt simple visualization techniques such as color or font size. Another future direction is the adaptation of techniques that detect and visualize transpositions in parallel texts. These transpositions can appear in multiple hierarchy levels such as word, paragraph, etc.

For the context of Chapter 8, we would like to apply our methodological workflow to other real-world interdisciplinary research projects such as iTeal [116] and ViTA [146].

Bibliography

- [1] M. Alharbi, T. Cheesman, and R. Laramée, “AlignVis: Semi-automatic Alignment and Visualization of Parallel Translations,” in *24th International Conference on Information Visualisation (IV)*, 2020, pp. 98–108.
- [2] M. Alharbi, T. Cheesman, and R. S. Laramée, “TransVis: Integrated Distant and Close Reading of Othello Translations,” *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, jul 2020.
- [3] M. Alharbi and R. S. Laramée, “SoS TextVis: An Extended Survey of Surveys on Text Visualization,” *Computers*, vol. 8, no. 1, pp. 17–35, 2019.
- [4] M. Alharbi and R. Laramée, “SoS TextVis: A survey of surveys on text visualization,” in *Computer Graphics and Visual Computing (CGVC)*, G. K. L. Tam and F. Vidal, Eds. The Eurographics Association, 2018.
- [5] N. Alharbi, M. Alharbi, X. Martinez, M. Krone, A. S. Rose, M. Baaden, R. S. Laramée, and M. Chavent, “Molecular Visualization of Computational Biology Data: A Survey of Surveys,” in *EuroVis 2017 - Short Papers*, B. Kozlikova, T. Schreck, and T. Wischgoll, Eds. The Eurographics Association, 2017.
- [6] “IEEE Xplore,” <http://ieeexplore.ieee.org/Xplore/home.jsp>, 2016, accessed: 2017-2-26.
- [7] “ACM digital library,” <http://dl.acm.org/>, 2016, accessed: 2017-5-26.
- [8] “Google scholar,” <https://scholar.google.co.uk/>, 2016, accessed: 2017-1-20.
- [9] D. A. Keim and D. Oelke, “Literature fingerprinting: A new method for visual literary analysis,” in *2007 IEEE Symposium on Visual Analytics Science and Technology*, 2007, pp. 115–122.
- [10] M. Alharbi, T. Cheesman, and R. Laramée, “VNLP: Visible natural language processing,” 2020, submitted.
- [11] M. Sedlmair, M. Meyer, and T. Munzner, “Design study methodology: Reflections from the trenches and the stacks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2431–2440, 2012.

- [12] S. Chatterjee and A. Firat, “Generating data with identical statistics but dissimilar graphics: A follow up to the anscombe dataset,” *The American Statistician*, vol. 61, no. 3, pp. 248–254, 2007. [Online]. Available: <http://www.jstor.org/stable/27643902>
- [13] E. R. Tufte and P. R. Graves-Morris, *The visual display of quantitative information*. Graphics press Cheshire, CT, 1983, vol. 2, no. 9.
- [14] W. Playfair, *Playfair’s commercial and political atlas and statistical breviary*. Cambridge University Press, 2005.
- [15] J. Snow, *On the mode of communication of cholera*. John Churchill, 1855.
- [16] R. C. Roberts, R. S. Laramee, G. A. Smith, P. Brookes, and T. D’Cruze, “Smart brushing for parallel coordinates,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 3, pp. 1575–1590, 2019.
- [17] P. Laubheimer, “Treemaps: Data visualization of complex hierarchies,” <https://www.nngroup.com/articles/treemaps/>, 2019, accessed: 2020-12-15.
- [18] G. Michailidis, *Data Visualization Through Their Graph Representations*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 103–120. [Online]. Available: https://doi.org/10.1007/978-3-540-33037-0_5
- [19] W. Cui, “Visual analytics: A comprehensive overview,” *IEEE Access*, vol. 7, pp. 81 555–81 573, 2019.
- [20] T. Cheesman, “delightedbeauty.org,” <http://www.delightedbeauty.org/>, 2011, accessed: 2017-02-16.
- [21] A. Šilić and B. Bašić, “Visualization of text streams: A survey,” *Knowledge-based and intelligent information and engineering systems*, pp. 31–43, 2010.
- [22] K. Kucher and A. Kerren, “Text visualization techniques: Taxonomy, visual survey, and community insights,” in *IEEE Pacific Visualization Symposium (PacificVis)*, 2015, pp. 117–121.
- [23] S. Liu, X. Wang, C. Collins, W. Dou, F. Ouyang, M. El-Assady, L. Jiang, and D. Keim, “Bridging text visualization and mining: A task-driven survey,” *IEEE Transactions on Visualization and Computer Graphics*, 2018.
- [24] S. Jänicke, G. Franzini, M. F. Cheema, and G. Scheuermann, “On Close and Distant Reading in Digital Humanities: A Survey and Future Challenges,” in *Eurographics Conference on Visualization (EuroVis) - STARs*, 2015, pp. 83–103.
- [25] F. Wanner, A. Stoffel, D. Jäckle, B. Kwon, A. Weiler, and D. A. Keim, “State-of-the-art report of visual analysis for event detection in text data streams,” in *Eurographics Conference on Visualization (EuroVis) - STARs*, 2014, pp. 125–139.

-
- [26] S. Jänicke, G. Franzini, M. F. Cheema, and G. Scheuermann, “Visual Text Analysis in Digital Humanities,” *Computer Graphics Forum*, vol. 36, no. 6, pp. 226–250, sep 2017. [Online]. Available: <http://doi.wiley.com/10.1111/cgf.12873>
- [27] A. Mueller, “word_cloud: A little word cloud generator in python,” https://github.com/amueller/word_cloud, accessed: 2018-12-15.
- [28] M. Alharbi and R. S. Laramee, “Parallel coordinates of sos,” http://cs.swan.ac.uk/~msalharbi/pc_public/, accessed: 2018-11-15.
- [29] W. Buhss, *William Shakespeare Othello, Venedigs Neger*. Berlin: Henschel Schauspiel Theaterverlag, 1996.
- [30] F. Günther, *William Shakespeare. Othello. Zweisprachige Ausgabe*. Munich: Deutscher Taschenbuch Verlag, 1995.
- [31] H. Schwarz, *Othello, der Maure von Venedig*. Typescript. Shakespeare-Bibliothek München, 1941.
- [32] W. Baudissin, *Shakespeares dramatische Werke*. Berlin: Reimer, 1832, vol. 8.
- [33] L. Bärfuss, *Othello*. Hartmann & Stauffacher, 2001.
- [34] E. Engel, *William Shakespeare Othello*. Berlin: Felix Bloch Erben, 1939.
- [35] S. Miksch and W. Aigner, “A matter of time: Applying a data–users–tasks design triangle to visual analytics of time-oriented data,” *Computers & Graphics*, vol. 38, pp. 286 – 290, 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0097849313001817>
- [36] O. Dictionary, “Data visualization definition,” https://en.oxforddictionaries.com/definition/data_visualization, accessed: 2020-12-15.
- [37] M. O. Ward, G. Grinstein, and D. Keim, *Interactive data visualization: foundations, techniques, and applications*. CRC Press, 2010.
- [38] M. Chen, D. Ebert, H. Hagen, R. S. Laramee, R. van Liere, K. L. Ma, W. Ribarsky, G. Scheuermann, and D. Silver, “Data, information, and knowledge in visualization,” *IEEE Computer Graphics and Applications*, vol. 29, no. 1, pp. 12–19, 2009.
- [39] M. Friendly, “A brief history of data visualization,” in *Handbook of data visualization*. Springer, 2008, pp. 15–56.
- [40] ACM, “Acm computing classification system,” <https://dl.acm.org/ccs/>, accessed: 2020-12-15.
- [41] L. McNabb and R. S. Laramee, “Survey of Surveys (SoS) - Mapping The Landscape of Survey Papers in Information Visualization,” *Computer Graphics Forum*, 2017.

- [42] D. Rees and R. S. Laramée, “A survey of information visualization books,” *Computer Graphics Forum*, vol. 38, no. 1, pp. 610–646, 2019.
- [43] D. A. Keim, F. Mansmann, J. Schneidewind, and H. Ziegler, “Challenges in visual data analysis,” in *Tenth International Conference on Information Visualisation (IV’06)*, 2006, pp. 9–16.
- [44] S. Card, J. Mackinlay, and B. Shneiderman, “Information visualization,” *Human-computer interaction: Design issues, solutions, and applications*, vol. 181, 2009.
- [45] W. Hersh, *Information retrieval: a health and biomedical perspective*. Springer Science & Business Media, 2008.
- [46] A. Inselberg, “The plane with parallel coordinates,” *The Visual Computer*, vol. 1, no. 2, pp. 69–91, 1985. [Online]. Available: <https://doi.org/10.1007/BF01898350>
- [47] B. Johnson and B. Shneiderman, “Tree-maps: a space-filling approach to the visualization of hierarchical information structures,” in *Proceeding Visualization ’91*, 1991, pp. 284–291.
- [48] M. Bruls, K. Huizing, and J. J. van Wijk, “Squarified treemaps,” in *Data Visualization 2000*, W. C. de Leeuw and R. van Liere, Eds. Vienna: Springer Vienna, 2000, pp. 33–42.
- [49] B. Shneiderman and M. Wattenberg, “Ordered treemap layouts,” in *Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS’01)*, ser. INFOVIS ’01. USA: IEEE Computer Society, 2001, p. 73.
- [50] Y. Tu and H. Shen, “Balloon focus: a seamless multi-focus+context method for treemaps,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1157–1164, 2008.
- [51] R. C. Roberts, C. Tong, R. S. Laramée, G. A. Smith, P. Brookes, and T. D’Cruze, “Interactive Analytical Treemaps for Visualisation of Call Centre Data,” in *Smart Tools and Apps for Graphics - Eurographics Italian Chapter Conference*, G. Pintore and F. Stanco, Eds. The Eurographics Association, 2016.
- [52] R. G. Raidou, M. Eisemann, M. Breeuwer, E. Eisemann, and A. Vilanova, “Orientation-enhanced parallel coordinate plots,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 589–598, 2016.
- [53] J. J. Thomas, *Illuminating the path:[the research and development agenda for visual analytics]*. IEEE Computer Society, 2005.
- [54] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon, *Visual Analytics: Definition, Process, and Challenges*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 154–175. [Online]. Available: https://doi.org/10.1007/978-3-540-70956-5_7

-
- [55] T. Cheesman, K. Flanagan, and S. Thiel, “translation array prototype 1: Project overview,” <http://delightedbeauty.org/>, accessed on 07.06.2019.
- [56] T. Cheesman, K. Flanagan, S. Thiel, J. Rybicki, R. S. Laramée, J. Hope, and A. Roos, “Multi-retranslation corpora: Visibility, variation, value, and virtue,” *Digital Scholarship in the Humanities*, vol. 32, no. 4, pp. 739–760, 2017.
- [57] T. Cheesman and A. Roos, “Version Variation Visualization (VVV): Case Studies on the Hebrew Haggadah in English,” *Journal of Data Mining & Digital Humanities*, vol. Special Issue on Computer-Aided Processing of Intertextuality in Ancient Languages, Jul. 2017.
- [58] Z. Geng, R. S. Laramée, T. Cheesman, A. Ehrmann, and D. M. Berry, “Visualizing translation variation: Shakespeare’s Othello,” in *Advances in Visual Computing*, G. Bebis, R. Boyle, B. Parvin, D. Koracin, S. Wang, K. Kyungnam, B. Benes, K. Moreland, C. Borst, S. DiVerdi, C. Yi-Jen, and J. Ming, Eds., 2011, pp. 653–663.
- [59] “World writings compared by swansea university web tool - bbc news,” <https://www.bbc.co.uk/news/uk-wales-south-west-wales-19561879>, accessed: 2020-10-30.
- [60] T. Cheesman, K. Flanagan, and R. S. Laramée, “Visualizing variation in collections of translations and adaptations of cultural heritage texts,” in *The Search Is Over! Exploring Cultural Collections with Visualization (TSIO) in conjunction with, Digital Libraries 2014*, London, UK, September 2014.
- [61] “Centre on digital arts and humanities – centre on digital arts and humanities,” <https://codah.swan.ac.uk/>, accessed: 2020-10-30.
- [62] M. Alrehiely and R. S. Laramée, *Visualization Of Version Variation*, 1st ed. Lambert Academic Publishing, 1 2015.
- [63] “Translations and visualizations at tedx swansea,” <https://www.youtube.com/watch?v=K84rCIMhJ4&t=733s/>, accessed: 2020-10-30.
- [64] Z. Geng, T. Cheesman, R. S. Laramée, K. Flanagan, and S. Thiel, “Shakervis: Visual analysis of segment variation of german translations of shakespeare’s othello,” *Information Visualization*, vol. 14, no. 4, pp. 273–288, 2015. [Online]. Available: <https://doi.org/10.1177/1473871613495845>
- [65] X. Liu, R. S. Laramée, and S. Walton, *Interactive Visualisation of Shakespeares Othello*, 1st ed. Lambert Academic Publishing, 02 2018.
- [66] “TransVis Presentation for the IEEE VIS Conference 2020,” <https://youtu.be/8HsUpCiA4gc/>, accessed: 2021-02-11.

- [67] A. Abdul-Rahman, J. Lein, K. Coles, E. Maguire, M. Meyer, M. Wynne, C. R. Johnson, A. Trefethen, and M. Chen, "Rule-based visual mappings—with a case study on poetry visualization," in *Proc. of the 15th Eurographics Conference on Visualization (EuroVis)*, 2013, p. 381–390.
- [68] U. Hinrichs, S. Forlini, and B. Moynihan, "In defense of sandcastles: Research thinking through visualization in digital humanities," *Digital Scholarship in the Humanities*, vol. 34, no. 1, pp. i80–i99, 10 2018. [Online]. Available: <https://doi.org/10.1093/lc/fqy051>
- [69] H. Van Den Berg, A. Betti, T. Castermans, R. Koopman, B. Speckmann, K. Verbeek, T. Van der Werf, S. Wang, M. A. Westenberg *et al.*, "A philosophical perspective on visualization for digital humanities," in *Proc. 3rd Workshop on Visualization for the Digital Humanities (VIS4DH)*, 2018.
- [70] T. Hogan, U. Hinrichs, and E. Hornecker, "The elicitation interview technique: Capturing people's experiences of data representations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2579–2593, 2016.
- [71] M. Alharbi and R. Laramée, "Sos textvis: An extended survey of surveys on text visualization," *Computers*, vol. 8, no. 1, 2019. [Online]. Available: <https://www.mdpi.com/2073-431X/8/1/17>
- [72] R. Reddy and G. StClair, "The million book digital library project," *Computer Science Presentation*, 2001.
- [73] P. Q. Andre and N. L. Eaton, "National agricultural text digitizing project," *Library Hi Tech*, vol. 6, no. 3, pp. 61–66, 1988.
- [74] D. Mendelsson, E. Falk, and A. L. Oliver, "The albert einstein archives digitization project: opening hidden treasures," *Library Hi Tech*, vol. 32, no. 2, pp. 318–335, 2014.
- [75] K. Kucher, C. Paradis, and A. Kerren, "The state of the art in sentiment visualization," in *Computer Graphics Forum*. Wiley Online Library, 2017.
- [76] G.-D. Sun, Y.-C. Wu, R.-H. Liang, and S.-X. Liu, "A survey of visual analytics techniques and applications: State-of-the-art research and future challenges," *Journal of Computer Science and Technology*, vol. 28, no. 5, pp. 852–867, 2013.
- [77] S. Liu, W. Cui, Y. Wu, and M. Liu, "A survey on information visualization: recent advances and challenges," *The Visual Computer*, vol. 30, no. 12, pp. 1373–1393, 2014.
- [78] V. Gupta and G. S. Lehal, "A survey of text summarization extractive techniques," *Journal of emerging technologies in web intelligence*, vol. 2, no. 3, pp. 258–268, 2010.
- [79] C. C. Aggarwal and C. Zhai, "A survey of text clustering algorithms," in *Mining text data*. Springer, 2012, pp. 77–128.

-
- [80] K. Jung, K. I. Kim, and A. K. Jain, "Text information extraction in images and video: a survey," *Pattern recognition*, vol. 37, no. 5, pp. 977–997, 2004.
- [81] S. K. Card, J. D. Mackinlay, and B. Shneiderman, *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.
- [82] A. B. Alencar, M. C. F. de Oliveira, and F. V. Paulovich, "Seeing beyond reading: a survey on visual text analytics," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, pp. 476–492, 2012.
- [83] Q. Gan, M. Zhu, M. Li, T. Liang, Y. Cao, and B. Zhou, "Document visualization: An overview of current research," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 6, pp. 19–36, 2014.
- [84] J. Nualart-Vilaplana, M. Pérez-Montoro, and M. Whitelaw, "How we draw texts: a review of approaches to text visualization and exploration," *El profesional de la información*, vol. 23, pp. 221–235, 2014.
- [85] N. Cao and W. Cui, *Overview of Text Visualization Techniques*. Paris: Atlantis Press, 2016, pp. 11–40. [Online]. Available: https://doi.org/10.2991/978-94-6239-186-4_2
- [86] P. Federico, F. Heimerl, S. Koch, and S. Miksch, "A survey on visual approaches for analyzing scientific literature and patents," *IEEE Transactions on Visualization and Computer Graphics*, 2016.
- [87] D. Steinbock, "Tagcrowd," *Internet URL: http://www.tagcrowd.com/blog/about/[accessed 2018-2-13]*, 2014.
- [88] F. B. Viegas, M. Wattenberg, and J. Feinberg, "Participatory visualization with wordle," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1137–1144, 2009.
- [89] A. Skupin, "A cartographic approach to visualizing conference abstracts," *IEEE Computer Graphics and Applications*, vol. 22, no. 1, pp. 50–58, 2002.
- [90] J. A. Wise, "The ecological approach to text visualization," *Journal of the Association for Information Science and Technology*, vol. 50, no. 13, p. 1224, 1999.
- [91] K. Andrews, W. Kienreich, V. Sabol, J. Becker, G. Droschl, F. Kappe, M. Granitzer, P. Auer, and K. Tochtermann, "The infosky visual explorer: exploiting hierarchical structure and document similarities," *Information Visualization*, vol. 1, no. 3-4, pp. 166–181, 2002.
- [92] H. Strobel, D. Oelke, C. Rohrdantz, A. Stoffel, D. A. Keim, and O. Deussen, "Document cards: A top trumps visualization for documents," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1145–1152, 2009.

- [93] J.-W. Liu and L.-C. Huang, "Detecting and visualizing emerging trends and transient patterns in fuel cell scientific literature," in *International Conference on Wireless Communications, Networking and Mobile Computing*. IEEE, 2008, pp. 1–4.
- [94] B. Lee, N. H. Riche, A. K. Karlson, and S. Carpendale, "Sparkclouds: Visualizing trends in tag clouds," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1182–1189, 2010.
- [95] M. A. Hearst, "Tilebars: visualization of term distribution information in full text information access," in *Proc. of the SIGCHI conference on Human factors in computing systems*. ACM Press/Addison-Wesley Publishing Co., 1995, pp. 59–66.
- [96] B. Shneiderman, "The eyes have it: a task by data type taxonomy for information visualizations," in *Proceedings 1996 IEEE Symposium on Visual Languages*, Sep. 1996, pp. 336–343.
- [97] N. Andrienko and G. Andrienko, *Exploratory analysis of spatial and temporal data: a systematic approach*. Springer Science & Business Media, 2006.
- [98] F. Moretti, *Graphs, maps, trees: abstract models for a literary history*. Verso, 2005.
- [99] "Google books," <https://books.google.com/>, accessed: 2017-04-17.
- [100] D. Keim, G. Ellis, and F. Mansmann, "Mastering the information age solving problems with visual analytics," in *Eurographics*, vol. 2, 2010, p. 5.
- [101] W. Cui, S. Liu, L. Tan, C. Shi, Y. Song, Z. Gao, H. Qu, and X. Tong, "Textflow: Towards better understanding of evolving topics in text," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2412–2421, 2011.
- [102] D. Luo, J. Yang, M. Krstajic, W. Ribarsky, and D. Keim, "Eventriver: Visually exploring text collections with temporal references," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 1, pp. 93–105, 2012.
- [103] N. Cao, J. Sun, Y.-R. Lin, D. Gotz, S. Liu, and H. Qu, "Facetatlas: Multifaceted visualization for rich text corpora," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1172–1181, 2010.
- [104] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.
- [105] K. W. Church and J. I. Helfman, "Dotplot: A Program for Exploring Self-Similarity in Millions of Lines of Text and Code," *Journal of Computational and Graphical Statistics*, vol. 2, no. 2, pp. 153–174, 1993.

- [106] M. Wattenberg, “Arc diagrams: Visualizing structure in strings,” in *Proceedings - IEEE Symposium on Information Visualization, INFO VIS*, vol. 2002-Janua. Institute of Electrical and Electronics Engineers Inc., 2002, pp. 110–116.
- [107] M. Wattenberg and F. B. Viégas, “The word tree, an interactive visual concordance,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, 2008.
- [108] C. Collins, F. B. Viegas, and M. Wattenberg, “Parallel tag clouds to explore and analyze faceted text corpora,” in *2009 IEEE Symposium on Visual Analytics Science and Technology*, Oct 2009, pp. 91–98.
- [109] C. Collins, S. Carpendale, and G. Penn, “DocuBurst: Visualizing Document Content using Language Structure,” *Computer Graphics Forum*, vol. 28, no. 3, pp. 1039–1046, jun 2009. [Online]. Available: <http://doi.wiley.com/10.1111/j.1467-8659.2009.01439.x>
- [110] F. van Ham, M. Wattenberg, and F. B. Viegas, “Mapping text with phrase nets,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1169–1176, 2009.
- [111] S. Jänicke, A. Geßner, M. Büchler, and G. Scheuermann, “Visualizations for text re-use,” in *International Conference on Information Visualization Theory and Applications (IVAPP)*, 2014, pp. 59–70.
- [112] D. Oelke, H. Strobel, C. Rohrdantz, I. Gurevych, and O. Deussen, “Comparative Exploration of Document Collections: a Visual Analytics Approach,” *Computer Graphics Forum*, vol. 33, no. 3, pp. 201–210, jun 2014. [Online]. Available: <http://doi.wiley.com/10.1111/cgf.12376>
- [113] S. Jänicke, A. Geßner, G. Franzini, M. Terras, S. Mahony, and G. Scheuermann, “TRAViz: A visualization for Variant Graphs,” *Digital Scholarship in the Humanities*, vol. 30, no. suppl_1, pp. i83–i99, dec 2015. [Online]. Available: https://academic.oup.com/dsh/article/30/suppl_{_}1/i83/365029
- [114] P. Riehmann, M. Potthast, B. Stein, and B. Froehlich, “Visual Assessment of Alleged Plagiarism Cases,” *Computer Graphics Forum*, vol. 34, no. 3, pp. 61–70, jun 2015. [Online]. Available: <http://doi.wiley.com/10.1111/cgf.12618>
- [115] A. Abdul-Rahman, G. Roe, M. Olsen, C. Gladstone, R. Whaling, N. Cronk, R. Morrissey, and M. Chen, “Constructive visual analytics for text similarity detection,” *Computer Graphics Forum*, vol. 36, no. 1, pp. 237–248, 2017. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.12798>
- [116] S. Jänicke and D. J. Wisley, “Interactive visual alignment of medieval text versions,” in *IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2017, pp. 127–138.

- [117] M. Hu, K. Wongsuphasawat, and J. Stasko, “Visualizing Social Media Content with SentenTree,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 621–630, jan 2017.
- [118] D. Gunning, “Explainable artificial intelligence (xai),” *Defense Advanced Research Projects Agency (DARPA), nd Web*, vol. 2, no. 2, 2017.
- [119] F. K. Došilović, M. Brčić, and N. Hlupić, “Explainable artificial intelligence: A survey,” in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2018, pp. 0210–0215.
- [120] A. Adadi and M. Berrada, “Peeking inside the black-box: A survey on explainable artificial intelligence (xai),” *IEEE Access*, vol. 6, pp. 52 138–52 160, 2018.
- [121] F. Hohman, M. Kahng, R. Pienta, and D. H. Chau, “Visual analytics in deep learning: An interrogative survey for the next frontiers,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 8, pp. 2674–2693, 2019.
- [122] A. Chatzimparmpas, R. M. Martins, I. Jusufi, and A. Kerren, “A survey of surveys on the use of visualization for interpreting machine learning models,” *Information Visualization*, vol. 0, no. 0, 2020. [Online]. Available: <https://doi.org/10.1177/1473871620904671>
- [123] A. Endert, W. Ribarsky, C. Turkay, B. W. Wong, I. Nabney, I. D. Blanco, and F. Rossi, “The state of the art in integrating machine learning into visual analytics,” *Computer Graphics Forum*, vol. 36, no. 8, pp. 458–486, 2017. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13092>
- [124] J. Choo and S. Liu, “Visual analytics for explainable deep learning,” *IEEE Computer Graphics and Applications*, vol. 38, no. 4, pp. 84–92, 2018.
- [125] D. Smilkov, S. Carter, D. Sculley, F. B. Viégas, and M. Wattenberg, “Direct-manipulation visualization of deep networks,” *ArXiv*, vol. abs/1708.03788, 2017.
- [126] D. Smilkov, N. Thorat, C. Nicholson, E. Reif, F. B. Viégas, and M. Wattenberg, “Embedding Projector: Interactive Visualization and Interpretation of Embeddings,” nov 2016. [Online]. Available: <http://arxiv.org/abs/1611.05469>
- [127] Y. Liu, E. Jun, Q. Li, and J. Heer, “Latent Space Cartography: Visual Analysis of Vector Space Embeddings,” *Computer Graphics Forum*, vol. 38, no. 3, pp. 67–78, jun 2019. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13672>
- [128] J. Krause, A. Perer, and K. Ng, “Interacting with predictions: Visual inspection of black-box machine learning models,” in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ser. CHI ’16. New York, NY, USA: Association for Computing Machinery, 2016, p. 5686–5697. [Online]. Available: <https://doi.org/10.1145/2858036.2858529>

- [129] C. B. Azodi, J. Tang, and S. H. Shiu, “Opening the Black Box: Interpretable Machine Learning for Geneticists,” pp. 442–455, jun 2020. [Online]. Available: <http://www.cell.com/article/S016895252030069X/fulltext><http://www.cell.com/article/S016895252030069X/abstract>[https://www.cell.com/trends/genetics/abstract/S0168-9525\(20\)30069-X](https://www.cell.com/trends/genetics/abstract/S0168-9525(20)30069-X)
- [130] T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady, “explainer: A visual analytics framework for interactive and explainable machine learning,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 1064–1074, 2020.
- [131] I. Tenney, J. Wexler, J. Bastings, T. Bolukbasi, A. Coenen, S. Gehrmann, E. Jiang, M. Pushkarna, C. Radebaugh, E. Reif, and A. Yuan, “The language interpretability tool: Extensible, interactive visualizations and analysis for nlp models,” 2020.
- [132] Y. Belinkov, S. Gehrmann, and E. Pavlick, “Interpretability and analysis in neural NLP,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*. Online: Association for Computational Linguistics, jul 2020, pp. 1–5. [Online]. Available: <https://www.aclweb.org/anthology/2020.acl-tutorials.1>
- [133] J. Zhang, Y. Wang, P. Molino, L. Li, and D. S. Ebert, “Manifold: A model-agnostic framework for interpretation and diagnosis of machine learning models,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 364–373, 2019.
- [134] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, “Beyond accuracy: Behavioral testing of NLP models with CheckList,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 4902–4912. [Online]. Available: <https://www.aclweb.org/anthology/2020.acl-main.442>
- [135] A. Gibbs, “The Diagram, a Method for Comparing Sequences. Its Use with Amino Acid and Nucleotide Sequences The phylogenetics of the global population of potato virus Y and its necrogenic recombinants. View project,” *Article in European Journal of Biochemistry*, 2005. [Online]. Available: <https://www.researchgate.net/publication/229616013>
- [136] S. Silvia, R. Etemadpour, J. Abbas, S. Huskey, and C. Weaver, “Visualizing Variation in Classical Text with Force Directed Storylines,” in *Proceedings of the Workshop on Visualization for the Digital Humanities*. IEEE, oct 2016. [Online]. Available: <http://www.juxtapsoftware>.
- [137] A. Inselberg and B. Dimsdale, “Parallel coordinates: A tool for visualizing multi-dimensional geometry.” Publ by IEEE, 1990, pp. 361–378.
- [138] Q. Castellà and C. Sutton, “Word storms: Multiples of word clouds for visual comparison of documents,” in *WWW 2014 - Proceedings of the 23rd International Conference on World Wide Web*. New York, New York, USA: Association for Computing Machinery, Inc, apr 2014, pp. 665–675. [Online]. Available: <http://dl.acm.org/citation.cfm?doid=2566486.2567977>

- [139] Y. Wang, X. Chu, C. Bao, L. Zhu, O. Deussen, B. Chen, and M. Sedlmair, “EdWordle: Consistency-Preserving Word Cloud Editing,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 647–656, jan 2018.
- [140] C. Fellbaum, *WordNet*. American Cancer Society, 2012. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781405198431.wbeal1285>
- [141] I. D. Melamed, “Manual annotation of translational equivalence: The blinker project,” *Technical Report 98-07, Institute for Research in Cognitive Science*, 1998.
- [142] N. A. Smith and M. E. Jahr, “Cairo: An alignment visualization tool,” in *The International Conference on Language Resources and Evaluation*, 2000, pp. 552–554.
- [143] D. Tufiş, “From word alignment to word senses, via multilingual wordnets,” *The Computer Science Journal of Moldova (CSJM)*, vol. 14, no. 1, pp. 3–33, 2006.
- [144] U. Germann, “Yawat :yet another word alignment tool,” in *The ACL-08: HLT Demo Session*, 2008, pp. 20–23.
- [145] T. Gilmanov, O. Scrivner, and S. Kübler, “Swift aligner, a multifunctional tool for parallel corpora: Visualization, word alignment, and (morpho)-syntactic cross-language transfer,” in *LREC*, 2014, pp. 2913–2919.
- [146] A. Abdul-Rahman, G. Roe, M. Olsen, C. Gladstone, R. Whaling, N. Cronk, R. Morrissey, and M. Chen, “Constructive visual analytics for text similarity detection,” *Computer Graphics Forum*, vol. 36, no. 1, pp. 237–248, 2017.
- [147] D. Briel, “Bligner,” <http://bligner.aligner.free.fr/>, accessed on 07.06.2019.
- [148] M. L. Forcada and R. Martin, “bitext2tmx: Bitext aligner/converter,” <http://bitext2tmx.sourceforge.net/>, accessed on 07.06.2019.
- [149] LinguisTech, “Sdl trados winalign,” https://linguistech.ca/SDLTrados_WinAlign_E_TUTCERTT_I_PartI, accessed on 07.06.2019.
- [150] L. Ahrenberg, M. Merkel, and M. Petterstedt, “Interactive word alignment for language engineering,” in *European Chapter of the Association for Computational Linguistics*, 2003, pp. 49–52.
- [151] A. Farkas, “LF aligner,” <http://sourceforge.net/projects/aligner/>, accessed on 29.5.2019.
- [152] M. Jankowska, V. Kešelj, and E. Milios, “Relative n-gram signatures: Document visualization at the level of character n-grams,” in *IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2012, pp. 103–112.
- [153] S. Schreibman, A. Kumar, and J. McDonald, “The versioning machine,” *Literary and Linguistic Computing*, vol. 18, pp. 101–107, 2003.

- [154] D. Wheeles and K. Jensen, “Juxta commons,” *Proceedings of the Digital Humanities*, vol. 5, p. 12, 2013.
- [155] R. L. Ribler and M. Abrams, “Using visualization to detect plagiarism in computer science classes,” in *The IEEE Symposium on Information Visualization*, 2000, pp. 173–178.
- [156] D. R. White and M. S. Joy, “Sentence-based natural language plagiarism detection,” *Journal on Educational Resources in Computing*, vol. 4, no. 4, pp. 1–20, 2004.
- [157] M. Freire, “Visualizing program similarity in the ac plagiarism detection system,” in *The Working Conference on Advanced Visual Interfaces(AVI)*, 2008, pp. 404–407.
- [158] M. Inc, “Microsoft support: How to use the windiff.exe utility,” <http://support.microsoft.com/KB/159214,2014/>, accessed on 15.06.2019.
- [159] V. Frick, C. Wedenig, and M. Pinzger, “Diffviz: A diff algorithm independent visualization tool for edit scripts,” in *2018 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, 2018, pp. 705–709.
- [160] L. Voinea, A. Telea, and J. J. van Wijk, “Cvsscan: Visualization of code evolution,” in *Proceedings of the 2005 ACM Symposium on Software Visualization*, 2005, pp. 47–56.
- [161] A. Telea and D. Auber, “Code flows: Visualizing structural evolution of source code,” *Computer Graphics Forum*, vol. 27, no. 3, pp. 831–838, 2008.
- [162] L. McNabb and R. S. Laramee, “Survey of surveys (SoS)-mapping the landscape of survey papers in information visualization,” *Computer Graphics Forum*, vol. 36, no. 3, pp. 589–617, 2017.
- [163] P. Caserta and O. Zendra, “Visualization of the static aspects of software: A survey,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 7, pp. 913–933, 2010.
- [164] A.-L. Mattila, P. Ihantola, T. Kilamo, A. Luoto, M. Nurminen, and H. Väättäjä, “Software visualization today: Systematic literature review,” in *Proceedings of the 20th International Academic Mindtrek Conference*, 2016, pp. 262–271.
- [165] L. Merino, M. Ghafari, C. Anslow, and O. Nierstrasz, “A systematic literature review of software visualization evaluation,” *Journal of Systems and Software*, vol. 144, pp. 165–180, 2018.
- [166] R. L. Novais, A. Torres, T. S. Mendes, M. Mendonça, and N. Zazworka, “Software evolution visualization: A systematic mapping study,” *Information and Software Technology*, vol. 55, no. 11, pp. 1860–1883, 2013.
- [167] C. Monroy, R. Kochumman, R. Furuta, and E. Urbina, *Interactive Timeline Viewer (ItLv): A Tool to Visualize Variants among Documents*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 39–49.

- [168] C. Jong, P. Rajkumar, B. Siddiquie, T. Clement, C. Plaisant, and B. Shneiderman, "Interactive exploration of versions across multiple documents," *Proceedings of the Digital Humanities 2009*, 2009.
- [169] M. Büchler, A. Geßner, G. Heyer, and T. Eckart, "Detection of citations and textual reuse on ancient greek texts and its applications in the classical studies: eaqua project," in *Digital Humanities*, 2010, pp. 113–114.
- [170] J. Walsh and W. Hooper, "Computational discovery and visualization of the underlying semantic structure of complicated historical and literary corpora," in *Proc. Digital Humanities*, 2011, pp. 10–11.
- [171] M. Behrisch, M. Krstajic, and T. Schreck, "The news auditor: Visual exploration of clusters of stories," in *EuroVA 2012 International Workshop on Visual Analytics*. Eurographics Association, 2012, pp. 61–65.
- [172] S. Howell, M. Kelleher, A. Teehan, and J. Keating, "A Digital Humanities Approach to Narrative Voice in The Secret Scripture: Proposing a New Research Method," *Digital Humanities Quarterly*, vol. 8, no. 2, 2014.
- [173] S. Jänicke and A. Geßner, "A distant reading visualization for variant graphs," in *Conference Abstracts of the Digital Humanities*, 2015.
- [174] B. Asokarajan, R. Etemadpour, J. Abbas, S. Huskey, and C. Weaver, "Visualization of Latin Textual Variants using a Pixel-Based Text Analysis Tool," in *EuroVis Workshop on Visual Analytics (EuroVA)*, N. Andrienko and M. Sedlmair, Eds. The Eurographics Association, 2016.
- [175] S. Jänicke and D. Joseph Wrisley, "Visualizing mouvance: Toward a visual analysis of variant medieval text traditions," *Digital Scholarship in the Humanities*, vol. 32, pp. 106–123, 2017.
- [176] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts, "Visual comparison for information visualization," *Information Visualization*, vol. 10, no. 4, pp. 289–309, 2011.
- [177] M. Brehmer and T. Munzner, "A multi-level typology of abstract visualization tasks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2376–2385, 2013.
- [178] R. E. Roth, "An empirically-derived taxonomy of interaction primitives for interactive cartography and geovisualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2356–2365, 2013.
- [179] J. S. Yi, Y. ah Kang, and J. Stasko, "Toward a deeper understanding of the role of interaction in information visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1224–1231, 2007.

-
- [180] G. Andrienko, N. Andrienko, J. Dykes, S. I. Fabrikant, and M. Wachowicz, “Geovisualization of dynamics, movement and change: Key issues and developing approaches in visualization research,” *Information Visualization*, vol. 7, no. 3-4, pp. 173–180, 2008.
- [181] M. Gleicher, “Considerations for visualizing comparison,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 413–423, 2018.
- [182] J. Heer and G. Robertson, “Animated transitions in statistical data graphics,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 6, pp. 1240–1247, 2007.
- [183] D. A. Keim and D. Oelke, “Literature fingerprinting: A new method for visual literary analysis,” in *IEEE Symposium on Visual Analytics Science and Technology*, 2007, pp. 115–122.
- [184] D. Oelke, D. Kokkinakis, and D. A. Keim, “Fingerprint matrices: Uncovering the dynamics of social networks in prose literature,” *Computer Graphics Forum*, vol. 32, no. 3pt4, pp. 371–380, 2013.
- [185] H. Siirtola, T. Säily, T. Nevalainen, and K.-J. Räihä, “Text variation explorer: Towards interactive visualization tools for corpus linguistics,” *International Journal of Corpus Linguistics*, vol. 19, no. 3, pp. 417–429, 2013.
- [186] S. Koch, M. John, M. Wörner, A. Müller, and T. Ertl, “Varifocalreader—in-depth visual analysis of large text documents,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 1723–1732, 2014.
- [187] M. Brehmer, S. Ingram, J. Stray, and T. Munzner, “Overview: The design, adoption, and analysis of a visual document mining tool for investigative journalists,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 12, pp. 2271–2280, 2014.
- [188] S. Havre, E. Hetzler, K. Perrine, E. Jurrus, and N. Miller, “Interactive visualization of multiple query results,” in *Proc. of the IEEE Symposium on Information Visualization 2001*, 2001, p. 105.
- [189] E. Suvanaphen and J. C. Roberts, “Textual difference visualization of multiple search results utilizing detail in context,” in *Proc. Theory and Practice of Computer Graphics, 2004.*, 2004, pp. 2–8.
- [190] S. Jänicke, A. Geßner, M. Büchler, and G. Scheuermann, “5 design rules for visualizing text variant graphs,” in *Digital Humanities*, 2014, p. 12.
- [191] H. Strobelt, S. Gehrmann, H. Pfister, and A. M. Rush, “Lstmvis: A tool for visual analysis of hidden state dynamics in recurrent neural networks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 1, pp. 667–676, 2018.
- [192] “BLACK BOX | meaning in the Cambridge English Dictionary.” [Online]. Available: <https://dictionary.cambridge.org/dictionary/english/black-box>

- [193] “Black Box | Definition of Black Box by Merriam-Webster.” [Online]. Available: <https://www.merriam-webster.com/dictionary/blackbox>
- [194] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, “A survey of methods for explaining black box models,” *ACM Computing Surveys*, vol. 51, no. 5, Aug. 2018. [Online]. Available: <https://doi.org/10.1145/3236009>
- [195] S. Feldman, “Nlp meets the jabberwocky: Natural language processing in information retrieval.” *Online*, vol. 23, no. 3, pp. 62 – 64, 1999.
- [196] I. Pak and P. L. Teh, “Text segmentation techniques: a critical review,” in *Innovative Computing, Optimization and Its Applications*. Springer, 2018, pp. 167–181.
- [197] G. Salton, A. Singhal, C. Buckley, and M. Mitra, “Automatic text decomposition using text segments and text themes,” in *Proceedings of the the Seventh ACM Conference on Hypertext*, ser. HYPERTEXT ’96. New York, NY, USA: Association for Computing Machinery, 1996, p. 53–65. [Online]. Available: <https://doi.org/10.1145/234828.234834>
- [198] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. USA: Cambridge University Press, 2008.
- [199] O. E. Dictionary and E. Idioms, “Oxford references online,” 1989.
- [200] M. Makrehchi and M. S. Kamel, “Automatic extraction of domain-specific stopwords from labeled documents,” in *Advances in Information Retrieval*, C. Macdonald, I. Ounis, V. Plachouras, I. Ruthven, and R. W. White, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 222–233.
- [201] C. Silva and B. Ribeiro, “The importance of stop word removal on recall values in text categorization,” in *Proceedings of the International Joint Conference on Neural Networks, 2003.*, vol. 3, July 2003, pp. 1661–1666 vol.3.
- [202] H. Joshi, J. Pareek, R. Patel, and K. Chauhan, “To stop or not to stop — experiments on stop-word elimination for information retrieval of gujarati text documents,” in *2012 Nirma University International Conference on Engineering (NUiCONE)*, Dec 2012, pp. 1–4.
- [203] D. Na and C. Xu, “Automatically generation and evaluation of stop words list for chinese patents,” *Telkomnika*, vol. 13, no. 4, p. 1414, 2015.
- [204] W. B. Frakes, *Stemming Algorithms*. USA: Prentice-Hall, Inc., 1992, p. 131–160.
- [205] C. Sönmez and A. Özgür, “A graph-based approach for contextual text normalization,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 313–324. [Online]. Available: <https://www.aclweb.org/anthology/D14-1037>

- [206] J. B. Lovins, "Development of a stemming algorithm," *Mech. Translat. & Comp. Linguistics*, vol. 11, no. 1-2, pp. 22–31, 1968.
- [207] G. M. Di Nunzio and F. Vezzani, *A Linguistic Failure Analysis of Classification of Medical Publications: A Study on Stemming vs Lemmatization*. Accademia University Press, 2019.
- [208] T. Korenius, J. Laurikkala, K. Järvelin, and M. Juhola, "Stemming and lemmatization in the clustering of finnish text documents," in *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, ser. CIKM '04. New York, NY, USA: Association for Computing Machinery, 2004, p. 625–633. [Online]. Available: <https://doi.org/10.1145/1031171.1031285>
- [209] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in *International conference on machine learning*, 2015, pp. 957–966.
- [210] S. E. Robertson and K. S. Jones, "Relevance weighting of search terms," *Journal of the Association for Information Science and Technology*, vol. 27, no. 3, pp. 129–146, 1976.
- [211] S. E. Robertson, "Overview of the Okapi projects," *Journal of Documentation*, vol. 53, no. 1, pp. 3–7, 1997.
- [212] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.
- [213] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.
- [214] A. Huang, "Similarity measures for text document clustering," in *The sixth new zealand computer science research student conference*, vol. 4, 2008, pp. 49–56.
- [215] S. Robertson, "Understanding inverse document frequency: on theoretical arguments for idf," *Journal of Documentation*, vol. 60, no. 5, pp. 503–520, 2004.
- [216] D. Sarkar, *Text Analytics with python*. Springer, 2016.
- [217] M. Harrower and C. A. Brewer, "Colorbrewer.org: An online tool for selecting colour schemes for maps," *The Cartographics Journal*, vol. 40, pp. 27–37, 2003.
- [218] R. C. Roberts, L. McNabb, N. AlHarbi, and R. S. Laramee, "Spectrum: A C++ Header Library for Colour Map Management," in *Computer Graphics and Visual Computing (CGVC)*, G. K. L. Tam and F. Vidal, Eds. The Eurographics Association, 2018.
- [219] E. Loper and S. Bird, "Nltk: the natural language toolkit," *arXiv preprint cs/0205028*, 2002.

- [220] T. Cheesman, K. Flanagan, and S. Thiel, “Translation Array Prototype 1: Project Overview,” The Arts and Humanities Research Council (AHRC), Tech. Rep. 1, 2012. [Online]. Available: www.delightedbeauty.org/vvv
- [221] M. Konrad, “A lemmatizer for german language text,” <https://github.com/WZBSocialScienceCenter/germalemma>, 2017.
- [222] H. Schmid, *Improvements in Part-of-Speech Tagging with an Application to German*. Dordrecht: Springer Netherlands, 1999, pp. 13–25. [Online]. Available: https://doi.org/10.1007/978-94-017-2390-9_2
- [223] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Information Processing & Management*, vol. 24, no. 5, pp. 513–523, 1988.
- [224] P. Koehn *et al.*, “Europarl: A parallel corpus for statistical machine translation,” in *Machine Translation Summit X*, vol. 5, 2005, pp. 79–86.
- [225] N. Boyles, “Closing in on close reading,” *On Developing Readers: Readings from Educational Leadership, EL Essentials*, pp. 89–99, 2012.
- [226] C. han Jong, P. Rajkumar, B. Siddiquie, T. Clement, C. Plaisant, and B. Shneiderman, “Interactive exploration of versions across multiple documents,” 2008.
- [227] C. Wieland, *Othello, der Mohr von Venedig*. Delphine Lettau, 1766.
- [228] C. Leonard, *Othello!* Typescript. Shakespeare Company Berlin, 2010.
- [229] F. Moretti, *Distant reading*. Verso Books, 2013.
- [230] H. Mehta, A. Bradley, M. Hancock, and C. Collins, “Metatation: Annotation as implicit interaction to bridge close and distant reading,” *ACM Trans. Computer-Human Interaction*, vol. 24, no. 5, 2017.
- [231] B. Jurish, “Finite-state canonicalization techniques for historical german,” Ph.D. dissertation, University of Potsdam, 2011.
- [232] Berlin-Brandenburgische Akademie der Wissenschaften, “Deutsches textarchiv,” <http://www.deutschestextarchiv.de>, 2014, accessed: 2018-02-10.
- [233] Rui Xu and D. Wunsch, “Survey of clustering algorithms,” *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.
- [234] M. Brehmer and T. Munzner, “A multi-level typology of abstract visualization tasks,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2376–2385, 2013.

-
- [235] A. Inselberg and B. Dimsdale, "Parallel coordinates: a tool for visualizing multi-dimensional geometry," in *The First IEEE Conference on Visualization: Visualization '90*, 1990, pp. 361–378.
- [236] M. Novotny and H. Hauser, "Outlier-preserving focus+context visualization in parallel coordinates," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 893–900, 2006.
- [237] J. Johansson and C. Forsell, "Evaluation of parallel coordinates: Overview, categorization and guidelines for future research," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 579–588, 2016.
- [238] V. Gold, C. Rohrdantz, and M. El-Assady, "Exploratory Text Analysis using Lexical Episode Plots," in *Eurographics Conference on Visualization (EuroVis) - Short Papers*, E. Bertini, J. Kennedy, and E. Puppo, Eds., 2015.
- [239] G. G. Robertson and J. D. Mackinlay, "The document lens," in *Proc. of the 6th Annual ACM Symposium on User Interface Software and Technology*, 1993, p. 101–108.
- [240] M. Chang and C. Collins, "Exploring entities in text with descriptive non-photorealistic rendering," in *IEEE Pacific Visualization Symposium (PacificVis)*, 2013, pp. 9–16.
- [241] R. Flatter, *Othello der Mohr von Venedig. Sonderabdruck für Bühnenzwecke*. Munich: Theater-Verlag Desch, 1952.
- [242] T. v. Zeynek, *Shakespeare: Othello Der Mohr von Venedig*. Munich: Ahn und Simrock Bühnen und Musikverlag, 1948.
- [243] M. Wachsmann, *William Shakespeare, Die Tragödie von Othello, dem Mohr von Venedig*. Berlin: Gustav Kiepenheuer Bühnenvertriebs-GmbH, 2005.
- [244] A. C. Telea, *Data visualization: principles and practice*. AK Peters/CRC Press, 2007.
- [245] M. J. Wolff, *Shakespeares Werke übertragen nach Schlegel-Tieck*. Berlin: Volksverband der Bücherfreunde, Wegweiser-Verlag, 1926.
- [246] K. Brunner, *Othello, der Mohr von Venedig*. Linz: Österreichischer Verlag für Belletristik und Wissenschaft, 1947.
- [247] F. Schiller and J. H. Voss, *Othello*. Stuttgart: Hermann Böhlhaus Nachfolger / J. B. Metzler, 1805.
- [248] J. W. O. Benda, *Othello, der Mohr von Venedig*. Hanover: Georg Joachim Göschen, 1826.
- [249] E. Ortlepp, *Othello der Mohr von Venedig*. W. Shakspeare's dramatische Werke, übersetzt von Ernst Ortlepp, 1839.

- [250] R. A. Schröder, *Shakespeare/deutsch*. Berlin, Frankfurt am Main: Suhrkamp, 1962.
- [251] H. Rothe, *Der Elisabethanische Shakespeare*. Baden-Baden: Holle, 1956, vol. 4.
- [252] H. Motschach, *Othello*. Drei Masken Verlag, 1992.
- [253] R. Schaller, *Shakespeares Werke*. Berlin: Rütten & Loening, 1959, vol. 4.
- [254] F. Gundolf, *Shakespeare in deutscher Sprache*. Berlin: Bondi, 1909.
- [255] M. El-Assady, V. Gold, M. John, T. Ertl, and D. A. Keim, "Visual text analytics in context of digital humanities," in *1st IEEE VIS Workshop on Visualization for the Digital Humanities as part of the IEEE VIS 2016*, 2016. [Online]. Available: <https://scibib.dbvis.de/publications/view/686>
- [256] S. Silvia, R. Etemadpour, J. Abbas, S. Huskey, and C. Weaver, "When the tech kids are running too fast: Data visualisation through the lens of art history research," in *Proceedings of the Workshop on Visualization for the Digital Humanities*. Berlin, Germany: IEEE, October 2018.
- [257] M. Deegan and W. McCarty, *Collaborative Research in the Digital Humanities: A Volume in Honour of Harold Short, on the Occasion of His 65th Birthday and His Retirement, September 2010*. USA: Ashgate Publishing Company, 2012.
- [258] M. K. Gold, *Debates in the digital humanities*. University of Minnesota Press, 2012.
- [259] T. Munzner, "A nested model for visualization design and validation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 921–928, 2009.
- [260] R. Kath, G. S. Schaal, and S. Dumm, "New visual hermeneutics," *Zeitschrift für germanistische Linguistik*, vol. 43, no. 1, pp. 27–51, 2015.
- [261] S. Simon, S. Mittelstädt, D. A. Keim, and M. Sedlmair, "Bridging the Gap of Domain and Visualization Experts with a Liaison," in *Eurographics Conference on Visualization (EuroVis) - Short Papers*, E. Bertini, J. Kennedy, and E. Puppo, Eds. The Eurographics Association, 2015.
- [262] R. Roberts., R. Laramee., P. Brookes., G. A. Smith., T. D’Cruze., and M. J. Roach., "A tale of two visions - exploring the dichotomy of interest between academia and industry in visualisation," in *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 2: IVAPP, INSTICC*. SciTePress, 2018, pp. 319–326.
- [263] V. Schetinger, K. Raminger, V. Filipov, N. Soursos, S. Zapke, and S. Miksch, "Bridging the gap between visual analytics and digital humanities: Beyond the data-users-tasks design triangle," 2020, eingeladen; Vortrag: 4th Workshop on Visualization for the Digital Humanities, Vancouver, Canada; 2020-10-20.

- [264] J. J. van Wijk, “Views on visualization,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 421–432, 2006.
- [265] S. Jänicke, A. Geßner, G. Franzini, M. Terras, S. Mahony, and G. Scheuermann, “Traviz: A visualization for variant graphs,” *Digital Scholarship in the Humanities*, vol. 30, no. suppl_1, pp. 83–99, 10 2015. [Online]. Available: <https://doi.org/10.1093/lc/fqv049>
- [266] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale, “Empirical studies in information visualization: Seven scenarios,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 9, pp. 1520–1536, 2012.
- [267] C. Plaisant, “The challenge of information visualization evaluation,” in *Proceedings of the Working Conference on Advanced Visual Interfaces*, ser. AVI '04. New York, NY, USA: Association for Computing Machinery, 2004, p. 109–116. [Online]. Available: <https://doi.org/10.1145/989863.989880>
- [268] M. Tory and T. Moller, “Human factors in visualization research,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 1, pp. 72–84, 2004.
- [269] S. Carpendale, *Evaluating Information Visualizations*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 19–45. [Online]. Available: https://doi.org/10.1007/978-3-540-70956-5_2
- [270] R. S. Laramée, “How to write a visualization research paper: A starting point,” *Computer Graphics Forum*, vol. 29, no. 8, pp. 2363–2371, 2010. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2010.01748.x>
- [271] R. Amar, J. Eagan, and J. Stasko, “Low-level components of analytic activity in information visualization,” in *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, 2005, pp. 111–117.
- [272] B. R. T. Röhle, *Digital Methods: Five Challenges*. London: Palgrave Macmillan UK, 2012, pp. 67–84. [Online]. Available: https://doi.org/10.1057/9780230371934_4
- [273] B. Elliott, “Anything is possible: Managing feature creep in an innovation rich environment,” in *2007 IEEE International Engineering Management Conference*. IEEE, 2007, pp. 304–307.