

Bridging the Gap Between Explanation and Exploration

Combining Text and Visualization for Dissemination of Analysis Results

DISSERTATION ZUR ERLANGUNG DES DOKTORGRADES
DR. RER. NAT.
DER FAKULTÄT FÜR WIRTSCHAFTSWISSENSCHAFTEN
DER UNIVERSITÄT DUISBURG-ESSEN

VORGELEGT
VON

SHAHID LATIF
AUS
TOBA TEK SINGH, PAKISTAN

BETREUER: PROF. DR. FABIAN BECK

INSTITUTE OF COMPUTER SCIENCE AND BUSINESS INFORMATION SYSTEMS
UNIVERSITY OF DUISBURG-ESSEN

ESSEN, APRIL 2022

© SHAHID LATIF
ALL RIGHTS RESERVED, 2022

Examiners

Prof. Dr. Fabian Beck
Prof. Dr. Silvia Miksch
Prof. Dr. Torsten Brinda

Date of Oral Exam

July 18, 2022

UNIVERSITÄT
DUISBURG
ESSEN

Offen im Denken

DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT
DUISBURG
ESSEN

Offen im Denken

ub | universitäts
bibliothek

Diese Dissertation wird via DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt und liegt auch als Print-Version vor.

DOI: 10.17185/duepublico/77124

URN: urn:nbn:de:hbz:465-20221124-141957-0

Alle Rechte vorbehalten.

Abstract

Visualization systems offer data exploration and are often designed to help domain analysts understand trends, outliers, and patterns in the data. These systems have little to no support for communicating knowledge or insights that are derived through the analysis. Data-driven storytelling, on the other hand, employs specially designed explanatory visualizations or combines a textual narrative alongside a visualization to communicate analysis results to a wider audience, but mostly has no support for exploration. Targeting a sweet spot between explanation and exploration, this doctoral thesis envisions a data representation that integrates explorable visualizations with an automatically generated textual narrative for the dissemination of analysis results to a broad audience.

Grounded in an empirical study on a set of already published data-driven stories, the thesis first describes the interplay of text and visualizations with a focus to derive effective means of achieving a coherent integration between the two media. The thesis then introduces a generic approach for generating an integrated visual and textual representation of data, followed by its instantiations to many diverse application domains and datasets including bivariate geographic data, bibliographic data, knowledge graphs, and source code quality data. Finally, considering the technical challenges and lack of authoring support for the construction of such data representation, the thesis contributes a novel and easy-to-use authoring tool that combines automation with a graphical user interface to establish linking between text and visualizations.

The proposed data representation can still be regarded as a visual analytics solution where text is considered as an active part of the system, just like any other visual element. The text summarizes core findings and hints at notable analysis insights, while intriguing users to explore and even verify the insights. Likewise, while exploring, the text provides additional information on analysis methods and domain-specific terminology, and links back to the explanation.

Contents

1	INTRODUCTION	1
1.1	Research Objectives	4
1.2	Summary of Contributions and Thesis Outline	5
2	BACKGROUND AND PRIOR RESEARCH	7
2.1	Data-driven Storytelling	7
2.2	Text and Visualization: Two Faces of the Same Data	8
2.3	Integration of Text and Visualization	10
2.3.1	Word-sized Graphics	10
2.3.2	Interactive Visualization–Text Linking	11
2.4	Automatic Text Generation for Data and Visualization	13
2.4.1	Text Generation for Statistical and Other Forms of Data	13
2.4.2	Text Generation for Data Visualization	15
2.5	Interactive Data Documents	15
2.5.1	Authoring Support for Data-driven Storytelling	16
2.5.2	Creating Exploratory Visualization	16
2.5.3	Establishing Linking of Text and Visualization	17
3	UNDERSTANDING VISUALIZATION–TEXT INTERPLAY	19
3.1	Research Questions	19
3.2	Methodology	21
3.2.1	Data Collection	21
3.2.2	Qualitative Analysis	23
3.3	Results: Insights and Visual Communication (RQ 1)	26
3.3.1	Analysis Insights and Context (RQ 1.1)	26
3.3.2	Visual Communication (RQ 1.2)	29
3.4	Results: Interplay of Text and Visualization (RQ 2)	31
3.4.1	Linking the Two Media (RQ 2.1)	31
3.4.2	Embedding of Visualizations into the Narration (RQ 2.2)	33
3.5	Results: Implicit Referencing (RQ 3)	34
3.5.1	Types of Implicit References (RQ 3.1)	35
3.5.2	Relation of References to Data and Visualization (RQ 3.2)	36

3.6	Study Limitations	37
3.7	Implications	38
3.7.1	Integration of Visualization and Text	38
3.7.2	Automatic Generation of Interactive Data Documents	39
4	GENERATION OF AN INTEGRATED TEXTUAL AND VISUAL REPRESENTATION	41
4.1	Automatic Generation Process	41
4.1.1	Content Determination and Prioritization	42
4.1.2	Text Generation	43
4.1.3	Document Integration	43
4.2	VIS Author Profiles: Bibliographic Data	45
4.2.1	Analyzing Publication Records	45
4.2.2	Existing Systems and Author Visualizations	50
4.2.3	Generation Pipeline	53
4.2.4	The System	54
4.2.5	Author-Profile Gallery for Different Personas	61
4.3	Interactive Map Reports: Bivariate Geographical Data	62
4.3.1	Scope, Data, and Content Determination	64
4.3.2	Analyzing Bivariate Geographic Data	65
4.3.3	Adaptable Text Generation	68
4.3.4	The System	69
4.3.5	Application Examples	72
4.4	Discussion	75
4.4.1	Quality and Accuracy of Information	75
4.4.2	Usefulness and Usability	76
4.4.3	Generalizability	76
4.4.4	Applicability and Extension	77
5	EXPLORATION – EXPLORATION AND EXPLANATION	79
5.1	Explorantion in Interactive Data Documents	80
5.2	Explorantion: A Concrete Application Example	83
5.2.1	Code Quality Data and Analysis	83
5.2.2	The System	85
5.2.3	Usage Scenario – A User’s Perspective	90
5.2.4	Key Takeaways	91
5.3	Explorantion and Chatbots	92
5.3.1	Motivation and The Problem	93
5.3.2	VisKonnnect	93
5.3.3	Sample User Queries and Responses	97
5.3.4	Strengths, Limitations, and Challenges	99
5.4	Explorantion in Virtual Reality Visualizations	101

5.4.1	Talking Realities	101
5.4.2	Application Examples	105
5.4.3	Limitations and Future Opportunities	110
5.5	Conclusion	111
6	AUTHORING INTERACTIVE DATA DOCUMENTS	113
6.1	Design Considerations	114
6.2	The User Interface and Usage Scenarios	114
6.2.1	Authoring	115
6.2.2	Reading	116
6.3	The Kori System	116
6.3.1	Data, Charts, and Visualization–Text References	116
6.3.2	Reference Detection	117
6.3.3	Reference Construction	120
6.4	Algorithmic Evaluation: Reference Detection	121
6.4.1	Dataset Curation	121
6.4.2	Results	122
6.5	User Evaluation: The Kori System	123
6.5.1	Participants	123
6.5.2	Procedure and Tasks	123
6.5.3	Results	125
6.6	Limitations, Challenges, and Future Opportunities	126
7	CONCLUSION	129
7.1	Limitations and Future Directions	130
7.1.1	Detailed User Evaluation	130
7.1.2	Advance Natural Language Text Generation	130
7.1.3	Closing the Loop from Explanation to Exploration in Authoring Tools	131
7.2	Outlook	131
	REFERENCES	146

Listing of Figures

1.1	Examples of visual analytic systems: Lex et al.’s approach ¹⁰⁴ visualizes set intersections and Malqui et al.’s system ¹¹⁰ analyzes passing sequences in a soccer game. Although both systems visualize datasets (IMDB, soccer match) that are relevant and interesting for common users (moviegoers and soccer fans) yet they are complex and lack explanatory aspect.	2
1.2	Excerpts from the data-driven stories published in various digital news media. They use a combination of text and visualizations to communicate data-driven insights.	3
2.1	An example demonstrating split-attention effect taken from the work of Sweller et al. ¹⁶⁸ (Left) Presentation with split-attention effect. (Right) Integrated presentation with no split-attention effect.	9
2.2	(Left) Document without word-sized graphics where the reader has to switch attention between graphics and text. (Right) Document using integrated word-sized graphics in line with the text and in tables to reduce a split-attention effect and save space for additional content (here, a table with further information). Colors ■■ indicate the textual and visual content that relate to each other.	11
2.3	Two examples of interactive referencing in literature: (Left) Kim et al. ⁸⁰ use highlights related text in tables while reading the corresponding text. (Right) Beck and Weiskopf ¹⁴ propose the idea of visually highlighting relevant parts of the visualization while interacting with associated text elements.	13
2.4	Interactive data document describing the biography of a fictional character, Jon Snow, from the TV series Game of Thrones. Two visualization–text linking interactions: (A) The result of hovering over an interactive text and (B) a node in the node-link diagram.	14
2.5	The envisioned data-driven documents lie at the sweet spot between visual analytics systems and data-driven stories.	16
3.1	Sources of stories in the sample data collection for Study I. Almost 50% (14/22) of the stories were gathered from three well-known sources: NYT, 538, and BBC.	22
3.2	Overview of the sample data collection for Study II. We augmented existing articles from Kong et al. ⁸⁷ with additional venues and chart types.	22

3.3	Frequencies of codes for 22 stories on sentence- and visualization-level, structured by code categories and subcategories. Gray-blue background encodes the frequency of sentences, yellow background the frequency of visualizations. Multiple codes can be assigned to a single sentence/visualization, hence, per story, the total count of sentences and visualizations does not correspond to the total number of assigned codes.	25
3.4	Flow and structure of stories. Each story is represented by a series of rectangles encoding the type of sentences (<i>heading</i> , <i>data-driven</i> , <i>embedding</i> , and <i>visualization-textlinking</i>) and <i>visualizations</i> . The width of each rectangle encodes the size of a sentence (word count) or a visualization (estimated word count equivalent). White gaps indicate paragraph spacing. Rectangles are vertically (equally) divided in case a sentence has multiple codes assigned to it. The thumbnails on the right show 17 visualizations from our sample collection.	33
3.5	Excerpt of an article from Pew Research ⁸⁵ . Demonstration of text-chart reference grouping: The colored texts are minimal references, while A and B show how they can be grouped from bottom to top.	34
3.6	Overview of various kinds of implicit references in our sample collection. We observed point, multi-point, and interval selections in the text-chart references (e.g., a phrase describing one or more visual marks or a numerical range). They are often grouped together to generate aggregate selections, while also forming a hierarchical reference tree: <i>minimal</i> phrase \rightarrow r^{st} level parent phrase \rightarrow 2^{nd} level parent phrase \rightarrow <i>root</i> sentence; see Figure 3.5 for an example.	35
4.1	Profile of author <i>Ben Shneiderman</i> . The text consists of three sections describing general information, research areas, and collaboration relationships. The visualization below provides information on joint work with co-authors on a timeline. The sidebar shows details on demand, whereas the top-right bar chart displays the temporal distribution of publications. Badges at the top summarize achievements. The cut-outs on the right are two different versions of the sidebar (list of collaboration groups and similar authors).	46
4.2	Scope and data pre-processing: VIS Author Profiles is restricted to generate profiles of authors available in the <i>VisPub</i> dataset. However, it retrieves other publications of these authors from <i>DBLP</i> data.	48
4.3	Decision graph explaining the flow of text generation for the first paragraph of a profile. Rectangular nodes represent <i>text</i> vertices, whereas nodes with rounded corners are <i>decision</i> vertices. The traversal of any path from <i>start</i> to <i>end</i> node produces a meaningful paragraph.	55
4.4	Excerpt (general information) from Benjamin Bach’s profile.	56
4.5	Excerpt (research topics) from Daniel Weiskopf’s profile.	57
4.6	Excerpt (co-author network) from Katherine E. Isaacs’s profile including the <i>co-author publication timeline</i> . Gray bars indicate the number of publications per year of the co-author, while red bars show joint publications with Isaacs.	58

4.7	Author profiles of several researchers falling into different author personas. For each persona, VIS Author Profiles has successfully generated meaningful profiles. (These example profiles span across two figures; please see the next figure for two more author profiles).	60
4.8	(Continuation of Figure 4.7) Two author profiles for the <i>senior researcher</i> persona.	61
4.9	A map report describing loss of lives due to storms in the USA during 2017. The map visualization uses two different encodings to visualize a <i>focus</i> and a <i>context</i> variable. The narrative (right column) provides an overview of the data analysis. Graphics in the text help establish linking between the two representations. Users can get additional details on a selected region or on a comparison of two selected regions (dashed rectangles).	63
4.10	Bivariate map visualizations used in the explorative study. (Left, P1) Deaths caused by storms in various states of the USA. (Right, P2) Average life expectancy and health expenditures across Europe.	64
4.11	Box plots showing the distribution of <i>deaths</i> caused by <i>storms</i> in the USA during 2017. The dataset contains univariate outliers in both variables.	66
4.12	Bagplot of <i>deaths</i> caused by <i>storms</i> in the USA during 2017. The <i>bag</i> (blue) contains almost 50% of the data points, and the <i>loop</i> (light blue) includes points outside the bag but inside the fence. Bivariate outliers are marked as red dots.	67
4.13	Decision graph that shows the text generation process. Round-rectangular decision nodes control the path, while rectangular text nodes add a text fragment when visited. The green path marks the narrative generation for the example in Figure 4.9.	69
4.14	An interactive map report describing average life expectancy and health expenditure across Europe in 2018.	73
4.15	An interactive map report showing the possible relationship between adolescent birth rates and the percentage of Internet users across countries of the world in 2015.	73
4.16	An interactive map report describing the percentage of obese people and alcohol consumption in the world during 2010.	74
5.1	Abstract representation of the interaction model. We have bi-directional interactions among text, embedded visualizations (emvis), and overview visualizations (vis).	82
5.2	Explorative code quality document for <i>Lucene 2.0</i> . (A) Textual overview in terms of quality attributes, code smells, and bugs, which includes embedded visualizations. (B) Overview visualizations: parallel coordinates plot and scatterplot. (C) The source code of a class is provided in the details view. (D) Description of a quality attribute alternatively presented in the details view.	85
5.3	Methodological explanation of classifying the complexity of a source code project in terms of low, regular, or good.	86
5.4	Low quality in <i>HTML_ElementImpl</i> found through active exploration of the embedded detail visualizations of coupling, cohesion, and inheritance.	87

5.5	Various instances of <i>vis-text</i> interactions. A persistent highlighting (click on <i>TraverseSchema</i>) marks the related elements with bold font in the text (<i>text-text</i>), a black line in the parallel coordinates, a black dot in the scatterplot (<i>text-vis</i>), and black background in embedded detail visualizations (<i>text-emvis</i>). Similarly, a non-persistent highlighting (hover on <i>UTF8Reader</i>) marks the corresponding elements in yellow. The details panel shows the class-specific description and source code of the persistently selected class, <i>TraverseSchema</i>	88
5.6	A selection of the classes in <i>Xerces 1.2</i> that have a <i>Functional Decomposition</i> smell; they are highlighted in the parallel coordinates plot and the scatterplot. The caption of the visualizations adapts to describe the selection.	89
5.7	VisKconnect interface visualizing the result of a query on soccer players: (A) An event timeline shows individual (uni-colored rectangles) and shared (multi-colored rectangles) life events of all historical figures in a user query. (B) An event map provides a geographic perspective of those events. (C) A relationship graph provides an overview of all shared events. (D) Clicking an event brings up a related article from Wikipedia. (E) A chat interface enables typing in a query and generates a short textual answer.	94
5.8	An excerpt of EventKG about the relationship between Mats Hummels and Miroslav Klose; <i>sub class*</i> denotes transitivity.	95
5.9	Chat responses based on templates (left) and GPT-3 (right).	97
5.10	Cut-outs of two responses generated by VisKconnect. (1) Event timeline for Example 2. (2) Relationship graph for Example 3.	98
5.11	Excerpt of a question about Einstein and Schrödinger (Example 4). The template-based answer identifies the Solvay conference as a shared event where the two scientists were photographed together.	98
5.12	Abstract representation of Talking Realities—a concept for producing data-driven narratives in virtual reality. (A) The interplay of three aspects (I) <i>Immersive Visualization</i> , (II) <i>Narrative Generation</i> , and (III) <i>Exploration</i> . (B) Exploration offers varying levels of guidance: <i>Guided Tours</i> walk the users through a predefined sequence of events, <i>Guided Exploration</i> provides hints at various possible perspectives to explore, and <i>Free Exploration</i> enables the users to freely examine the dataset.	102
5.13	The visualization is animated in synchronization with the audio narrative.	104
5.14	Graphical interface of the prototype and various animations. (A) The visualization displays the aggregated inter-continental flights for one regular day. The color gradient (red:departure to green:arrival) denotes the direction of flights. Animations showing (B) the longest flight from an airport, (C) large airports of a country, and (D) most flights to any other airport. The right (gray) box discloses the audio transcription that are played when users see corresponding animations.	107
5.15	Virtual menus providing guidance on the possible aspects of exploration on selecting an airport (A) and a country (B). Audio controls (C) can be accessed at any time to replay, skip the current playback, and to adjust the volume.	108

5.16	A mock-up illustrating the application of Talking Realities to <i>mtcars</i> dataset. (A) Scatterplot visualizes horsepower, miles per gallon, acceleration, and cylinders for 406 different car models (1970–1982). The right (gray) box shows the transcription of data-driven audio that summarizes insights related to (A,C) correlations among all visualized variables (B) unique models (outliers) with respect to number of cylinders, (C) details on demand for a selected car model, and (D) changes in the values of different properties over the years. The first text block refers to both sub-figures A and C.	109
6.1	The Kori system consists of a chart gallery (left), edit area (middle), and a link setting panel (right). Kori automatically suggests potential references (dotted gray underline) as a user types. Besides, it supports manual creation of links through simple interactions (4–6). The steps 1–10 describe a usage scenario to create an interactive story.	115
6.2	Salient features of Kori. (A) It uses natural language processing to suggest potential references between charts and the text while a user types. (B) Users can construct references by directly manipulating ✍ the visual marks on the chart. (C) It is possible to manually trigger suggestions. (D) In direct manipulation ✍ mode, users can easily expand their current selection to multiple visual marks of the same type. The encircled numbers mark the sequence of interactions for each activity.	117
6.3	Four stages of our automatic reference suggestion pipeline. It accepts text and Vega-Lite specifications as input and begins with extracting features of a chart. These features are then matched against user text to find point references. The third step identifies numerical intervals in the user text. Finally, related references are grouped together to form higher level references.	118
6.4	Dependency parsing for two sample sentences. (top) Success case: “ <i>minimum temperature</i> ” is successfully grouped with “ <i>between 6 and 10</i> ”. (bottom) Failure case: “ <i>price</i> ” is wrongly grouped with “ <i>between 2006 and 2008</i> ” as it is closer to “ <i>price</i> ” than “ <i>over \$400.</i> ”	120
6.5	Quantitative evaluation of the reference detection approach. Precision, recall, and F_1 at various maximum n -gram sizes for the validation dataset (55 text-chart pairs). . .	122
6.6	Copy-edited excerpts of stories created by participants. The symbol 🗨 marks the reference suggestions. (A) P3 relied on reference construction using direct manipulation ✍ . (B) In contrast, P8 got 9 suggestions (1 wrong). She manually constructed 2 references: “ <i>1930 claimed about 3 million lives.</i> ” and combined two suggestions (“ <i>mass movement</i> ” and “ <i>extreme temperatures</i> ”) into a single reference. (C) P5 used the first reference “ <i>very similar</i> ” to verify a fact he was describing. (D) P6 got only two suggestions and created most references using the filtering 🗨 interface.	124

Listing of Tables

4.1	Assessment of fulfillment of information needs in author profile pages of existing digital library systems; degree of fulfillment: <i>no</i> ○ ○ ○, <i>partly</i> ● ○ ○, <i>largely</i> ● ● ○, and <i>yes</i> ● ● ●.	50
4.2	User-defined parameters for configuring the map reports.	70
4.3	Parameter configuration for shown examples	72
5.1	Eleven class-level software metrics (name and acronym) used for code quality analysis, grouped by quality attributes.	84
5.2	Various types of analysis that can be performed on a dataset.	104

TO MY BRAVE AND COURAGEOUS MOTHER, TASNEEM K. LATIF, AND IN LOVING MEMORY OF
MY FATHER, ABDUL LATIF.

Acknowledgments

First and foremost, I wish to express my gratitude to my advisor, Fabian Beck, for his immense support and guidance at every step of the way, for countless brainstorming sessions and research discussions, for always appreciating my work and providing constructive critique, for his constant availability, and for teaching me how to be a good researcher. I really admire his pixel-perfect attitude toward designing research prototypes. He always encouraged me to grow as an independent researcher by conducting special training sessions on various aspects of research early on and supporting my research visits in Germany and abroad. A special thanks to Silvia Miksch for accepting the role of secondary examiner on short notice.

I wish to thank all my co-authors who supported my research in one way or the other. Thanks to Daniel Weiskopf for inviting me to VISUS, University of Stuttgart, and providing the opportunity to work on an interesting project and instilling the importance of a broader perspective of research into my mind. I am grateful to Haris Mumtaz who was an excellent collaborator at VISUS and shared my enthusiasm to publish our research at a high-quality venue even if it meant pushing our boundaries. I would also like to thank Nam Wook Kim for being a wonderful mentor during my stay at Boston College as visiting scholar. He inspired me with his passion for research on building user interfaces, excellent technical skills, and a keen eye for visual design and aesthetics. Finally, I would like to extend my gratitude to Siming Chen (Fraunhofer Institute), Yoon Kim (MIT), Simon Gottschalk (University of Hanover), Shivam Agarwal, and Elena Demidova (University of Bonn) for being supportive co-authors.

I thank my colleagues Hagen Tarnier, Cedric Krause, Sebastian Surminski, and Michael Rodler who were always supportive and willing to help especially regarding acclimatizing to German work culture and in matters that required advanced German language knowledge. Special thanks to Shivam Agarwal for long philosophical discussions on diverse topics, for listening to my half-baked ideas, for providing feedback on my ongoing projects, and for always asking thought-provoking questions.

I would like to acknowledge the support of undergraduate and graduate students who supported my research by helping with technical implementation, data pre-processing, or other similar tasks. Also, thanks to anonymous reviewers and participants of the user studies and evaluations.

Finally, the support of my family and friends was always a source of strength. Without their backing, it has not been possible. A huge thanks to them.

Preface

This work is a PhD dissertation submitted at the Institute of Computer Science and Business Information Systems, University of Duisburg-Essen. The research has been supervised by Prof. Dr. Fabian Beck. As the secondary examiner, I suggest Prof. Dr. Silvia Miksch. The research has been in parts supported by the Deutsche Forschungsgemeinschaft (DFG) [grant number: 424960846].

Prior Publications

Parts of this work—including ideas, text, results, and figures—have already been published in scientific journals including Transactions on Visualizations and Computer Graphics (TVCG)^{95,100}, Visual Informatics (VI)⁹⁴, Computer Graphics and Applications (CGA)⁹⁹, Computer Graphics Forum (CGF)⁹⁶, and Computing in Science and Engineering (CiSE).⁹³ The work has been presented at various scientific conferences including IEEE VIS (2018, 2019, 2021), EuroVis (2018, 2019, 2021), and PacificVis (2019).

In particular, this thesis is based on and connected to the following peer-reviewed, and already, published research articles:

- Shahid Latif and Fabian Beck, “VIS Author Profiles: Interactive Descriptions of Publication Records Combining Text and Visualization,” in IEEE Transactions on Visualization and Computer Graphics, vol. 25, no. 1, pp. 152-161, 2019, doi: <https://doi.org/10.1109/TVCG.2018.2865022>. – **Chapter 4**
- Shahid Latif, Zheng Zhou, Yoon Kim, Fabian Beck, and Nam Wook Kim, “Kori: Interactive Synthesis of Text and Charts in Data Documents,” in IEEE Transactions on Visualization and Computer Graphics, vol. 28, no. 1, 2022, doi: <https://doi.org/10.1109/TVCG.2021.3114802>. – **Chapter 3, 6**
- Shahid Latif, Siming Chen, and Fabian Beck, “A Deeper Understanding of Visualization-Text Interplay,” in Geographic Data-driven Stories. Computer Graphics Forum, vol. 40, pp. 311-322, 2021, doi: <https://doi.org/10.1111/cgf.14309>. – **Chapter 3**
- Shahid Latif, Hagen Tärner, and Fabian Beck, “Talking Realities: Audio Guides in Virtual Reality Visualizations,” in IEEE Computer Graphics and Applications, doi: <https://doi.org/10.1109/MCG.2021.3058129>. – **Chapter 5**

- Haris Mumtaz, Shahid Latif, Fabian Beck, and D. Weiskopf, “Explorative Code Quality Documents,” in IEEE Transactions on Visualization and Computer Graphics, vol. 26, no. 1, pp. 1129-1139, 2020, doi: <https://doi.org/10.1109/TVCG.2019.2934669>. – **Chapter 5**
- Shahid Latif and Fabian Beck, “Interactive Map Reports Summarizing Bivariate Geographic Data,” in Visual Informatics, vol. 3, no. 1, pp. 27-37, 2019, doi: <https://doi.org/10.1016/j.visinf.2019.03.004>. – **Chapter 4**
- Shahid Latif, Shivam Agarwal, Simon Gottschalk, Carina Chrosch, Felix Feit, Johannes Jahn, Tobias Braun, Yannick Christian Tchenko, Elena Demidova, and Fabian Beck, “Visually Connecting Historical Figures Through Event Knowledge Graphs,” IEEE Visualization Conference (VIS), 2021, pp. 156-160, doi: <https://doi.org/10.1109/VIS49827.2021.9623313>. – **Chapter 5**

My other published work^{97,98,93} is not in the main scope of this thesis. Therefore, it has not been discussed in the larger context but discussed as background and related work.

Essen, April 4, 2022

—Shahid Latif

“Do our reading environments encourage active reading? Or do they utterly oppose it? A typical reading tool, such as a book or website, displays the author’s argument, and nothing else. The reader’s line of thought remains internal and invisible, vague and speculative. We form questions, but can’t answer them. We consider alternatives, but can’t explore them. We question assumptions, but can’t verify them. And so, in the end, we blindly trust, or blindly don’t, and we miss the deep understanding that comes from dialogue and exploration.”

—Bret Victor

1

Introduction

Data is ubiquitous and the quantity of data produced in our society—in almost all fields of life—is increasing at a staggering pace. Data holds an enormous amount of information that can be discovered for the betterment of society. Researchers and analysts apply data analysis techniques to find valuable information and insights from raw data. Decision makers, in turn, rely on these insights to take better decisions to tackle problems at hand, for instance, to fight an infectious disease, improve public transportation, or even counter issues like global warming and climate change. Although the general public is a direct beneficiary of these informed decisions, in many situations (e.g., to stop the spread of an infectious disease) the public can play their part—and an effective role—if they apprehend and use the derived analysis insights. However, the results of data analysis are often difficult to comprehend, especially when they discuss complex data. Therefore, conscious efforts must be made to communicate complex data analysis results to wider audiences in order to reap the benefits. For instance, self-explanatory illustrations of mathematical concepts (like exponential curves and logarithmic scales) and the impact of movement and lockdown on the spread of COVID-19 have helped spread awareness about the seriousness of the pandemic. Visualization designers and journalists created engaging, insightful, and easy-to-understand stories—including explanatory visualizations—for this purpose. This is just one instance of many possible applications which warrants that the capability to comprehend and disseminate data to a broad audience is becoming inevitable.

Data visualization leverages visual perception and graphical representation of data to provide effective tools for better understanding meaningful insights and notable patterns. However, conventional visualizations focus on rapid analysis and exploration of data and are designed to support domain experts in finding patterns, detecting outliers, formulating hypotheses, and confirming them. They provide great exploration capability yet lack an explanatory aspect that is critical to reach an audience beyond experts. For instance, Figure 1.1 shows two visual analytics systems. Both examples (character-

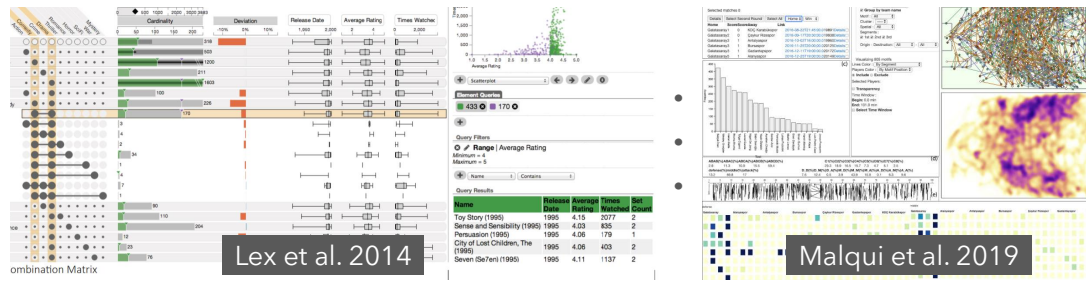


Figure 1.1: Examples of visual analytic systems: Lex et al.’s approach¹⁰⁴ visualizes set intersections and Malqui et al.’s system¹¹⁰ analyzes passing sequences in a soccer game. Although both systems visualize datasets (IMDB, soccer match) that are relevant and interesting for common users (moviegoers and soccer fans) yet they are complex and lack explanatory aspect.

istics of movies across different genres^{104*} and ball passing strategies in a soccer game¹¹⁰) are relevant for non-expert users (e.g., movie enthusiasts, soccer fans). Yet, comparatively complex visual encoding, domain terminology, and many interconnected views render them overwhelming for common users. To increase the legibility of such intricate visualization systems beyond expert users, it is crucial to emphasize the communication aspect of analysis results, which is lacking in most of the visual analytics systems.

Nowadays, visualizations—specifically designed for communication purposes—are proving to be a great source of communicating data-driven facts and insights to the general public. This concept is often referred to as *narrative visualization*¹⁵¹ or *data-driven storytelling*.¹⁴³ It is the ability to turn data into intuitive and self-explanatory stories through the use of explanatory data visualizations. Several impactful news media outlets (New York Times, Washington Post, FiveThirtyEight, Financial Times, The Guardian, and others) produce such stories, embedding a textual narrative with the visual representation of data and publish regularly on a variety of topics such as politics, sports, economics, and culture. While visualizations in these stories show the data, the narrative explicitly explains insights and context, thereby making the whole representation self-explanatory. Figure 1.2 shows three excerpts of data-driven digital stories (complete stories: A[†], B[‡], and C[§]) comprising a narrative and visual representation of data. However, most of these stories offer little to no exploration of data. Sometimes, they do include explorable visualizations (e.g., the choropleth map in Figure 1.2 A is interactive: it offers a tooltip, it can switch between nationwide or state-level data, and may even load a different data variable), but this exploration is restricted to the main narrative of the story; users mostly have to follow the author’s line of argument and a directed progression. Unlike a visual analytics system, they cannot break free from the main narrative, freely explore various dimensions of data, and develop their own narrative and understanding.

While traditional visualization systems offer comprehensive and untethered exploration, the data-driven storytelling focuses on the communication aspect of data analysis. In fact, both can be con-

*<http://vcg.github.io/upset>

†<https://projects.fivethirtyeight.com/redistricting-maps>

‡<https://web.northeastern.edu/naturalizing-immigration-dataviz>

§<https://www.vox.com/a/weather-climate-change-us-cities-global-warming>

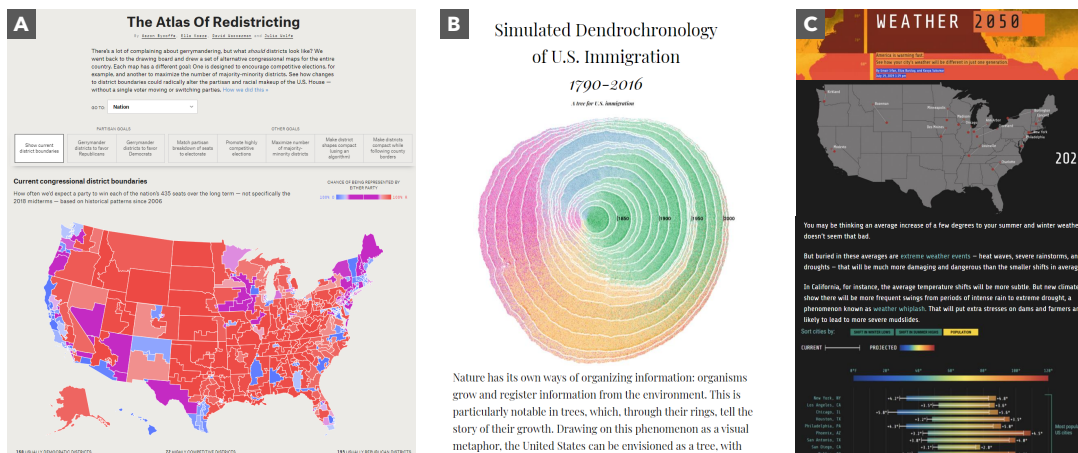


Figure 1.2: Excerpts from the data-driven stories published in various digital news media. They use a combination of text and visualizations to communicate data-driven insights.

considered as two extremes of a continuum between exploration and explanation. By combining them, we can have the best of both worlds: the expressive and *exploratory* power of visualizations, and the flexibility and *explanatory* nature of text resulting into a representation, hereafter referred to as **interactive data documents**, that support *exploration*—a term coined by Ynnerman and others¹⁸⁹ that stems from *exploration* and *explanation*. In such a representation, exploration and explanation should blend in smoothly to reach a point on the continuum. The use of a (natural language) textual narrative brings in an easy-to-understand and self-explaining characteristic. It provides great flexibility for integrating context and backdrop, and can express implicit data explicitly. At the same time, visualizations can provide an overview, spotlight visual patterns, and allow exploration.

To make interactive data documents self-explaining, intuitive, and explorative, natural language text plays a pivotal and multifaceted role. To begin with, text can explain unintuitive visual encoding and appear as descriptive captions of visualizations. Further, it can summarize core analysis insights that can serve as anchor points to guide the exploration process. While exploring visualizations, it can provide explanations or link back to the (textual) insights for better understandability. It can further elaborate analysis methods to even increase the transparency of analysis. Exploiting interactivity and flexibility, text can even provide domain-specific terminology and exemplify it with the current context. Since text is driven by the data and interactively linked with visualizations and user interactions, it would require greater adaptability as opposed to text in data-driven storytelling, which does not have to be fluid. This warrants the use of natural language generation¹³⁷ to automatically produce textual explanation on-the-fly which—unlike pre-written text by a human expert—can be made adaptable to data changes resulting as interactive actions of an end user.

From a technical perspective, authoring of the envisioned interactive data documents is challenging as it entails automatic text generation and linking generated text to various parts of a visualization at a fine-grained level (e.g., brushing-and-linking text with visualization) existing authoring solutions

do not have enough support. Existing tools have poor support for authoring such documents as they often require advanced programming skills, taking care of nitty-gritty details, and going through complex application code. As the authors of these interactive documents are digital journalists and visualization designers, the current workflow as supported by existing low-level coding libraries (D3¹⁹, JavaScript, React) or markup tools (e.g., Idyll³¹) could be tedious and cumbersome. To offload the technical overhead, a desirable solution could move in the direction of an authoring tool that uses an intelligent user interface often called a mixed-initiative interface, that can partly automate some tasks (e.g., identifying and linking relevant text fragments to visualizations) and provide an intuitive user interface to do the rest.

This thesis aspires to bridge the gap between explanation and exploration by conceiving the idea of interactive data documents. In particular, it investigates the role of natural language text as an explanatory medium in visualization systems and reasons for integrating it with exploratory visualizations to make the analysis accessible, self-explaining, and intuitive for a broader audience. The thesis contributes novel methods to develop fully automated interactive data documents for a variety of diverse datasets and application domains including but not limited to bibliographic, geographic, software code quality, and knowledge graph data. Considering the technical challenges and limitation of authoring tools for generating similar representation of data, it also discusses the design of an authoring system using a novel mixed-initiative user interface.

1.1 Research Objectives

To approach the overarching problem, the thesis is subdivided into three main research objectives, each focusing on a specific aspect but contributing toward the final goal.

Text and visualizations are two integral components of the proposed interactive data documents. Therefore, as the first step, it is vital to investigate their role in existing representations of data to develop a deeper understanding of their interplay. Since most visualizations systems do not include any text, the most natural source to extract this information are data-driven stories. Such stories published in high quality digital news media outlets provide an opportunity to get a deeper understanding of the visualization–text interplay. These stories have already been used in other empirical studies to inform the design strategies of similar stories.^{151,69,21} The objective here is to look closely at the reported analysis insights and discover how they are communicated through narrative and visuals. What purposes text serve in these stories, and how is it embedded inside or along with visualizations? Finally, what strategies are employed to perceptually link the two media.

Research Objective – RO 1

Understand different roles of textual narrative in data-driven stories and discover how it is interwoven with the visual representation of data.

In a usual data analysis workflow, insights are first identified, then prioritized, and ultimately described as a narrative. These are then presented alongside visualizations in a data-driven story. The

narrative in these stories is authored by a human expert. However, interactive data documents would demand textual explanations to be flexible and adaptable to data changes in reaction to user interactions. For instance, to always align a caption with ever-changing data in a visualization to describe the current state, it would require on-the-fly production of text. Therefore, automatically generating text using natural language generation techniques can be beneficial for such purposes. It would easily adapt to data changes and require less effort than manually writing explanations to cater every possible use case that may arise during the free exploration of data. The generated text could serve many purposes: It can hint at notable insights and intrigue users to explore the visualizations. The ability of exploration would provide an opportunity to even verify the generated insights and facts. Moreover, while exploring a visualization, users would get explanations, not only about what the visualization depicts, but also brief notes on the analysis methods and esoteric terminology. This would ultimately lead to better transparency in data communication.

Research Objective – RO 2

Leverage natural language generation techniques to automatically and on-the-fly produce textual explanations of data and derived insights.

Since text and visualization in an interactive data document will describe the same underlying data at different levels of abstractions, implicit connections exist between the two representations. For an interplay, both representations need to be coherently integrated. This goes beyond placing visualizations close to relevant text. The challenge is to seamlessly integrate generated textual explanations with explorable visualizations to provide an obtrusive, yet non-distractive, blend. The explanations should fuse so naturally with visualizations that it becomes effortless to discover them and—once discovered—they become an invaluable part of the exploration process. Apart from the conceptual side of things, the objective, here, is to also investigate realization of such an integration. Existing authoring solutions require programming expertise that are mostly absent from visualization designers and journalists who are authoring interactive content for the general public. Therefore, to facilitate the authoring process, there is a need for an easy-to-use authoring tool.

Research Objective – RO 3

(3.1) Seamless integration of text and visualizations in an interactive data document where the two representations are in a symbiotic relationship and augment each other as well as (3.2) design of an easy-to-use authoring tool.

1.2 Summary of Contributions and Thesis Outline

The thesis consists of seven chapters; four main chapters discuss the three research objectives, while the other chapters provide background, related work, and conclusion.

In the beginning, Chapter 2 reviews the prior research that has been done on the core concepts this thesis builds on. In particular, the research on data-driven storytelling, natural language generation and processing in the context of data visualization, and integration of visualization and text is relevant.

Chapter 3 (RO 1) presents two empirical studies on different sets of data-driven stories published in news media. They aim at understanding the fine-grained interplay of text and visualizations. The first study investigates the role of every sentence and visualization within stories to reveal how they interplay. Moreover, it explores the positioning and sequence of various parts of the narrative to find patterns that further consolidate the stories. The second study focuses on identifying implicit references between text and visualizations. Drawing from the findings, this chapter further discusses study implications with respect to best practices and possibilities to automate the report generation.

Chapter 4 (RO 2) introduces two interactive visualization systems that present automatically generated insights through a mix of visualizations and text. First, VIS Author Profiles looks at publication records from various perspectives, mixing low-level publication data with high-level abstractions and background information. It is a novel approach to generate integrated textual and visual descriptions to highlight patterns in publication records. It leverages template-based natural language generation to summarize notable publication statistics, evolution of research topics, and collaboration relationships. Seamlessly integrated visualizations augment the textual description and are interactively connected with each other and the text. The underlying publication data and detailed explanations of the analysis are available on demand that make the whole system transparent to the end user. Second, Interactive Map Reports advocates the use of natural language text for augmenting map visualizations and understanding the relationship between two geo-statistical variables. Here, the text generation process is flexible and adapts to various geographical and contextual settings based on small sets of parameters.

Chapter 5 (RO 3.1) discusses the concept of exploranation and its application in proposed interactive data documents. The main focus is to describe a layout and interactive linking model of the document that truly supports exploranation. It initializes the generic concept first and then instantiate it for a variety of different domains including software engineering, knowledge graphs, and virtual reality. Chapter 6 (RO 3.2) presents Kori, a mixed-initiative interface enabling users to construct interactive references between text and charts. Kori leverages natural language processing to automatically suggest references as well as allows users to manually construct other references effortlessly. While the tool assists authors in creating interactive references, the readers profit from an improved synthesis of text and charts leveraging those references. A user study complemented with algorithmic evaluation of the Kori system suggests that the interface provides an effective way to compose interactive data documents.

Finally, Chapter 7 reflects on the major contributions and their implications, as well as provide an outlook into future opportunities and research challenges.

2

Background and Prior Research

Traditional visualization systems focus on visual exploration of data and are designed to help domain experts discover meaningful patterns, identify outliers, and test their hypotheses. Oftentimes, these systems have complex design and are tailored to specific needs of domain analysts and do not clearly communicate analysis results. With the increasing accessibility of data in the public domain, the communication of data is becoming valuable and inevitable. Nowadays, the general public is a direct beneficiary of data analysis in many fields of life. This demands an effective communication of data analysis to a much wider audience—beyond the audience of experts—and has led to the emergence of data-driven storytelling.¹⁴³

2.1 Data-driven Storytelling

Data-driven storytelling—also known as *narrative visualization*—focuses on the communication aspect of visualization and aims for *making data more understandable* for a broad audience.^{143,88} Storytelling leverages design elements, (visual) annotations, embellishments, and a textual narrative to communicate data.¹⁵¹ It connects visualizations with a narrative to produce a data representation that is intuitive and self-explaining. The proportion of text in data-driven stories varies from short captions of visualizations to annotations explaining the main takeaways to long explanations about insights, context, and backdrop of the story. The inclusion of explanations at various stages guides readers through the analysis findings and assist in reading the accompanying visualizations. As a matter of fact, the self-explanatory nature of text and its linking to visual representation is what contributes to the increased intuitiveness and consequently to their widespread outreach.

In the recent past, journalists and visualization designers have been regularly issuing data-driven stories to inform the general public on the happenings in the world about diverse topics such as cul-

ture, sports, politics, and science. As these are published in impactful news media outlets like New York Times, Financial Times, BBC, and many others, researchers in the visualization community have investigated them to reveal effective design strategies and pinpoint what makes them self-explanatory and suitable for a broad audience. Many characteristic factors—related to layout, navigation, role of visualizations, messaging, interactivity, and level of control—have been found to play an important role in how users read and interact with the stories.^{115,151,10} Researchers have employed empirical research to inform design space of these characteristic factors.^{151,69,70} However, the role of text is still under-explored. Text—be it a longer narrative alongside a visualization or brief annotations inside a visualization—is a vital part of data-driven stories and should be investigated at a similar level as visualization. Currently, we lack an in-depth understanding of what different roles it plays—not only as an explanatory medium but also with respect to facilitating exploration process—in data-driven stories and how it interacts with visualizations; this is the first objective of the thesis (RO 1).

Visual analytics systems and data-driven stories can be thought of as two extremes of data visualization; the former embraces exploration and the latter explanation. However, unlike stories published in print media, digital journalism (e.g., Web-based reading) has opened up possibilities to include some form of interactivity and exploration capability in data-driven stories. Existing research has suggested that stories can range from self-running presentations that users consume like watching a video or a slideshow, on the one hand, to interactive ones allowing exploration of data, on the other hand.⁸⁸ However, the exploration is very limited to few standard interactions in most cases. In addition, this limited interactivity is aligned with an author-driven narrative that users follow and has no option to go beyond to explore data from their own perspective.

2.2 Text and Visualization: Two Faces of the Same Data

Text is a flexible medium when it comes to explaining, while visualization is better at revealing patterns and providing an overview of the data. In a data-driven storytelling scenario, they describe the same underlying data, but at different levels of abstraction. For example, writing a narrative based on a scatterplot, the textual narrative may explain the outliers (referring to a few points), clusters (referring to a group of points), or relationship between the plotted data dimensions (referring to scatterplot as a whole). As a consequence, natural implicit connections form between text and visual marks in the visualization; users discover them as they read through the text. Through these links, both complement each other and make the resulting representation intuitive, engaging, and immersive.^{143,73} However, describing information at two different modalities comes with a caveat: the split-attention effect.⁸ The problem originates from the fact that, in a bimodal—including two media: text and visualizations—representation of data, the visualizations need to be placed at a slightly different physical location from the relevant text. This far-off placement of graphics forces the readers to switch their attention back and forth between text and graphics, which can cause a split-attention effect⁹ that increases the cognitive effort to comprehend the information.¹⁶⁶

Sweller, Van Merriënboer, and Pass¹⁶⁸ introduced cognitive load theory. The theory assumes a limited capacity of working memory that users have at their disposal while consuming information. The working memory has partially independent components to process auditory and visual informa-

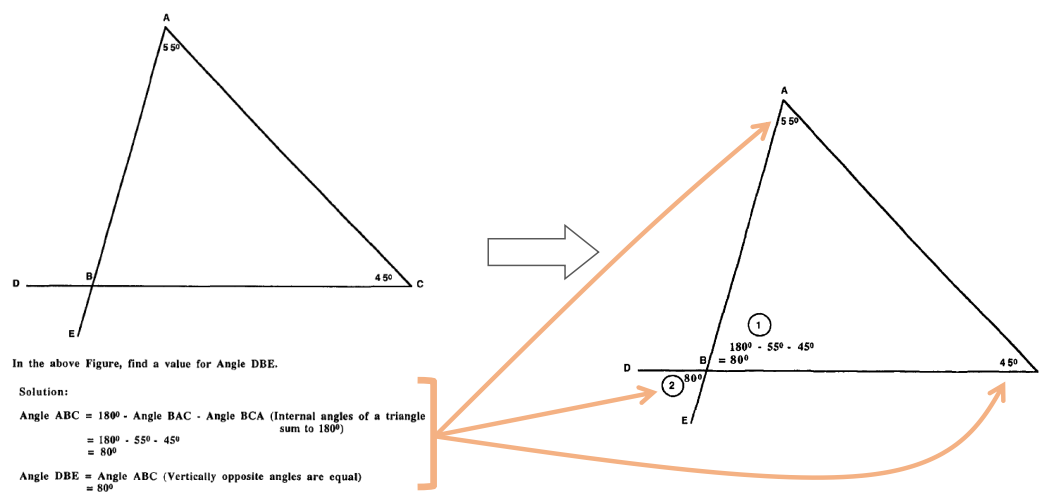


Figure 2.1: An example demonstrating split-attention effect taken from the work of Sweller et al.¹⁶⁸ (Left) Presentation with split-attention effect. (Right) Integrated presentation with no split-attention effect.

tion, as well as holding information in memory. An improper design of information can overload the working memory and reduce the efficiency in consumption of information. Figure 2.1 shows an example of explaining a simple geometric problem. On the left, the solution is presented with a diagram and series of equations. In this representation, the diagram alone communicates nothing, but only when it is read together with the equations. To understand it, users must integrate the two sources mentally; for instance, begin with an equation, hold it in the working memory, and then look for related reference in the diagram. This process can be cognitively demanding and stems purely from the particular style of presentation. As an alternative, the solution presented on the right integrates the equations right inside the diagram therefore eliminating the need of mental integration, thus, saving working memory of users.

In a data-driven story, it is challenging, from readers' perspective, to synthesize information across two distinct media as these are spatially separated apart.⁸⁷ The readers have to switch their attention back and forth to find implicit references between text and visual marks in a visualization that encode data values (e.g., bars, lines, points) and vice versa. This phenomenon incurs a significant cognitive burden on readers' working memory and can have a negative impact on learning.^{9,26} The cognitive load theory argues that information should be presented in a way that it does not overload the working memory of users. In particular, talking about data documents that includes two distinct media, the information should be coherently integrated in a way that reduces the split-attention effect.

2.3 Integration of Text and Visualization

Although text and visualizations are spatially separated apart in a data-driven story, they are semantically associated as they describe the same underlying data. In line with cognitive load theory, two possibilities naturally arise to bring the two representations closer: The first is to reduce the physical distance through the use of small visualizations (word-sized graphics) that can be completely embedded inside the lines of text where they belong. Second, we can improve the linking through visual cues and interactivity. Taking inspiration and advancing the state-of-the-art, we discuss *word-sized graphics* and *interactive visualization-text* linking as two powerful means to bring text and visualizations closer in the context of interactive data documents.

2.3.1 Word-sized Graphics




Micro visualizations embedded into the lines of text are known as *sparklines*¹⁷³, *word-sized*¹⁸, or *word-scale*⁵² graphics. Tufte defines them as “*data-intense, design-simple, word-sized graphic[s]*”.¹⁷³ In contrast to regular-sized visualizations, these graphics are produced at the height of a word. Due to their small size, they can be completely embedded within the lines of text. This allows readers to remain focused on the same spatial area while consuming the information. For instance, fluctuations in EUR–USD currency exchange rates (2012–2016) can be easily seen in this word-sized line chart 

Figure 2.2 illustrates how the use of word-sized graphics (e.g., , ) integrated within a text or table can reduce the split-attention effect by presenting most of the visualizations next to the relevant text. The representation on the left includes two large visualizations and passages of text (marked in blue and green) that correspond to each visualization. This representation forces readers to switch their attention back and forth to relevant visualization while reading text and constructing references in working memory. The representation on the right contains the same information, but large visualizations are replaced by their word-sized counterparts. Since these are embedded right next to the text that references them, it reduces the split-attention effect by providing physical integration; users do not need mental integration anymore as they would in the former representation. Likewise, in a table, word-sized visualizations can provide a visual comparison of information by placing multiple instances next to each other.

Although word-sized graphics have been widely discussed in the literature, their integration in data-driven stories and other interactive data documents is still scarce. Beck and Weiskopf¹⁸ present a survey on their actual use in existing research. It was discovered that most of the existing research introduces word-sized graphics just as examples or uses them to augment other visualizations.³⁹ With respect to their integration in other media, they are most frequently included in the source code of a computer program; for instance to observe its runtime behavior^{169,64}, performance bottleneck¹⁶, or to monitor variable values over a program execution.^{58,165} Beyond software engineering, they have been leveraged for literature data analysis.^{121,18,15} However, their inclusion in longer text, especially as part of data documents, is not very common.

Only a few researchers have explored their usability with respect to integrate visualization and textual representations of data. Goffin and others⁵³ have explored the design space of word-sized graphics

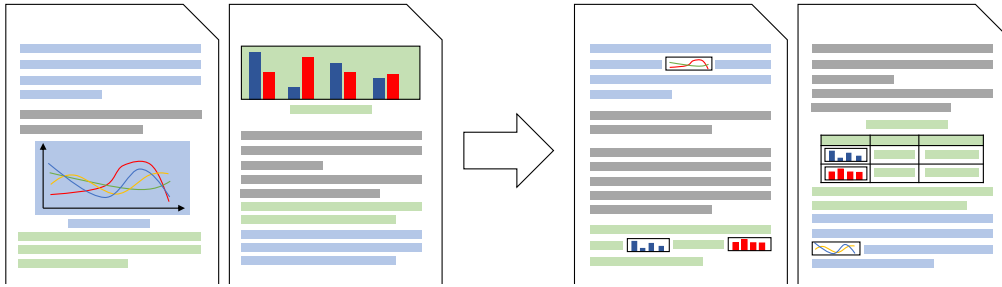








Figure 2.2: (Left) Document without word-sized graphics where the reader has to switch attention between graphics and text. (Right) Document using integrated word-sized graphics in line with the text and in tables to reduce a split-attention effect and save space for additional content (here, a table with further information). Colors ■ ■ indicate the textual and visual content that relate to each other.

for text documents and several placement options (e.g., in line with text, as an overlay on top of text, between two lines of text, and so on) to integrate them within the lines of text. They found that information encoded in word-sized visualizations and put right next to the related text was more prominent to the users and had a positive effect in memorability as well as recall. Though various placement options did not have any significant impact on the reading behavior.⁵² Likewise, Beck and Weiskopf¹⁸ proposed that the word-sized visualizations embedded inside lines of text may reduce split-attention effect. Further, Beck and Weiskopf suggested that interactive counterparts of word-sized visualization can also be beneficial as a quick in-place unit of information (often showing parts of the data) and ultimately leading users—when interacted with—to the large visualizations which would be more comprehensive. This way, word-sized visualizations can serve as a bridge between text and large visualizations, reducing the distance between the two representations in the information space.

However, in most cases, the use of word-sized graphics is restricted to simple line, bar, or proportion charts with a very few exceptions—Hlawatasch et al.⁶² visualize trajectories. Probably that gives a misconception that word-sized graphics can only encode fairly simple data as few straightforward visualization types (e.g., bar, line, proportion) Interestingly, they can represent any type of graphic including but not limited to bar charts, scatter plots, box plots, or even node-link diagrams. This thesis argues that word-sized graphics are much more flexible with respect to the type of visualization and the data they can encode. Our prior work investigates the use of both static and interactive word-sized graphics in achieving a coherent representation of content presented across two media.⁹³ This particular paper extends the design space of existing word-sized graphics to represent comparatively complex data including multivariate  , spatial  , and graph or relational data  . Interested readers may look up details in the paper⁹³, especially with respect to the applicability of these word-sized graphics in a realistic scenario.

2.3.2 Interactive Visualization–Text Linking

As discussed earlier, text and visualization describe the same data and, therefore, implicit references exist between the two representations. For instance, while describing a scatterplot, the text usually

references the entities encoded by individual dots, groups of dots, or a visual annotation in the scatterplot. Users discover these implicit references as they first read through the text and then tend to look at the relevant part of a visualization, thereby increasing the cognitive load on their working memory. In an interactive data document, a way to reduce the distance between text and visuals in the information space is to convert these implicit references to explicit and interactive links.¹⁸ The interactive visualization–text linking corresponds to providing visual cues (e.g., visual highlighting) to quickly guide users’ attention to the relevant portion of a visualization while interacting with the corresponding text fragments and vice versa. This linking aligns with the signaling principle that argues for guiding users’ attention from one medium to the other in a multimodal representation as a possible antidote to reduce cognitive burden.¹¹²

Existing research has shown that this interactive linking can facilitate users in reading a data document, particularly with regard to visual exploration of data, as well as in interpreting visualizations. For instance, *VisJockey*⁹⁰ offers the possibility to animate the related parts of a comparatively complex visualization (e.g., parallel coordinates) by interacting with the corresponding text fragment. Similarly, Figure 2.3 shows two more examples of interactive linking between text and visualization or table. (Right) Beck and Weiskopf¹⁸ use visual highlighting to guide users attention to the relevant part of the bar chart while clicking on the interactive text fragment (printed in boldface in the figure). (Left) Kim et al.⁸⁰ advocate for an interactive *text–text* linking to facilitate the reading of a document that comprises many tables. Their approach links the main body of text with the associated text in tables. Moreover, existing research applies this type of linking directing from textual narrative to visualizations assuming the standard reading strategy—read the text first and then explore the visualizations. However, in an interactive data document, users may wish to explore the visualizations first and then read the corresponding text. Following this alternative reading strategy, Beck and Weiskopf¹⁸ propose an abstract idea of a bidirectional interactive linking between text, word-sized visualizations, and regular visualizations. Building on Beck and Weiskopf’s abstract linking model¹⁸, our prior research instantiated this model for graph data.⁹⁷ The approach uses a declarative syntax to produce an interactive data representation as shown in Figure 2.3; the details can be found in the paper.⁹⁷

Recent research in the visualization community has also looked into the gains of interactively combining text and visualizations. An effective linking and layout strategy can have a positive impact on comprehension and information recall.¹⁹⁴ The impact is even more obvious among the users that had low visualization literacy.⁹¹ Particularly studying the impact of explicit visualization–text linking (visual marks in the visualization were highlighted when hovering over a relevant phrase of text) Zhi et al.¹⁹⁴ found that participants recalled information better when it was interactively linked across both representations. Another experiment by Barral et al.^{13,91} achieved somewhat similar results, yet using a different type of linking method. In contrast to explicit linking¹⁹⁴, Barrel and others used a gaze-driven approach; the relevant parts of a visualization were highlighted based on participants’ eye fixation on a related sentence in the textual narrative. This adaptive gaze-driven linking helped improve comprehension, particularly among participants with low visualization literacy. When studying the impact of explicit visualization–text linking in the context of a Bayesian reasoning problem, Ottley et al.¹³¹ discovered that people tend to consolidate information well across the text and visualizations when they are interactively linked.

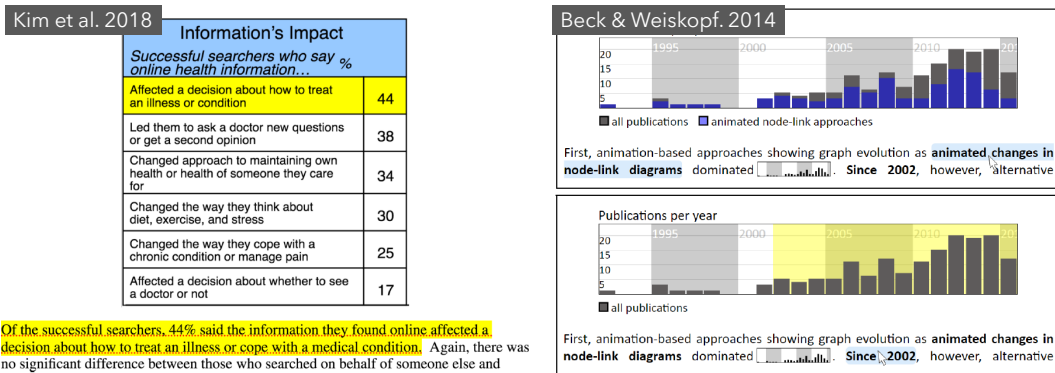


Figure 2.3: Two examples of interactive referencing in literature: (Left) Kim et al.⁸⁰ use highlights related text in tables while reading the corresponding text. (Right) Beck and Weiskopf¹⁴ propose the idea of visually highlighting relevant parts of the visualization while interacting with associated text elements.

We have evidence that a joint and well-connected representation of data including both text and visualization is beneficial for the end users, especially for a broader audience. However, achieving such an integrated representation where both text and visualization augment each other, yet keeping the cognitive burden¹⁶⁷ low that may arise from context switching in a bimodal representation at the same time, is challenging and under-explored. One of the main objectives of this research (RO 2) is to explore various linking methods that bring text and visualization closer and reduce the gap in the information space (Chapter 3).

2.4 Automatic Text Generation for Data and Visualization

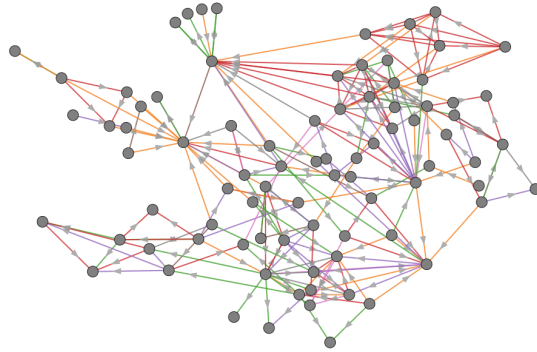
As an alternative to human-authored text, automatic generation approaches can be employed to produce a narrative from data, for instance, to explain analysis insights. These approaches fall under the scope of natural language generation (NLG) which deals with producing natural language text from data and other abstracted forms of information.¹³⁸ The most frequent and well-known use case of natural language generation are personal assistants (e.g., Google, Siri, Alexa), weather forecasts, and directions you get in a vehicle navigation system. There exist various generation approaches in general⁴⁸ ranging from the ones using artificial intelligence (e.g., Generative Pre-trained Transformer 3) to the ones relying on simple templates often utilized in weather forecasting and navigation systems. Despite its widespread applicability to many domains, only a few have investigated the generation of text for data visualizations, and these approaches are the focus of discussion here.

2.4.1 Text Generation for Statistical and Other Forms of Data

Existing approaches generate textual content ranging from simple quantitative univariate data^{36,68,71} to comparatively complex imaging data.^{76,122} The previous work is scattered across many domains:

Jon Snow's Biography

Jon Snow is the son of Rhaegar Targaryen and Lyanna Stark. However, Jon's true descent is kept secret until very recently. He is raised as the bastard son of Lord Eddard Stark, Lyanna's brother, at Winterfell. Considering Eddard's lawful children have better claim to Winterfell, Jon is sent to the Night's Watch, where in a dramatic turn of events he becomes the Lord Commander.



The node-link diagram shows relationships among various characters of the TV series *Game of Thrones*.

At Castle Black, he is killed in a mutiny and then brought back to life by the Red Priestess Melisandre. Upon receiving a letter from his foe Ramsay Bolton, he joins forces with Sansa Stark and together they defeat Boltons, reclaim Winterfell, and Jon is named as the King in the North. After meeting the Dragon Queen Daenerys Targaryen, true heir to the iron throne, Jon pledges his sword to her and steps down as the King in the North.

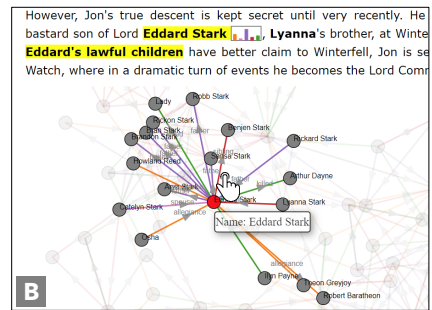
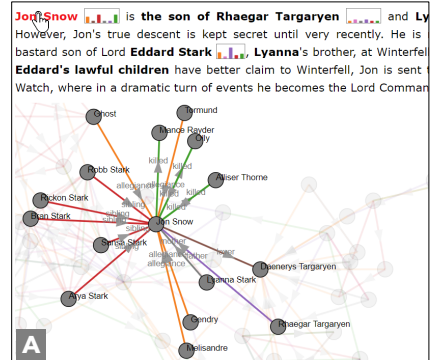


Figure 2.4: Interactive data document describing the biography of a fictional character, Jon Snow, from the TV series *Game of Thrones*. Two visualization-text linking interactions: (A) The result of hovering over an interactive text and (B) a node in the node-link diagram.

GALIWeather¹³⁵ automatically generates linguistic descriptions for short-term weather forecasts based on the analysis of climate data, the *WIP* system¹⁷⁸ generates instructional and maintenance manuals for simple machines, and *SoftLearn Activity Reporter*¹³⁶ uses verbalization to interpret the performance of students in a virtual learning environment. Automatic text generation has also been used in the context of software engineering^{25,101,102,118,33}, geographic data^{175,34}, and transportation.^{20,175} Some approaches deal with generating textual reports for source code, such as code documentation and summarization.^{159,124,113} Another interesting use case of natural language text is the communication of data analysis results to the visually impaired population¹⁷⁰; text can ultimately be read out aloud. While these approaches generate text from data, they do not consider or coordinate with visualizations and text is not always presented alongside visual representation of data.

Other approaches focus on summarization of the analysis results that are to be put right next to a visualization. For instance, Sripada and Gao¹⁶² present the scuba diver's depth-time profile with a line plot. Similarly, Jain and Keller⁷⁴ generate summaries of healthcare data gathered by sensors installed in the homes of elderly people who are living alone. The main objective here is to support medical staff and nurses to quickly monitor and act in case of an anomaly. *Query-to-question* (Q2Q)¹²⁶ system summarizes the progression and sequence of user interactions from the analysis of interaction log files.

However, these approaches still consider visualization and text as two separate mediums, yet they are presented together.

2.4.2 Text Generation for Data Visualization

Unlike data-driven storytelling, the use of text is not common in visual analytics systems at all. Majority of these systems even lack descriptive captions of visualizations. Only a handful of these systems combine textual narratives to provide guidance to the users in exploring the data. Text can be employed to provide interventions while the users explore a visualization, either to explain or to offer guidance. For instance, *Voder*¹⁶⁰ generates short textual insights or data facts to guide users in exploring a multivariate dataset. The generated data facts serve as interactive widgets and suggest other relevant visualizations to further facilitate the data exploration process. Likewise, *CauseWorks*³⁰ uses longer textual narrative well-connected to a causal network visualization to explain causality. It was discovered that the coupling of causality visualizations with a textual narrative significantly increases accuracy and narrative acts as a pivotal component augmenting visualizations. In contrast, for long text documents including many tables (e.g., annual budget reports), visualizations can augment the document with visualizations to make it more interpretable. One such example is *Elastic Documents*¹¹; it provides an interactive viewing interface augmenting text and tables with on-demand contextual visualizations. When compared against a conventional PDF viewer, it was found that this combination of text, tables, and visualizations improves the quality of summarization as well as comprehension to a moderate extent. Another system, *Method Execution Reports*¹⁷, automatically summarizes the execution behavior of a software program and includes interactive word-sized graphics inline with the text.

From an end user's perspective, there are several characteristics (e.g., working memory, perceptual speed, information needs) that matter while interacting with a visualization system.¹⁷¹ These characteristics can be leveraged to suggest meaningful interventions (like the ones provided in *Voder* and *CauseWorks*) to help users—especially those with low abilities—for processing visualizations.¹⁷¹ To generate interventions for guiding exploration and offering explanations about a visualization, both representations need to be completely intermingled. As opposed to data-driven stories—where text is human-authored and appears alongside visualizations and is often not adaptable to user interactions due to very limited exploration capability—we require a much more flexible generation approach that considers joint creation of textual and visual content. This is another major objective of the thesis (RO 3, Chapter 4).

2.5 Interactive Data Documents

Visual analytics systems support exploration but have little to no support for communication of knowledge or insights that are gained through the analysis. Data-driven storytelling, on the other hand, is suitable for the explanation and dissemination of data analysis results to a broad audience, but has very little support for exploration. Targeting a sweet spot between explanation and exploration, this thesis aims for an explorative solution, **interactive data documents** offering both explanations and ex-

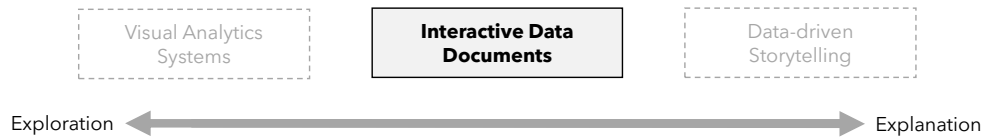


Figure 2.5: The envisioned data-driven documents lie at the sweet spot between visual analytics systems and data-driven stories.

ploration (Figure 2.5). A novel and seamlessly integrated combination of text and visualization brings the benefits of both media. Authoring an interactive data document would involve construction of three main components: natural language text, exploratory visualizations, and the interactive linking between the two. While we have already discussed the first component in the previous section, this section discusses the challenges and existing support with respect to the latter two components. However, we begin by looking at the authoring support for data-driven stories first, as they also closely relate to interactive data documents from a technical design perspective.

2.5.1 Authoring Support for Data-driven Storytelling

With the increasing popularity of data-driven stories in digital journalism, more and more researchers are exploring ways to author them with ease. Existing tools that help create such stories allow users to add textual descriptions and annotations, as well as to customize visual marks and layouts.^{164,109,32,140} Many of these tools focus on allowing users to build a narrative around a single, oftentimes static and non-interactive visualization.^{147,141,106,142,186,83,82} Others support creating a complete story with a sequence of logically connected visualizations and textual explanations as annotations.^{55,7,146,21,81,116} Existing research also explores novel forms of data-driven stories including videos⁷, comics⁸¹, and slideshows.^{55,146,21}

Most of the existing authoring approaches can be broadly classified into two types. First are the ones that support manual creation of data stories: Among these, *DataClips*⁷ provides an authoring interface for data videos with different templates that users can customize; *Data Illustrator*¹⁰⁷ supports data binding to expressive charts for making data stories memorable; Ren et al.¹⁴⁰ discuss the design space of annotations and present an interactive tool to create them; and Brehmer et al.²¹ facilitate the authoring of timeline narratives. In contrast, the second type of authoring approaches provides automatic support. Notable among them are *Datashot*¹⁷⁹ and *Calliope*¹⁵³. The former automatically derives data facts from tabular data and generates infographics to provide an overview, and the latter supports the automatic generation of a story sequence directly from a given dataset.

2.5.2 Creating Exploratory Visualization

Visualizations will contribute the major part of exploration in the interactive data documents. Prior research has also explored ways to author interactive visualizations with ease and using minimal programming. The use of declarative syntax such as Markdown has made it even easier to create interactive visual content.^{78,97,98} Vega¹⁴⁹ and Vega-Lite¹⁴⁸ use a simple JSON syntax to produce interactive visu-

alizations. Low-level libraries like D3¹⁹ provide good support for developing highly customizable and explorable visualizations. Computational environments such as *Jupyter Notebook*, *R Markdown*, and *Observable** are systems that aim at creating and sharing computations or graphics in a reproducible way. It is possible to create interactive content using these systems, but they focus on interactive coding experience and target technical users.

2.5.3 Establishing Linking of Text and Visualization

While existing tools provide ways to add annotations and textual explanations inside or close to visualizations as discussed in Section 2.5.1, they rarely go beyond positional linking, let alone interactive linking. Realizing the benefit of interactive visualization–text linking, some scientific publishers such as *Authorea*[†] and *Elsevier*¹³⁰ attempt to support this integration, but are limited to very simple linking (e.g., finding a related figure given an explicit text reference). Kong et al.⁸⁷ recently used crowdsourcing to reconstruct references between textual phrases and visual marks on the charts in existing data stories and highlight their importance in reading such a document. Similarly, Metoyer et al.¹¹⁷ automatically integrate short textual annotations at various points in the visualization when users highlight a passage of text.

In these tools, textual parts are mostly considered passive supportive elements—semantically connected yet separated from the associated visualizations. Therefore, practitioners resort to programming libraries (e.g., D3¹⁹) or frameworks (e.g., *Idyll*³¹) in order to create interactive references between the two (see an interactive data document about Boston’s subway system[†]). *Idyll*³¹ introduces a markup language combined with reactive programming in JavaScript for creating interactive data documents for the web. The focus of *Idyll* is broad, and it allows building custom visualizations using D3 or Vega-Lite and binding them to the text. This generalizability comes at the cost of programming custom visualizations. Hence, solutions including *Idyll* or D3 require programming expertise.

In contrast to existing tools and programming frameworks, we target users who do not have programming expertise and aim to provide an accessible user interface for constructing interactive data documents including interactive linking between text and visualizations; the third objective of the thesis (RO 3).

*<https://observablehq.com>

†<http://mbtaviz.github.io/>

3

Understanding Visualization–Text Interplay

Data-driven storytelling combines the expressive power of visualizations with a textual narrative to communicate analysis findings to a broader audience. In such stories, both representations seem to complement each other; visualizations provide an overview of the data while the accompanying text hints at insights and blends in the context and backdrop to make the story engaging, compelling, and intuitive. Since text and visualization describe the same underlying data, an interplay exists between the two. This interplay relates to spatial arrangement, positioning, sequencing of visuals in the narrative, the embedding of text inside visualizations, and how various parts of the story reference other parts. Prior research has discovered—by means of small-scale user studies—that these factors, especially the spatial arrangement and interactive linking of text and visualization influence the readers’ engagement, comprehension, and information recall.^{131,194,13,91} A deeper understanding of visualization–text interplay can reveal effective design strategies for textual narrative and visualization for authoring joint representation of data.

This chapter aims to understand the visualization–text interplay at a fine-grained level. It begins by presenting research questions focusing on different aspects of the interplay. It then describes two empirical studies aiming to answer these research questions. Finally, it is concluded with a systematic characterization of the learned integration strategies, and possibilities to automatically generate parts of the textual narrative that would describe analysis insights.

3.1 Research Questions

For understanding visualization–text interplay, we formulated three research questions, each focusing on a different aspect. The first question deals with analysis insights and their textual and visual communication. It is obvious that text in a data-driven story serves various purposes. A major part

of the text reports insights that result from the data analysis, while a comparable proportion conveys the context behind these insights (e.g., the backdrop of a story, opinion of researchers or politicians) to make the story interesting for the readers. Likewise, various visualizations are employed to provide an overview of the data as well as highlight notable insights. Learning about what insights are communicated and how they are presented would help in automating their generation.

RQ 1 – What are the reported analysis insights and how is the related data visually communicated?

RQ 1.1 What are the analysis insights presented in the textual narrative, and how is context blended with these insights?

RQ 1.2 How are visualizations used as a complement to communicate the data?

The main strength of data-driven stories lies in the fact that they closely integrate narrative and visualization at different levels.¹⁵¹ For instance, placing visualizations in the proximity of relevant text reduces the distance in the information space and facilitates readers to quickly glance at the corresponding visualization while reading a specific paragraph. Similarly, text inside a visualization may hint at the main takeaway of that visual and contribute to its self-explainability. Oftentimes, several visualizations are employed to demonstrate distinct aspects of the same data; their sequence in the story can be crucial. For example, including an overview visualization up front could familiarize users with the data before presenting a certain aspect of analysis. The second research question aspires to understand the connections between the two media.

RQ 2 – How do textual narration and visualization interplay?

RQ 2.1 What links exist between the two media?

RQ 2.2 How and in what sequence are visualizations embedded into the narrative?

Since both text and visualization are based on the same underlying data, it is natural to have implicit references between the two representations. Such references relate to the phrases of text that has a visual representation in a visualization. For instance, imagine a scatterplot of countries; every mention of a country or continent name in the text would refer to a single or group of points (visual marks) in the scatterplot (Figure 3.5). Previous research has shown that users, in particular the ones who lack visualization literacy, often have a hard time consolidating information that is presented across text and visualizations.¹³¹ Extracting such implicit references and converting them to explicit and *interactive references* helps in better consumption of information.^{13,91,194} With this focus, the third research question aims at exploring the design space of implicit references in existing stories.

RQ 3 – What implicit references exist between text and visualization and how do they relate to the data?

RQ 3.1 What type of implicit references exist between text and visualizations?

RQ 3.2 How do these references relate to data and visualizations?

3.2 Methodology

To answer the research questions (RQ 1 – RQ 3), we adopt a similar approach as applied in several existing works.^{151,69,115} We perform two empirical studies: first to answer RQ 1 and RQ 2, and second—due to slightly different data needs—to answer RQ 3. For both studies, we follow a qualitative approach focusing on fewer examples but a fine-grained and deep analysis as we—unlike previous research—are particularly interested in exploring the possibilities of automatic generation in addition to deriving the best practices for designing similar content. This is also why the stories should have high quality, both with respect to their textual narration and visual data representation.

As RQ 1 and RQ 2 emphasize a lot more on the communication of analysis insights, here the stories relating to geographic data are particularly interesting as the spatiotemporal nature of data makes the reporting challenging. Unlike reporting plain time series (e.g., the revenue of a company) or results of public-opinion polls, it usually requires multiple visualizations to show different aspects of the spatiotemporal data; some with a geographic focus and others with a temporal one. We find examples of geographic narratives across diverse journalistic branches such as politics, economics, science, and health. The COVID-19 pandemic further provided the unique opportunity to collect various polished examples from the same context. We investigate the role of every sentence within each of the narrative categories and how sentences are interwoven with the visual representation. Besides, we explore the positioning and sequential patterns among various parts of the stories.

In RQ 3, we are particularly interested in implicit references and what visualization features (e.g., legend, visual marks, axes) of a visualization they refer to. Therefore, a lot more examples with a variety of visualization types are needed. In the existing literature, Kong et al.⁸⁷ already performed a similar study on a small-scale dataset. We use their dataset as our basis and expand it with more examples and even diversify it with respect to visualization types.

In the rest of this section, we refer to these two studies as *Study I* and *Study II* while explaining the data collection and analysis process. Afterward, the results of both studies are organized into different sections—one section per research question.

3.2.1 Data Collection

To ensure quality and diversity, we manually picked examples from well-known news media outlets and research publication venues (only valid for Study II).

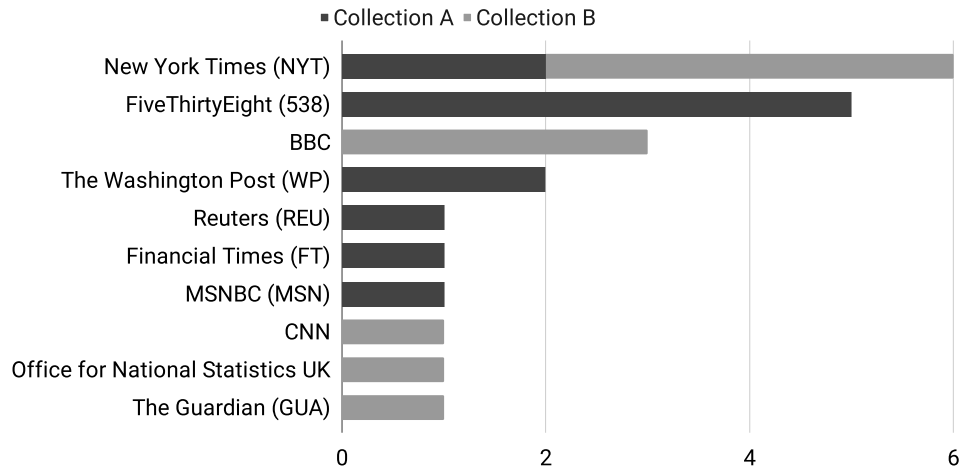


Figure 3.1: Sources of stories in the sample data collection for Study I. Almost 50% (14/22) of the stories were gathered from three well-known sources: NYT, 538, and BBC.

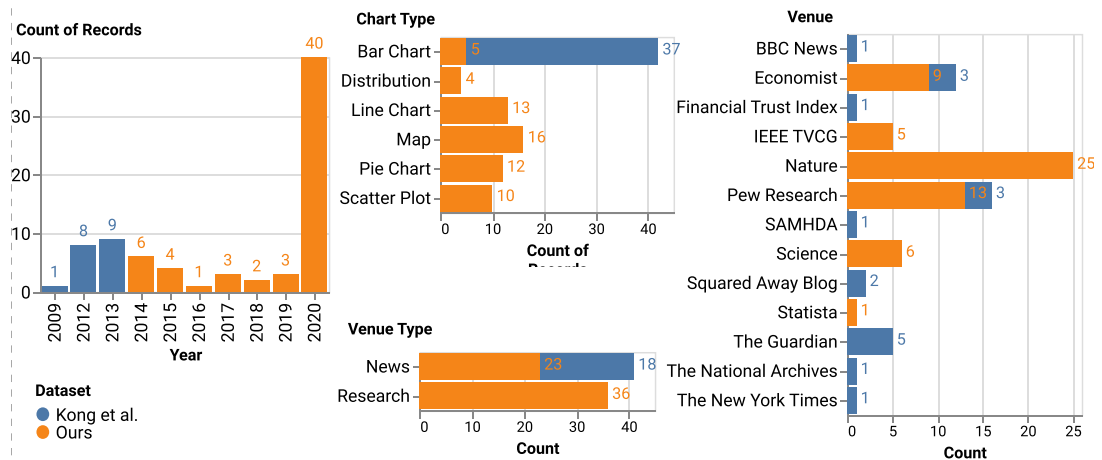


Figure 3.2: Overview of the sample data collection for Study II. We augmented existing articles from Kong et al.⁸⁷ with additional venues and chart types.

STUDY I

Twenty-two stories were collected from 10 well-known digital journalistic sources including New York Times (NYT), FiveThirtyEight (538), and BBC; the full list of sources is shown in Figure 3.1. The stories are published between 2016 and 2020. Our story selection criteria involved the presence of at least one geographic visualization and a comparable proportion (in terms of screen real estate) of textual and visual narrative. Another but less strictly applied criterion was the presence of interactivity. We began with searching for stories that contained visualization–text interactions (e.g., interacting with text visually highlights the relevant part of the visualization or vice versa). Having found only 3 such stories, we loosened the criterion of interactivity to visualizations alone in the story. Later, seven stories were also included that did not offer interactivity. In our sample collection, fifteen out of 22 stories offer some form of interactivity.

In the first phase, we picked 12 stories (Collection A) on a variety of themes such as culture, economics, politics, science, and health to maximize the diversity of topics. In the second phase, we chose another 10 stories (Collection B) on a single topic: the COVID-19 pandemic. These 10 stories have the same context yet cover various aspects of the pandemic. The two collections complement each other; one embraces diversity, while the other focuses on certain comparability.

STUDY II

Since Kong et al.⁸⁷ already conducted a similar study on a small-scale dataset, we began with their data collection as our basis. However, their dataset was limited to 18 articles gathered from a variety of news media outlets. Besides, it was restricted to bar charts alone. We expanded this collection with additional chart types and sources, resulting in 77 articles comprising 110 paragraph–chart pairs. We targeted three main sources: research articles published in (i) Visualization journals (e.g., TVCG), (ii) Nature—the world’s leading multidisciplinary science journal—, and (iii) articles published in the digital news media as web browser-based stories. Within these sources, we randomly and manually picked examples to maximize the diversity of visualization types. Figure 3.2 shows the distribution of our sample collection regarding their venues, chart types, and timeline while comparing it to the Kong et al.’s collection.

3.2.2 Qualitative Analysis

The analysis aims at understanding the fine-grained interplay of visualizations and textual narration. Therefore, it goes down to the individual sentence level to understand how sentences are related to data and visualizations. An open coding approach was followed in both studies; the details of the exact process are as follows:

STUDY I

Every story in the collection of 22 stories was divided into individual sentences and visualizations. This resulted in 1,203 sentences and 118 visualizations (638/66 for Collection A and 565/52 for Collection

B). The coding (i.e., labeling the sentences and visualizations) proceeded as follows: two coders (both coauthors of this paper) used 4 stories from Collection A as seeds and independently assigned descriptive codes to sentences as well as visualizations. In a follow-up meeting, the codes were discussed; similar codes were merged, and conflicting code assignments were resolved. This initial coding scheme was then rolled out to the rest of the eight data stories in Collection A. For this, a sequential process was adopted: one coder did the coding first, and then the other coder checked and refined the first coding. The analysis of Collection A provided us with a coding taxonomy that was then verified and further fine-tuned with its application to Collection B. We followed the same process to analyze stories in Collection B. Over the course of several meetings, we kept on resolving and consolidating the codes and categories, ultimately resulting in 45 distinct codes across 4 categories and 12 subcategories.

Overall, the coding process resulted in 25 codes for sentences and 20 codes for visualizations (cf. Figure 3.3). In total, there are 1,812 code assignments for sentences and 569 for visualizations. Our coding scheme allowed for multiple code assignments to a sentence or visualization. We group these codes along the categories **data-driven** and **embedding** for textual narrative (sentences), **visualization** for visualization-specific codes, and **visualization–text linking** for the interplay between the two media (e.g., a sentence that references a visualization or a visualization that has a textual annotation). As shown in Figure 3.3 (leftmost column), the colored coding categories have further subcategories that will be discussed along with reporting of the results (Section 3.3 and Section 3.4). All codes and code categories are always underlined with the respective color while reporting the results for improved readability and figure–text linking. The categories and subcategories are printed in bold font to discern them from the codes.

STUDY II

The sample collection for the second study consists of 110 paragraph–chart pairs extracted from 77 different articles. Likewise, in Study I, we divided each paragraph–chart pair into sentence–chart pairs for our analysis. This division resulted in 227 pairs, including 82 different charts across six distinct chart types. In line with Kong et al.’s⁸⁷ methodology of identifying implicit references, we first manually constructed *minimal* references between text and charts. A reference from a sentence to a chart is minimal if adding more words could increase matching data points in the chart while removing words would make the matching ambiguous. Two researchers independently followed an open coding process to analyze the sentence–chart pairs. The researchers used the collection of Kong et al.⁸⁷ as a basis to derive the initial codes. These codes were then applied to our collection and were expanded. As for Study I, we iteratively resolved any conflicts that arose during the process to reach a consensus and kept on merging similar codes.*

*This analysis was performed along with Nam Wook Kim and Zheng Zhou from Boston College. While Zheng Zhou was mainly responsible for data collection and labelling, Nam and I closely worked together to derive initial codes, organized them into categories, and kept on refining the codes. Toward the end, Nam further expanded the coding scheme to include more code categories (e.g., hierarchical grouping), and together we discovered core insights that are described in Section 3.5.¹⁰⁰

3.3 Results: Insights and Visual Communication (RQ 1)

First, we study the ingredients of the stories, namely the individual sentences and visualizations. Figure 3.3 gives a qualitative overview of what these ingredients are, but also reports related quantities (i.e., how frequently a certain code is assigned). These quantities are not meant to generalize beyond a specific story but help us judge the general character of a story (e.g., working a lot with direct quotes) and find interesting outliers (e.g., a unique style of reporting). In the following, we systematically discuss these ingredients along the code categories and subcategories, clarifying their meaning as well as describing their typical use and notable examples.

3.3.1 Analysis Insights and Context (RQ 1.1)

Generally, we observe two main categories of **textual narrative** in data-driven stories: the actual **data-driven** text and the text that serves as the **embedding** in the story, for instance, structuring text like headings or contextual information like dataset descriptions. The **Data-driven** text does not just list the raw numbers but summarizes analysis findings at a higher level as *insights*. Although there seems to be no agreed definition of *insight* in the visualization community²⁸, it may be defined as “*complex, deep, qualitative, unexpected, and relevant*”¹²⁹ or “*an individual observation about the data [...], a unit of discovery*”¹⁴⁵. In the following, we define an *insight* as a non-trivial, qualitative, and relevant observation about the data. An example of an insight from A02 is: “[i]n some states, like Montana and Alaska, nearly the entire adult population is registered [as organ donors].”

In geographic stories, **geotemporal** entities—**location** and **time**—are usually key terms of the textual description of the insights. Almost all stories contain (20 of 22; see Figure 3.3) identifiers of **locations**. While most locations are referenced by their specific names (e.g., “Boston” – A09, “Massachusetts” – A02, “USA” – B09), a variety of collective terms according to geopolitical, geographic, or administrative units are also used. For instance, A01 describes counties suffering high casualties as: “[r]ural Appalachia stands out; nine counties in Kentucky and three in West Virginia make the list.” Appalachia is a region in the eastern US and is not marked on the map visualization; the reader’s knowledge is presumed. Other variations include “Dakotas”, “among the peaks of the Rocky Mountains” (A01), and “Midwest” (A02). The directional phrases such as “west of the Mississippi” (A01) and “southern tip of Bangladesh” (A05) are another way of referencing location. **Time** identifiers are also frequent in our examples, but not as frequent as location identifiers (contained in 16 vs. 20 stories; 61 vs. 144 occurrences). Depending on the data, time may be identified at various levels of granularity (e.g., day, month, year, decade, or even century). Time identifiers include fix dates (e.g., “on April 30” – A02), longer events (e.g., “Hurricane Katrina along the Gulf Coast in 2005” – A04), or time intervals (e.g., “since 1980” – A01, “from 2000–2016” – A04, “past decade” – A09, “1970s” – B01). Consecutive sequences of timely events may span across multiple sentences. For instance, “By Nov 8, [...] By mid-October, [...] As of Nov 26, [...]” – A05).

A specific type of insight **identifies** interesting data items as **outliers**, **extrema**, and **clusters**. We observe locations that are local or global **outliers**. The former compares a location with its neighbors, while the latter characterizes it with a much larger geographical region. For instance, A04 states a local

outlier as: *“Only two rural counties in the entire area that stretches from Mississippi across to Florida [...] even crack the list [...]”* A temporal outlier highlights unique temporal behavior: *“[f]or the first time in more than 50 years, the majority of America’s public school children are living in poverty”* (A11). An example of a geotemporal and global outlier in A10 is *“California has had more of these public mass shootings than any other state.”* Extrema correspond to the locations assuming the maximum or minimum values of a data variable. They are closely related to outliers. In most cases, outliers are extrema having specific importance with respect to a geotemporal variable. A cluster refers to a group of locations showing similar values for one or multiple data variables. Clusters include a list of two or more locations (*“North and South Dakota”* – A01) or refer to a higher level of grouping (e.g., *“Dakotas”* – A01, *“Midwest”* – A02). Clusters are described with the metric on the basis of which they are identified. For instance, *“counties with the lowest mortality rates, 18 out of 20 fall west of the Mississippi”* (A01) refer to a cluster of counties showing specific values of mortality rates.

Summarize insights report geographical variation, average (i.e., mean, median, or mode), or temporal variation. A geographic variation reports the varying value of a variable across a geographic region. For instance, *“[t]he South and West of the country [...] seen a big rise in the number of infections”* (B10). It mostly summarizes those variations that are peculiar. To describe the average, less technical words such as *“average values”*, or *“on average”* (e.g., *“[e]ach year, about 8,000 people will get that chance”* – A02) are widely used. Statistical terms like *“median”* or *“mean”* were also observed. It was surprising to see that some stories describe even the statistical significance: *“What is more, unemployment, while being statistically significant across the country, was not associated with the Le Pen vote in urban areas”* (A07). Temporal variations correspond to the reporting of a time series. We observed more instances of the reporting of peaks, nadirs, and steep inclination or declination, for instance, *“[...] demand for energy globally has fallen off a cliff”* (B01). Long-term trends are also noted, like *“[...] trend in demand has been downhill ever since”* (B01). Portions of a time series are compared with other portions, specifically, the ones that are recurrent and show seasonal patterns: *“[t]his compares with 73% last week and a peak of 85% between 3 April and 13 April 2020”* (B06). A summary of the temporal variations may be presented as a single sentence: *“the situation got really bad in late March but by May, cases were declining and most states had begun to ease restrictions put into place to halt the spread of the virus”*–B10.

Compare insights deal with part-to-whole comparisons, report correlation, and rank. Part-to-whole insights refer to a proportion of a total (e.g., 20% of the counties). These proportions are reported as exact percentages (e.g., *“23.5 percent”* – A11) or rounded (e.g., *“more than half”* – B08, *“one-third”* – B06). While reporting a countable variable—for instance, the number of participants of a survey in B06—we observed the use of a reference of ten (e.g., *“4 in 10”* to describe 41 percent of participants). The use of quantifiers like *“vast majority of the counties”* (A01) is another way of describing proportions without giving exact numbers. More than half of the part-to-whole comparisons are in B06—it communicates the results of a survey to gauge the social impact of COVID-19 in Great Britain. The correlation insights refer to the reporting of relationships between multiple variables. They include descriptions of positive or negative relationships and discuss causality. For instance, A07 discusses the impact of various socio-economic parameters (e.g., education, income) that played a role in French elections. It goes beyond comparing two variables and discusses intersection

effects: “[w]hile areas with higher median annual income were more likely to vote disproportionately for the centrist candidate, the effect of income is negated when education is taken into account.” Moreover, rank insights report the order of data entities with respect to a variable, for instance, “Brazil reported more than 32,000 new cases on Wednesday, the most in the world, and the United States was second [...]” (Bo3). These insights may not always reflect the numeric ranks but may also use comparative words, for instance, “[...] black workers seem to be struggling far more than white or Hispanic workers” (Ao4). The relative ranking is another way of comparing objects, for instance: “[a]fter Appalachia, the region that features most heavily is the Dakotas” (Ao1).

A considerably large portion of the textual narrative integrates different types of embedding (see Figure 3.3). A part of this embedding is the sentences that structure the story. All stories begin with a title (a type of heading; here, colored differently as black is later used to better discern sections in Figure 3.4). In 11 stories, the title serves as the main driving question of the story (e.g., Ao2, Ao4, Bo4). Five of the stories have a title that conveys the main takeaway (Ao2, Ao3, Ao7, Ao8, Bo8). Thirteen stories also contain additional driving questions (25 in total and 20/25 for stories of Collection A) at various positions in the narrative. Transitional sentences or headings are a way to switch between different topics.

Context is another form of embedding and provides additional information and opinion. All stories include a background that may help readers better understand the story and data. For instance, before reporting how the organ donation system works, Ao2 first describes the causes and symptoms of liver cancer. In rather technical stories like Ao2 or Ao3, the specific technical terminology and other related concepts are explained as domain knowledge. For instance, Ao3 uses a third of the narrative to explain the concepts of production and audibility of seismic waves. The technical terminology is explained in straightforward language, for instance, “[w]hen researchers track seismic activity, they’re sensing the waves that make the Earth roll and rumble” – Ao3. Stories in Collection B describe the impact of COVID-19 where only a few sentences introduce domain knowledge. Dataset descriptions include information on who gathered the data, how it was collected, and whether it was preprocessed or filtered for a specific reason (e.g., “[a]reas with very low populations were removed to limit their potential to skew the analysis” – Ao7). Almost 80% (18/22) of the stories include direct (40) and indirect (100) quotes. We observed two main sources of these quotes. One source is researchers who worked on the problem and gathered the data (e.g., in Ao3, Ao8, A12, Bo5). In such quotes, they share the methodology, insights, eureka moments of their research, or describe the findings. The second source of the quotes is the policymakers (e.g., in Ao2, B10). These quotes included their opinions or implications. Eleven of the stories include external references, for instance, to the full dataset, a research paper, or another story. Interpretations connect insights with historical facts: “American Indian populations have historically suffered from poor health outcomes and challenges in health care access, contributing to high mortality rates.” (Ao1). Or they infer and deduce other insights: “[i]f you’re a New Yorker, that doesn’t seem very fair” (Ao2). Authors also attach their personal judgment: “[o]rgan donation is good and kind, but it isn’t fair” (Ao2).

3.3.2 Visual Communication (RQ 1.2)

While the textual narrative explicitly explains the analysis insights, visualizations complement the text by showing relevant data. In our collection, 45 visualizations offer interactive exploration capabilities in 13 out of 22 stories. We found, that unless annotations are made, it stays up to the reader to find insights. Still, the authors of a story select a certain way to visually communicate the data. The **visualization** category in Figure 3.3 shows the codes regarding the **type**, **purpose**, and **exploration** of the visualizations, as well as whether they carry a **legend** or **visual annotation (properties)**. In our collection of 118 visualizations, we identified 8 distinct types of visualizations and 4 main modes of exploration.

First, we try to identify for what main **purpose** a visualization was included in the story. Although we do not know the original intentions of the authors, we were able to roughly categorize the visualizations into an **overview**, **detail** (with respect to certain aspects), and **comparison** visualizations. One visualization can share two or more purposes, for instance, to provide an overview as well as to facilitate comparison. We do not discuss the purposes separately but mixed with the following discussion of visualization **types**, as both coding subcategories interact.

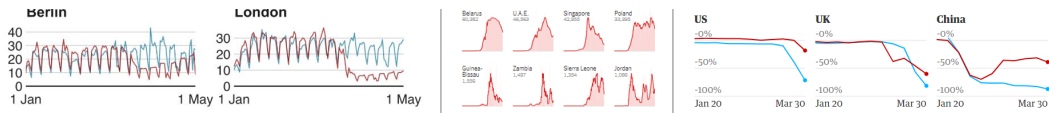
We observe that every story includes an **overview** visualization as the first visual data representation. **Map** visualization is a straightforward way of providing an overview of geographical data, which more than half of the stories (13/22) contain as the first visualization. We classify these *maps* as statistical (31) and geographical maps (5). Statistical maps are either thematic maps encoding data as colored regions (18)—also known as choropleths—or encode data in glyphs (e.g., circles, rectangles, or other markers) overlaid on the map (13). Geographical maps, on the other hand, do not encode any additional data. Satellite images or a street view are examples of such maps. Maps, particularly choropleths are mostly restrictive to a single variable and may not allow for comparisons across multiple variables. However, multiple versions of choropleth maps (5 in Collection A, 2 in Collection B) placed next to each other (or side by side) allow for **comparisons** of multiple variables.

Tabular visualizations (13) provide both *comparison* and *overview*. All tables in our collection either use visual encoding—as font color or cell backgrounds—or embed micro visualizations. Often, they communicate variation or uncertainty (e.g., distribution) in addition to, for instance, sum or average values. See two such tables from A01 below:

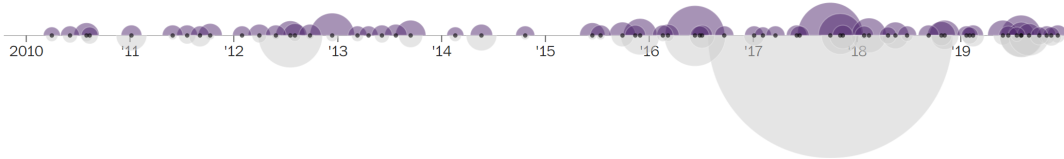
COUNTY	STATE	1.2k	1.3	1.4	1.5	1.6	1.7	1.8
Union	Florida							
Buffalo	South Dakota							
Oglala Lakota	South Dakota							

CAUSE OF DEATH	1980	TREND	2014
Cardiovascular diseases	507.4		252.7
Cancers	240.2		192.0
Neurological diseases	80.3		95.4

Besides the overview and comparison of aggregated geographical data, another aspect is the communication of geotemporal variations. Animating the map visualization is one way of accomplishing it; we observed five such instances. In tables, micro line plots show the temporal variations of geographic entities that have been arranged in rows of the table (see the right table above). Beyond maps and tables, small multiples are another way of providing geotemporal overview and comparison. We mostly observed the use of **line** and **area** charts in small multiples. For instance, three such examples are shown below (taken from B01, B03, and B07 respectively):

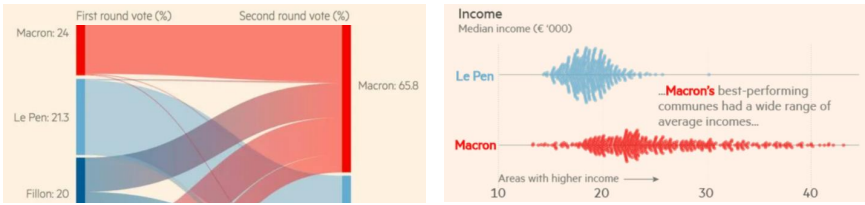


Including a time series next to a map visualization is yet another way to simultaneously communicate both geographical and temporal aspects. In such cases, the map displays the aggregated values for a certain time span, while the line plot shows temporal variations across that time span. Multiline plots (e.g., Bo2-V4 in Figure 3.4) can also provide comparisons across geotemporal data. Each geographic region (e.g., a city, state, or country) is denoted by a separate line and a specific region can be highlighted—on hover—to allow comparisons with all other regions (Bo2). We also observe the use of a rather non-standard (overlapped) area plot for showing a temporal overview (A10); the below timeline visualization shows the lives lost during various mass shootings in the US. Purple semicircles denote the number of people killed compared to the ones injured, shown as light gray semicircles.



Bar plots offer comparisons across different categorical variables and include simple bar plots (6), group bar charts (2), and stacked bar charts (8). Stacked bar charts can provide part-to-whole comparisons as well. For instance, Bo6 uses many bar charts to report the results of a survey on the social impact of the COVID-19 pandemic in Great Britain.

The detail visualizations go deeper with respect to certain aspects of the data analysis. In our collection, we observe the use of point plots (e.g., scatter plots), distribution plots, and diagrams. Distribution plots are limited to univariate data and include histograms (15), dot plots (2), and range plots (3). Comparatively, many more detail visualizations are observed in A05, A07, and Bo6. For instance, A07 reports the French presidential election results; the story begins with a spatial overview and comparison of votes for both candidates (one choropleth for each of the candidates placed side by side). The story, then, discusses various predictors that played a role in the election. The Sanky diagram illustrates the shift of allegiances of voters between the first and second rounds of the election. Similarly, Beeswarm distribution—a type of dot—plot compares the distribution of voters for the candidates across multiple social parameters (e.g., education, income, etc.).



Furthermore, scatter plots with trend lines show the correlation of votes with respect to the education level and income of voters.

We observed the use of infographics in some visualizations, especially in A10 and Bo1. A10 uses gun icons to give an impression of the kind of weapons used in mass shootings. Similarly, avatars of

1,204 victims and 183 shooters visually communicate their age (e.g., child or adult) and gender; users can hover to get details about each victim or shooter. Similarly, flags of two countries (the US and France) serve as intuitive labels in a comparison area plot in Bo1.

About a third of the visualizations (44/118) allows for interactive **exploration**. The simplest form of exploration is to offer details-on-demand as a **tooltip** (16/44). Eight visualizations (all maps) offer **multiple scale zooming** allowing readers to explore the data at various levels of geographical granularity; for instance, first provide an overview on the state level and then the city or county level. Almost half of the interactive visualizations (20/45) offer a **data selection control**. It lets readers choose a data dimension of their interest. The visualizations in Ao6, A10, and Bo5 are attached to a single central data selection control. While Bo5 just highlights the selected data object (e.g., a city) in all linked tabular visualizations, Ao6 and A10 include multiple views showing different aspects of the data. Five visualizations (all maps) include a **time slider** to play or pause an animation.

3.4 Results: Interplay of Text and Visualization (RQ 2)

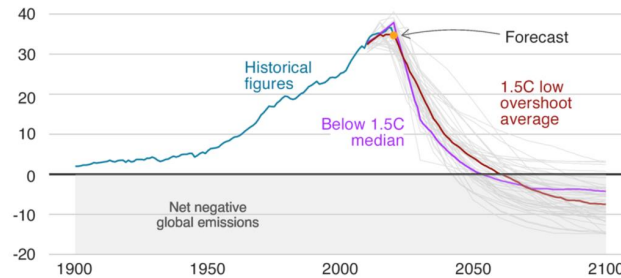
Based on the ingredients discussed in the previous section, we can now study the interplay between visualizations and text, more specifically, the various ways of linking the textual and visual representation as well as their joint organization in one story.

3.4.1 Linking the Two Media (RQ 2.1)

Links between visualization and text can be explicit or implicit. This section focuses on reporting the explicit links that can be unambiguously identified. We also noticed various ways of implicit links during our analysis; for instance, just referencing the same identifier or any data insights from the visualization and the text creates such implicit links. However, they were not explored as part of RQ 2; the next section (Section 3.5) studies these implicit links in detail. Moreover, by positioning a visualization close to the related text, the two are likely perceived as belonging together (the positional interplay of the two media is discussed in more detail for RQ 2.2). With respect to the explicit links, we discern two subcategories of codes as described in the following and summarized in Figure 3.3.

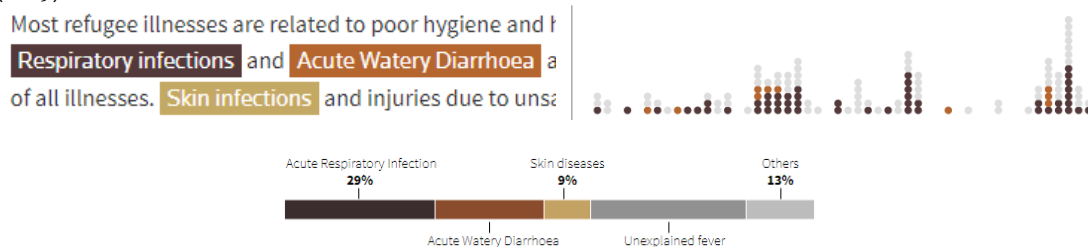
First, **text-in-vis** linking blends in textual content inside a visualization and includes **captions** (also comprising visualization titles), **annotations**, and **tooltips**. Almost 86% of the visualizations in our collection include a descriptive caption. The length of a caption may vary with the complexity of a visualization. We also observed that captions are more expressive in complex and non-standard visualizations, for example, the Sankey diagram, and the Beeswarm plot in Ao7. In 26 visualizations, captions communicate the main insight or takeaway from the visual. Ten of these 26 visualizations belong to Ao7. An example of a caption describing the main takeaway in a choropleth map (Ao4) reads: “[m]any rural counties are doing OK”, followed by a subcaption “[p]ercentage change in per capita personal income, 2000–2016” which explains what data is displayed on the map. In most of the stories that begin with an *interactive* overview visualization (e.g., Ao1, Ao8, Ao9, Bo2), the title of the story also serves as the caption of the first visualization, thereby serving as a connection between the two media. **Textual annotations** are another way of blending textual explanations or labels

in a visualization. They may include data labels—labels of states in a choropleth map or dots in a scatterplot—in 45/118 visualizations) or explanations (in 10/118 visualizations). While most of the annotated points or regions are picked up and explained in the textual narrative, a few stories include longer explanations inside the visualization (A04, A07, A09). For instance, textual annotations may explain every region of the chart (B01):



Almost half (46%) of the visualizations in our collection contain some variant of a textual annotation. Tooltips are another way of incorporating short on-demand textual explanations for interactive visualizations. One choropleth in A11 offers a tooltip that is always activated, and it gets updated on the selection of regions.

Second, text-to-vis linking references visualizations as the users read through the text. Before reporting insights, visualizations are often first introduced in the textual narrative (visualization introduction). This part of the narrative may include an explanation of visual encoding (e.g., “[t]he red, blue, black and white colors reflect the cheap plastic sheeting available to make shelters at the time” – A05) or a certain specificity of a visualization that is not obvious (e.g., “map is drawn to maximize the number of districts that usually vote Republican [...]” – A06). We observed fewer introductory sentences for visualizations in Collection B. It may be because visualizations are mostly standard and relate to rather well-known COVID-19 data. Visualizations in our collection did not carry identifiers, so they may not be referenced like in a scientific document (e.g., “Figure X”). Instead, they are cross-referenced by the name of the visualization (e.g., “see the scatter plot”) or by directional phrases (e.g., “the map below”) in case, there are multiple visualizations of the same type close by. We observed 36 instances of named or directional cross-references. We also observed color-linking in two stories (A05, A10). Various parts of the textual narrative are formatted (e.g., font colors or colored highlighting) to match and connect them with visual marks on the visualization. One such example is shown below (A05):



Hovering over these text blocks highlights the relevant segments of the charts. The 5 instances of color linking, we observed, were all interactive.

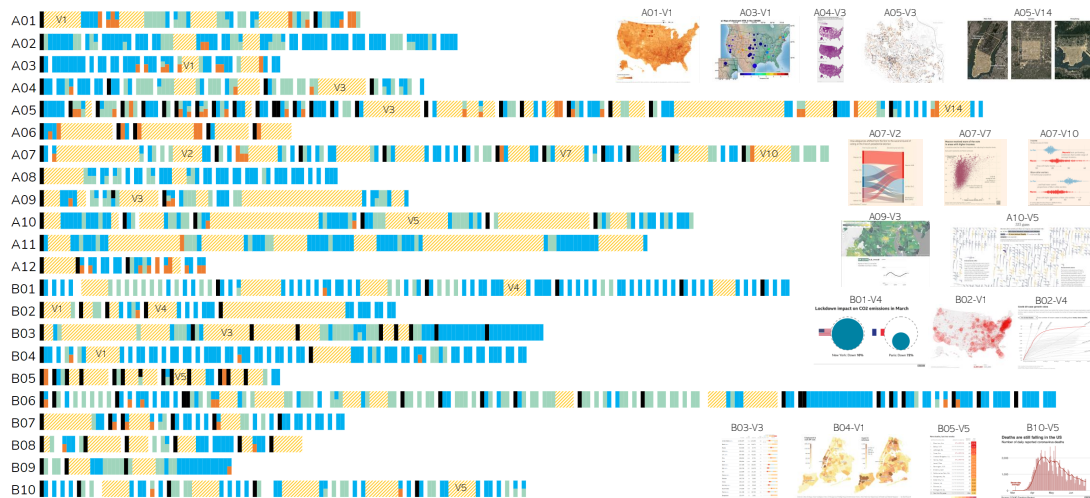


Figure 3.4: Flow and structure of stories. Each story is represented by a series of rectangles encoding the type of sentences (*heading*, *data-driven*, *embedding*, and *visualization-text linking*) and *visualizations*. The width of each rectangle encodes the size of a sentence (word count) or a visualization (estimated word count equivalent). White gaps indicate paragraph spacing. Rectangles are vertically (equally) divided in case a sentence has multiple codes assigned to it. The thumbnails on the right show 17 visualizations from our sample collection.

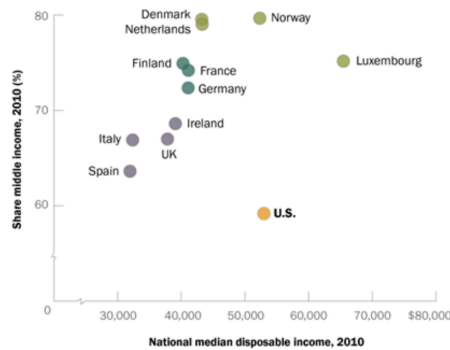
3.4.2 Embedding of Visualizations into the Narration (RQ 2.2)

Visualizations are embedded at various points in the story. Figure 3.4 shows the flow (left to right) and the structure of the stories in our collection. Every rectangle corresponds to either a sentence or a visualization and is scaled according to the space it consumes. To get a comparable scale for space consumption across both representations, we converted the sizes of visualizations (in pixels) to the number of words that would fit in the same space. We use a web browser’s developer tools to inspect the sizes of paragraphs and visualizations. Dividing the pixels of a paragraph by the word count of that paragraph resulted in pixel density per word. We averaged this pixel density across all stories, resulting in a value of 1,469.57. We computed the word count for each visualization by dividing the size of the visualization by the average pixel density. This provided us with an estimate to analyze the spatial importance and arrangement of content across the two media. Since our mapping is a rough estimate—diverse font styles, editorial guidelines, and story genres were not accounted for—we have only used it to do a coarse-grained analysis and refrained from inferring fine-grained patterns.

The proportion of textual narrative varies from 8% in B03 to 76% in A02 (Figure 3.4). We classify all stories into three groups according to the varying proportion of text and visualizations. Fourteen stories are *visualization-dominant* where visualizations occupy more than 60% of the total content. Five stories (A02, A03, A04, B06, B08) are *text-dominant* and include more than 60 percent of textual content. Only three stories (A01, A04, B01) are *balanced* as they contain textual content in the range of 40–60%.

Figure 3.4 allows us to study the arrangement and sequence of content. All stories begin with a

Share of middle-income adults rises with national household income; U.S. appears an exception



... One group is comprised of Spain, Italy, the UK and Ireland. The national median disposable income in these four countries ranged from \$30,000 to \$39,000 in 2010 and the middle-class shares ranged from 64% to 69%, ...

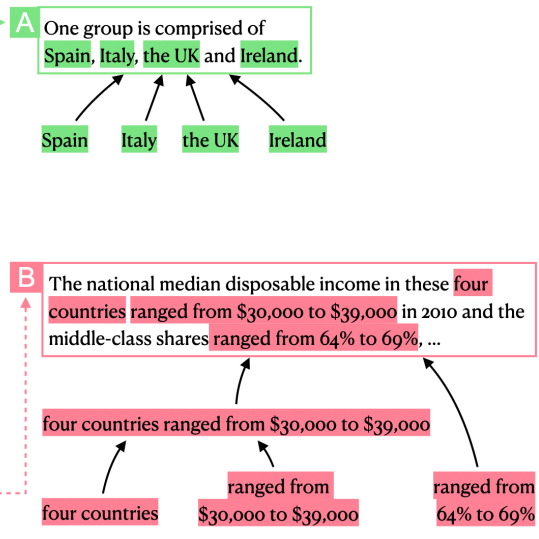


Figure 3.5: Excerpt of an article from Pew Research⁸⁵. Demonstration of text–chart reference grouping: The colored texts are minimal references, while A and B show how they can be grouped from bottom to top.

title (heading) and are mostly (18 of 22) organized in multiple sections as indicated by further headings. As we can observe from the blank spaces in Figure 3.4, which map to the spacing between paragraphs, most stories also make use of paragraphs for further text structuring. However, the diversity is obvious—from no use of sections and paragraphs (except for text breaks for adding the visualizations) in A11 to fine-grained section structuring in A05 and mostly single-sentence paragraphs in B01.

Nine out of 22 stories include an overview visualization right below the title to begin the story. While six (A01, A08, A09, A12, B02, B07) of these contain a map as an opening visual—A1 and A12 have animated maps—, others include a line plot (B03) or a small dashboard (B08, containing two stacked bar charts). Overall, thirteen out of 22 stories have map as their first visualization. Detail and comparison visualizations usually appear after the overview visualization and are often placed in different sections of the story following a semantic grouping (A04, A07, B01, B05, B07, B09, B10). Figure 3.4 shows a few characteristic examples of detail and comparison visualizations for A04, A07, B01, B03, and B05 along with their positions in the stories.

3.5 Results: Implicit Referencing (RQ 3)

While the previous section (RQ 2) investigates explicit links between text and visualizations, the focus of this section (RQ 3) is on understanding various types of implicit links and how they relate to underlying data and visualizations.

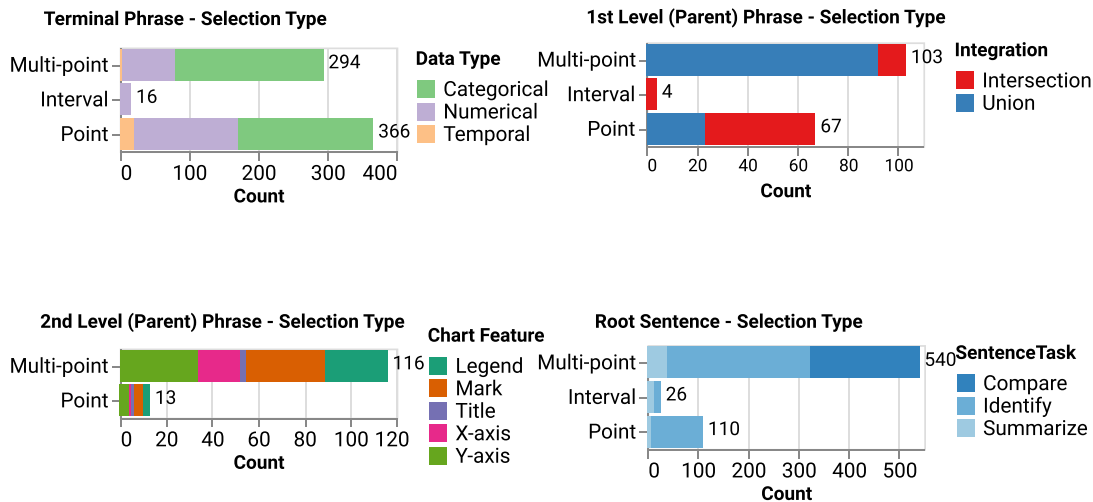


Figure 3.6: Overview of various kinds of implicit references in our sample collection. We observed point, multi-point, and interval selections in the text-chart references (e.g., a phrase describing one or more visual marks or a numerical range). They are often grouped together to generate aggregate selections, while also forming a hierarchical reference tree: *minimal* phrase \rightarrow 1st level parent phrase \rightarrow 2nd level parent phrase \rightarrow root sentence; see Figure 3.5 for an example.

3.5.1 Types of Implicit References (RQ 3.1)

A key insight is that every text-chart reference bears a similarity to a selection operation in a visualization. A visualization usually offers *point* (e.g., clicking one or more visual marks) and *interval* (e.g., brushing a region of visual marks) selections. Likewise, a text phrase can refer to one or more visual marks by directly mentioning item names or an interval of data points by stating the axis extents. Just to give an example, in Figure 3.5A, each country name in the phrase “Spain, Italy, the UK, and Ireland” refers to each corresponding visual mark in the scatter plot, while the phrase “ranged from \$30,000 to \$39,000” in Figure 3.5B refers to all visual marks falling in the numerical interval [\$30,000, \$39,000].

We observed 366 point, 294 multi-point, and 16 interval *minimal* references in a total of 676 references. Every implicit reference relates to some feature of a chart. In our sample collections, references were mainly associated with axes (283), individual marks (262), and legends (109). Figure 3.6 (left) shows the distribution of various kinds of references and what chart features they were connected to. References to axes and legends also ultimately lead to a selection of a set of visual marks in the chart area (cf. Figure 3.5B). The underlying data type of the referenced chart features was mostly categorical (407). Comparatively, fewer numerical (244) and temporal (25) data values were referenced. This trend holds for point and multi-point references—194 categorical, 152 numerical and 20 temporal in point references, against 213 categorical, 77 numerical and 4 temporal in multi-point references. Sixteen interval references refer to 15 numerical and 1 temporal data type. Interestingly, we observed only 4 (out of 676) visual references such as “red arrow” or an “orange slice”.

It was found that minimal references can be grouped together to create higher-order references. This grouping resembles the syntactic parse tree of a sentence. In such a parse tree, leaves—a word or a phrase—are terminal nodes that serve as independent units e.g., verbs, nouns, pronouns etc. Likewise, a minimal reference is a word or text phrase in our reference tree that establishes an independent connection to a chart feature. For example, in Figure 3.5A, each country name is a minimal reference and can be combined with other (minimal) references to construct a parent (higher-order) reference of up to four countries. Eventually, the whole sentence lies at the root of the reference tree, referring to all four countries (Figure 3.5A). In contrast, the second sentence in Figure 3.5 has three minimal references: “four countries”, “ranged from \$30,000 to \$39,000”, and “ranged from 64% to 69%”. The first two can be integrated to create a parent reference. This parent reference can then be combined with the third minimal reference “ranged from 64% to 69%” resulting in the root sentence (Figure 3.5B). When references are grouped, the leftover text—which is not part of any minimal reference—should be added to the parent reference, which otherwise would become fragmented. This is the reason why the final sentence (root reference) contains all text rather than only the minimal references.

In our sample collection, we observed up to four levels of references, with the last level (root reference) being the sentence itself. There were 175 first-level and 28 second-level ancestor references, with an average of 2.48 and 2.68 minimal references respectively. While 131 sentences have at least first-level references, only 27 (out of 227) sentences have up to second-level references. These statistics roughly reveal how reference trees can be constructed. For higher-order references, we observed more multi-point references than point references—103 versus 67 at the first-level and 25 versus 6 at the second-level. We only observed 4 interval references at the first-level and no interval coreference at the second-level; this is partly due to the fact that we code it as *point* or *multi-point* when an *interval* reference was combined with either of them.

3.5.2 Relation of References to Data and Visualization (RQ 3.2)

As discussed earlier, references can be grouped with other references. This grouping incurs data transformation; when references are grouped, the associated visual marks either undergo a *union* or *intersection* operation. To give an example, we refer to Figure 3.5B. When the phrases “ranged from \$30,000 to \$39,000” and “ranged from 64% to 69%” are combined, the resulting higher-order reference corresponds to the intersection of the two intervals (i.e., the points falling in the intersection). On the other hand, the grouping of “Spain”, “Italy”, “UK”, and “Ireland” goes through a union operation of the corresponding four visual marks. A reference grouping incurring a union operation indicates that each minimal reference is independent of the other, while an intersection grouping means that one reference constrains the other like a filtering transformation. We observed more union than intersection operations (i.e., 115 unions versus 59 intersections at the first-level parent). The variation increases as we climb up the reference hierarchy: 27 versus 4 at the second-level and 25 versus 1 at the third-level. This is not surprising as more text phrases would increase the chances of mentioning additional visual marks in the chart rather than narrowing down the selection.

It was observed that references (minimal or higher-order) closely relate to visualization tasks. All minimal references correspond to *identification* task. For instance, country names in the sentences

basically identify a country denoted by a point (visual mark) in the scatterplot (Figure 3.5A). In our sample collection, the minimal references identified mostly point (361) and multi-point (315) selections. Likewise, higher-order references also involve advanced visualization tasks in addition to identification tasks. The references at first-level correspond to *comparison* (33), *summarization* (12), and *identification* (130) tasks. Similarly, references at second-level relates to *identification* (20) and *comparison* (11) tasks. Finally, at the root (sentence) level we observed 149 *identification*, 49 *comparison*, and 29 *summarization* tasks. The visualization tasks at a higher-level generally include more than a single or multi-point selection; for instance, these may describe an extreme data point, compare multiple points, or summarize a range of data points. A concrete example of a *comparison* task looks like: “[s]ome upper-middle-income countries, like the Dominican Republic and Thailand, seem to have deadlier roads than much poorer places such as Liberia.”; the sentence compares deaths caused in road accidents for three countries based on a scatterplot visualization. Similarly, an example of a sentence relating to the *summarization* task is “[...] when countries reach a GDP [...] of about \$30,000, death rates usually start to come down.”

It was observed that almost half (311 out of 676) of the reference phrases were exact matches to the labels in the associated visualizations. Moreover, eighty-two references were partial matches. For instance, the phrase “\$1 increase” in a sentence against “assuming a \$1 increase in the minimum wage” in the chart label.

We also observed ambiguities in the text–chart references that could make the automatic identification challenging: references included inferences (64)—e.g., “former communist states” in a sentence while the chart shows explicit names of those states. The inference variation was common among references that were inferred from the visual encoding alone, as the corresponding charts did not contain textual labels. Other variations include synonyms (64), stemming/lemmatization (29), and abbreviations (8).

For the text phrases referring to the numerical intervals, we frequently observed approximated or rounded-off numbers (21)—similar to abbreviations—especially for large numbers or numbers with decimal points. Text phrases may also refer to derived statistical measures such as mean, variance, or other computed numbers (17)—e.g., “Nearly six-in-ten (58%) in the U.S.”, which requires a transformation of the underlying data to be identified. Charts sometimes contain annotations showing these measures, but often they do not. Therefore, it is a type of ambiguity that is equally challenging to resolve as other linguistic ambiguities while aiming for automatic extraction.

3.6 Study Limitations

A major limitation of our studies is the comparatively small sample size (22 stories for Study I and 77 articles for Study II) which may not be a complete representative sample. However, the examples were gathered from high-quality sources that were published quite recently (in the last 5 years) and, therefore, provide a better basis for observing the latest trends and patterns. However, we tried to counterbalance the small sample size by going down to individual sentences and visualizations and doing a much deeper analysis. As assigning descriptive codes as part of a qualitative analysis is always subjective, we tried to compensate this by redundant coding of two researchers (coders) followed by

joint discussions of potentially ambiguous and conflicting code assignments. The specific limitations corresponding to each study are described below:

Study I: Half of the stories (11/22) came from just two sources: the New York Times (NYT) and FiveThirtyEight (538). Therefore, the results may have been biased by their particular style of reporting. Although the diversity of the examples in Collection A of the sample is broad, it may not still cover the full spectrum of possible design space. Since all the stories in our sample collection were published digitally (accessible via an internet browser), they may also be restricted by technological constraints that must be considered in order for them to be widely available (e.g., browser performance, cross-platform compatibility, the choice of visualizations).

Study II: Despite the fact that our sample collection is diverse and more expansive when compared to the dataset of Kong et al.⁸⁷, the number of samples (here, the text—chart pairings) is still small. Moreover, they are limited to simple charts and do not contain complex or multiple-coordinated visualizations. The nature of the analysis method is qualitative, and it requires large manual effort and, thus, is hard to scale to hundreds of samples; it could benefit from computational linguistic methods that may further discover semantic and structural insights into the space of text—chart references.

3.7 Implications

While the empirical results have reported detailed findings addressing the initial research questions, in the following, we highlight what can be learned from these studies from a broader perspective. This perspective considers the practical aspects of authoring our envisioned interactive data documents.

3.7.1 Integration of Visualization and Text

We discovered many ways of integrating text and visualizations in data-driven stories to make them compelling and interesting for a broader audience. Please note that here we go beyond our initial **visualization—text linking** code category and discuss the integration at a broader level in the light of results obtained in Study I and II.

The integration can broadly be classified into implicit and explicit linking. The first type of **explicit linking** is the positional linking. Almost all visualizations in our sample collection were placed very close to the text that describes or references them (RQ 2.2). The visualization put next to the text helps readers better understand the descriptions. Besides, it avoids unnecessary scrolling or other similar interactions for connecting the visual with the corresponding text. The positional linking also depends on the presentation style—referred to as “*genres*” by Segel and Heer¹⁵¹—of the story. For instance, in one scroll-down story (A05), an overview map visualization is placed as a background that keeps on updating while other detail visualizations and textual content blend in on top as the reader scrolls through the story. The textual elements that appear inside a visualization (RQ 2.1, **text-in-vis**) are yet another variation of explicit linking. **Captions**, **annotations**, and **tooltips** blend in textual explanations next to or on top of a visualization. We observed that 86% (91/118) of the visualizations include captions and, in about 29% (26/91) of the cases, these captions convey the main takeaway. These kinds of text elements can make the visualization self-explanatory and make specific insights

stand out. Longer explanatory annotations make it possible to even include non-standard visualization (e.g., Sanky diagrams, Beeswarm plots) in a story. Generally, informative captions can reduce the mental effort to process a data visualization.¹⁸⁰

It was observed in 77% (17/22) of stories that visualizations are explicitly referenced in the text (RQ 2.1, **text-to-vis**). This, of course, goes beyond just cross-referencing that is commonly used to refer to numbered figures in a conventional writing style. In data-driven stories, it is important to describe *what* is visualized and *how* it is visualized (e.g., explain non-intuitive encoding), especially if the visualization might not be familiar to every reader (visualization introduction). Consistent color linking is another, rather less frequent but interesting way of visualization–text linking. It corresponds to the use of consistent colors that can make the visualization-related parts of the text stand out.

Implicit linking (RQ 3.1) is another type of connection that binds mostly **data-driven** text to related visualizations. We refer to it as *implicit* because these links are present, but users discover them while reading through the text. These references can be converted to interactive links to facilitate the reading process. For instance, highlight associated visual marks of a visualization when hovering over the relevant text. Previous research has already shown the benefit of such visualization–text integration using specific examples.^{90,91,11,163,194} While these individual references would already help readers in quickly switching from textual to the visual representation of data, too many of those might overwhelm or even annoy the user. It is, therefore, desirable to group them into higher-level references at sentence or even paragraph levels. However, it might be challenging to use just the visual highlighting in the visualization for higher-level references (e.g., sentences describing comparisons or geographical variations). The linking can go even beyond visual highlighting and animate parts of a visualization that corresponds to a linked sentence or paragraph.^{90,99}

3.7.2 Automatic Generation of Interactive Data Documents

One of the main motivation to perform these empirical studies was to look into the possibilities of automatically generating parts of interactive data documents. In the following, we discuss the implications of our studies on the parts of the existing stories that can be realistically generated using automatic approaches.

In contrast to visualizations, where often the raw data can be visualized, textual content requires significant selection and prioritization. Some data-driven findings are straightforward to compute, for instance, extrema, clusters, and correlations (RQ 1.1). However, additional background on demography and geography is necessary to group those entities to form natural clusters for a human reader (see discussion on locations). Content prioritization may also be necessary because otherwise too many findings (e.g., a long list of entities as part of ranking) will be reported that might annoy the reader. Temporal and graphical variations are challenging but still realistic to generate. Having prioritized the content for presentation, it must be presented in natural language text. For instance, instead of reporting long lists of entities, the text should use collective or even vernacular names for geographic locations. As for text generation technologies⁴⁸, template-based approaches can be used for **data-driven** text. However, they may require larger manual efforts to consider all possible cases. On the other hand, machine learning approaches are more flexible as they work with an underlying

grammar model but are harder to control and test. However, it may be challenging to seamlessly interlace (**data-driven**) with the text that provides **context** as observed for the examples in our sample collection (RQ 2.2). Hence, an automatic solution may clearly discern between the different types of textual explanations, for instance, *data-driven explanations*, *educational explanations*, and *methodological explanations*¹²⁵ to be transparent (more details in Chapter 5).

For the automatic generation, it might be easier to just focus on visualization-dominated data documents (RQ 2.2) instead of creating a complex textual narrative and embedding short **data-driven** text in visualizations or **vis-in-text** elements. We believe that automatically generated data-driven content would profit from additional interactions to link the text and visualization (RQ 3). As discussed in the findings of RQ 3.2, implicit references closely relate to visualization and underlying data. Almost half of these references in the text exactly match to features of a visualization (RQ 3.2). Such references can be identified using simple keyword matching. For other variations like synonyms, abbreviations, and semantically related words (e.g., Barack Obama, democrat), advanced natural language processing techniques such as word2vec models^{119,120} can be employed. Since the grouping of references follows a similar structure as the syntactic parse tree of a text, we can leverage linguistic semantics to create meaningful higher-level references; it is a challenging problem, though. More details on this can be found in Chapter 6.

Automatically generating text snippets, identifying, and suggesting potential references between a given text and corresponding visualization can facilitate the authors of interactive data documents. It can speed up the creation process and enable journalists and designers to create interactive documents without worrying about the nitty-gritty details of programming. However, every automatic generation approach is prone to errors. Therefore, the goal should not be to replace a human author but to facilitate them; one way is to combine automation with a user interface that would allow authors to quickly fix problems—resulting from automation—leading to a mixed-initiative interface. Before discussing such an authoring solution (Chapter 6), the next chapter presents a generic approach to fully automate the generation of interactive data documents and instantiate the approach for two different application domains.

4

Generation of an Integrated Textual and Visual Representation

While visualizations are easily generalizable to multiple datasets—for example, a scatterplot can always show the relationship between two quantitative variables irrespective of the underlying context—, textual explanations on the other hand are highly domain-dependent. A sentence describing a scatterplot may change altogether if the data changes, and even moreso when the context changes too (e.g., from describing economic characteristics of countries to pollutants causing air pollution). Hence, human-written explanations do not suffice to ply such adaptability. It would be tedious to manually author explanations for every possible scenario. For example, generating a profile page for every researcher working in the visualization community based on bibliographic data would require a large manual effort. Alternatively, automatic text generation is more adaptable to changing data needs. Our approach relies on natural language generation to create text for describing data-driven insights. This chapter argues for automatically detecting, prioritizing, and then summarizing analysis insights as textual explanations and embedding these explanations in a visualization system according to the visualization–text linking concepts discussed in Chapter 3. It presents two interactive systems for bibliography and bivariate geographical data respectively to make data analysis accessible. The novelty of the approach lies in the joint and integrated generation of natural language text and visualization.

4.1 Automatic Generation Process

Interactive data documents leverage a joint representation of data comprising natural language text and visualization to make the analysis process accessible and explicit. Notable analysis insights can be

identified and summarized as textual explanations with explicit connections to visualizations. This section describes an end-to-end automatic solution to produce an interactive data document from raw input data. It walks through the generic generation process irrespective of a particular application before its instantiation to two specific application scenarios.

4.1.1 Content Determination and Prioritization

Any approach involving natural language generation begins with content determination, which basically decides what information would be conveyed. For visualizations, often the raw data can be shown, but the textual content requires substantial selection and prioritization. Although Section 3.7.2 reveals analysis insights and findings that are usually presented as text in a data-driven story, these findings are grounded in a very specialized type of stories, i.e., the ones describing geographical data. As a matter of fact, the content determination process greatly depends on the domain application and target audience; it is hard to generalize beyond a specific application. Therefore, for a specific application, the approach should first derive information needs. It can be done either by conducting a requirement analysis from an end user’s perspective (Section 4.2) or an explorative study with domain experts (Section 4.3). Having derived the information needs, suitable visualizations (including word-sized visualizations) should be chosen along with the determination of analysis insights that are to be communicated through textual explanations. To fulfill the information needs, various types of analyses can be performed, oftentimes leading to many results (or insights). It may not be possible to realistically communicate all these results to the user. Doing so would make textual explanations unnecessarily lengthy.

Thus, we need to prioritize the findings, for instance, just listing the *most prominent* clusters instead of all. Mostly these findings can be considered as lists of data items (e.g., co-authors of a researcher, cities that suffer deaths as a consequence of storms) that need to be sliced to a reasonable length for presentation in the text. The underlying assumption here is that every item has an importance value attached, and hence, can be represented as a numeric value. For instance, in a list of co-authors (data items), the number of joint publications can be considered as the importance value. Similarly, in a list of cities, the number of casualties might be the importance value. The selection problem may sound trivial—one could just select the top x items. However, if we as human authors of a text would select a number of important items, we do not restrict ourselves to a fixed number, but choose a good cut-off point dynamically. We try to avoid that the list grows too long, but we also do not cut off at a position where the distance to the next item is small. Mathematically, the problem reduces to selecting a to b items from a sorted list $L = (l_i)_{i=1}^n$ of n numeric items ($n > b$, $l_i \geq l_{i+1}$). We select cut-off index $c \in [a, b]$ where the difference of list elements $l_{c+1} - l_c$ is maximal. However, there can be several maximal differences—in that case, we pick the smallest index, formally:

$$c = \min \left(\arg \max_{k \in [a, b+1]} l_{k+1} - l_k \right) \quad (4.1)$$

Finally, the list is cut after element l_c and hence only contains the top c elements. In the following, we refer to this procedure as Equation 4.1.

4.1.2 Text Generation

The next step is to convert the identified and prioritized list of analysis insights into natural language text. To accomplish this, various types of generation approaches can be used.⁴⁸ Some approaches use artificial intelligence and are fully automated (e.g., Generative Pre-trained Transformer 3²⁴). These approaches, however, require substantially large training datasets and are harder to control and, consequently, difficult to integrate with the visualizations as their results cannot always be predictable. Another type of approaches are template-based; they work with well-formed, pre-written phrases with gaps in them and produce the output text when these gaps are filled with data. For instance, an announcement generation system at a train station can be considered a very simple example of a template-based system; the template “[*train*] will leave for [*city*] at [*time*],” where the gaps [*train*], [*city*], and [*time*] are filled from a data table might produce: “ICE577 will leave for Frankfurt at 13:30.” Template serves as the foundation of such text generation systems.

In contrast to the advanced text generation approaches, we use template-based text generation because of its good applicability and sufficient flexibility. Another crucial aspect is predictability, which is of great importance for our use case as we want to closely integrate generated text with a visual representation of data. Commercial tools such as *Wordsmith*^{*} or *Arria NLG Studio*[†], or libraries like SimpleNLG⁴⁹ allow for building customizable templates for text generation and use a grammar model to do the grammar-related tasks (e.g., subject-verb agreement, handling of singular/plural). However, integrating word-sized graphics and establishing interactive visualization-text linking with the output of such systems would require more effort. Consequently, we decided to implement our template-based generation approach as decision graphs (shown in Figure 4.3 and Figure 4.13); the approach is inspired by the work of Beck and others.¹⁷

Although there is a widespread impression that the template-based text approaches are not as well-founded as the advanced (also known as plan-based) approaches, Deemter et al.¹⁷⁶ contradict this notion by sketching a template-based approach that has the same level of theoretical well-foundedness and usefulness (including maintainability) as the advanced approaches. Template-based approaches are, sometimes, referred to as “*programs that simply manipulate character strings, in a way that uses little, if any, linguistic knowledge*”¹³⁷ However, our templates are better described as “*making extensive use of a mapping between semantic structures and representations of linguistic surface structure that contain gaps.*”¹⁷⁶

4.1.3 Document Integration

In contrast to other generation systems, a novel aspect of our approach is the coherent integration and linking of jointly generated text and graphics. In addition to the positional linking provided by the multiple-coordinated-views layout, the next step of integration is the embedding of word-sized graphics in the lines of text, which already visually connects the visualized data to the related text phrase. Moreover, the use of regular-sized visualizations, which are interactively linked to the text and

^{*}<https://automatedinsights.com/wordsmith>

[†]<https://www.arria.com>

word-sized graphics, allows exploration of data. The content is further connected by visualization–text interactions (it is discussed in more detail in Chapter 5); it describes the linking of text, word-sized graphics, and regular visualizations. For instance, clicking on linked text fragments highlights relevant parts of the visualization and shows related data in a *details* panel. (Interactive) word-sized visualizations are included as part of the text templates; they can also be considered as parameters (gaps) in the template. We further use info icons ⓘ to mark the availability of additional explanations and present this information on click.

The following two sections present two full-fledged systems, *VIS Author Profiles* and *Interactive Map Reports*, and demonstrate the application of our automatic generation approach to two different datasets and domains: namely bibliographic data and bivariate geographical data, respectively. Although both systems follow the same process, there are considerable differences in terms of layout, use of word-sized graphics, linking, and interactivity. These differences stem from the diverse nature of dataset for each application domain, as well as from the different focus—Interactive Map Reports emphasizes more on the generalization of the generation approach for different datasets (of the same type but different context). Considering the broad target audience, an important consideration is to make these systems more accessible and self-explaining. For this purpose, we describe the following set of design principles that should be applied while instantiating the approach to any application domain.

- **Prioritize findings and keep them short** – Since text is very explicit, the approach leverages natural language text to explain the data and analysis results. With the increasing number of insights, a danger with text, however, is that it can become lengthy. When the explanations would make the main text become lengthy, it is important to prioritize insights or findings in the order of their importance. Even in this case, the information should not be unavailable to users. The longer or additional information should be made available on demand.
- **Make algorithms and data transparent** – The approach, especially the natural language generation, should not sound like a black box. As it often relies on heuristics and other algorithms, it is important to provide background on our algorithms and make underlying data relevant to a context available on demand to allow users to validate the textual descriptions and build trust in the descriptions.
- **Better say nothing than say something wrong** – The explicit description in text holds the approach responsible for what it says; therefore, it is better to leave out descriptions if uncertainty is too high and focus on information that is certain. Of course, uncertainty decreases with a higher quantity of information (e.g., better data availability) or a higher quality of information (e.g., more reliable heuristics). The approach can calibrate the parameters or heuristics for when to omit or include a certain finding as part of an iterative fine-tuning process that will usually be followed throughout the generation process.

4.2 VIS Author Profiles: Bibliographic Data

Bibliographic data (publication records) contain rich information and can play an important role in assessing the expertise and experience of researchers, for instance, when hiring faculty members, forming a program committee, or finding potential collaborators. Existing digital library systems show relevant author-centric information, but only add little abstraction to raw publication records. For instance, they abstract publication metadata to co-author relationships but only provide them as a list rather than an explorable co-author visualization. Users have to go through different views and apply various filters to gather the required information. As an alternative, visualizations that show publication and author data have been suggested. Although existing visualizations provide high-level abstractions about author profiles, their focus is often narrower, or they grow difficult to read and complex when adding more information.

In contrast, VIS Author Profiles combines visualizations with natural language text and leverage the advantages of both representations. It describes a novel way of presenting publication records and related analysis results for scientific authors. VAP* is a Web-based interactive data document that generates author profiles in the form of interactive reports (Figure 4.1). The text describes general statistics, research topics, and collaboration networks. In addition, interactive visualizations allow for exploration of trends and extended collaboration relationships.

4.2.1 Analyzing Publication Records

We begin by discussing the application of analyzing publications records: In particular, we describe author types and identify scenarios for which such analysis is targeted, how to frame the scope, and what information an analyst requires.[†]

AUTHOR PERSONAS

VAP focuses on authors of scientific papers, a group of researchers ranging from (Master's or PhD) students who have just published their first paper to senior researchers or professors who may have already published hundreds of research papers. VAP aims at creating meaningful reports for all these researchers. To be specific, researchers (authors) can be categorized into distinct personas in the sense that each group reflects a certain role or stereotype:

- **Student:** A student (Bachelor, Master, or PhD level) who contributes to research projects within a study program or dependent employment under the supervision of experienced researchers.

*The interactive system is available online at: <https://mrshahidlatif.github.io/vis-author-profiles>

[†]The analysis of publication records for discovering information needs in response to the two scenarios—Recruiting and Identifying Experts—was performed in a close collaboration with Fabian Beck. In the early stages of development, while I was doing pre-processing of data and implementing the VAP system, we were continuously using it as a test bed to derive high level information needs. Furthermore, Fabian conducted the comparison and assessment of existing systems in how well they fulfil the information needs.



Figure 4.1: Profile of author *Ben Shneiderman*. The text consists of three sections describing general information, research areas, and collaboration relationships. The visualization below provides information on joint work with co-authors on a timeline. The sidebar shows details on demand, whereas the top-right bar chart displays the temporal distribution of publications. Badges at the top summarize achievements. The cut-outs on the right are two different versions of the sidebar (list of collaboration groups and similar authors).

- **Researcher:** A postdoctoral researcher, assistant professor, research scientist, or lecturer who is conducting their first independent research and might start supervising students.
- **Senior Researcher:** An associate or full professor, senior research scientist, or senior lecturer who can build on years of experience in research and supervision.
- **Occasional Contributor:** An outsider to the studied scientific community who occasionally contributes to academic work within the community.

Please note that every author cannot be unambiguously assigned to a certain persona. This list, however, assists in structuring the collection of authors and ensuring that our technique eventually generates relevant descriptions for researchers of varying degrees of expertise.

SCENARIOS

Publication records of researchers provide information on their research activities, topics they have worked on, and their collaboration network with other researchers. VAP focuses on the following two scenarios that build on publication records and require an author-centric view; other use cases like literature search or the analysis of a research field are beyond the scope of VAP.

S1 – Recruiting: Every hiring procedure revolves around evaluating a candidate’s suitability for the position at hand. In Academia, experience and accomplishments of a researcher closely relate to the candidate’s publications. Hence, in addition to a formal CV, the publication record of the candidate can provide rich information. Another scenario that can be considered a variant of this scenario is admission to a funding program.

S2 – Identifying Experts: Unlike recruiting, where specific persons apply, finding experts for an academic task or role is open to further suggestions. Besides comprehending the data of a single researcher, it is also required to explore similar researchers. Typical examples include looking for a reviewer for a research paper, selecting candidates for a program committee, or looking for possible collaborators or supervisors. In these cases, expertise with respect to certain research topics as well as experience regarding academic collaboration with other groups of researchers are key criteria to contact a potential researcher.

SCOPE AND DATA

Research articles, papers, and books are published in almost every field of science. However, the specific publication culture differs greatly between fields. Without knowing a research community, it is hard to understand the specifics of a publication culture that limits the process of summarizing a publication record in a useful way. Therefore, VAP focuses on publications from computer science, which is our own research area and where we have a decent understanding of the publication culture.

The comprehensive availability of publication data is a prerequisite for building a system to generate author profiles. DBLP* offers publication metadata. To investigate research topics, we needed keywords that authors assigned to their papers. However, this information is present neither in DBLP nor in any general data collection for all computer science. Consequently, we decided to focus on the *visualization* community, where the *Visualization Publication Data Collection*[†] provides such information. The generated author profiles in VAP are restricted to authors listed in this data set, with the integration of DBLP data to include publications of those authors that appear outside visualization venues. This narrower focus allows us to consider the specifics of the visualization community, for instance, its branching into *scientific visualization*, *information visualization*, and *visual analytics*. Figure 4.2 shows an overview of the data sources, preprocessing, and final dataset.

The final and curated dataset includes 5,086 authors and 128,961 publications till August 29, 2017. To retrieve research topics (often denoted by keywords), we enrich the data by categorizing the most frequent publication venues (i.e., journals, conferences, workshops, etc.) into research communities of computer science. A classification of 688 venues with 56 keywords provided a community assignment for 61,469 publications. We further enrich these high-level keywords with additional keywords extracted from paper titles based on a manually created mapping of 26 typical terms (e.g., *visualizing* → *visualization*). To identify subtopics within the visualization community, we leverage the author-assigned keywords. Since they are inconsistent, we use the mapping that the *KeyVis*[‡] project pro-

*<https://dblp.org/>

†<http://www.vispubdata.org>

‡<http://keyvis.org/>

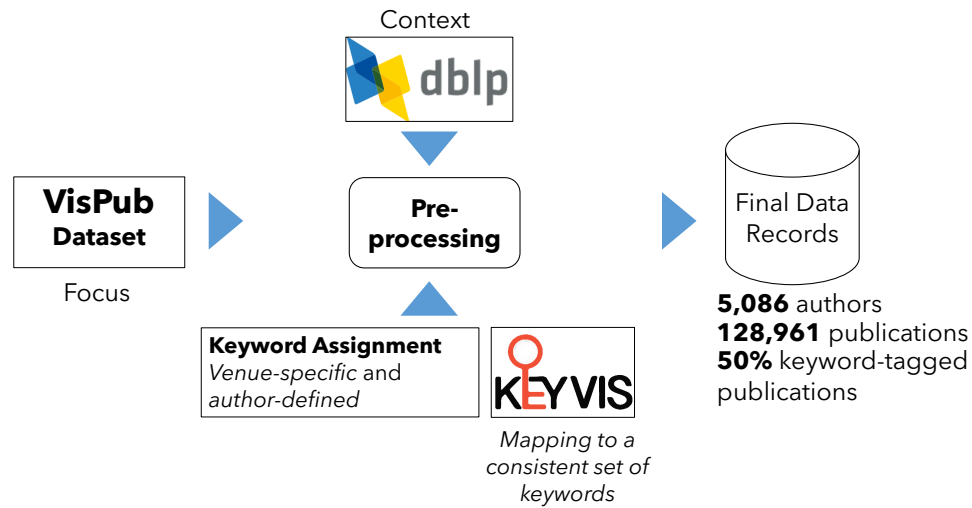


Figure 4.2: Scope and data pre-processing: VIS Author Profiles is restricted to generate profiles of authors available in the *VisPub* dataset. However, it retrieves other publications of these authors from *DBLP* data.

vides to map them to a standardized set of keywords. We select a subset of such aggregated keywords that—in our opinion—best reflects certain subareas of the visualization community and map these to simpler terms, preserving their original meaning as far as possible.

INFORMATION NEEDS

The next step is to derive information needs for supporting users regarding the aforementioned scenarios (S_1 and S_2). A list of publications can provide rich insights about the research topics of a researcher, as well as an overview of the researcher’s collaborators. Besides, publication statistics and temporal evolution of publication activity hint at experience levels and academic achievements.

Firstly, general publication statistics provide an idea of how actively a researcher is publishing, which could, often, be a central criterion for recruiting (S_1). In computer science (particularly the visualization community), both journals and conference proceedings can be considered premier publication venues. On average, however, journal articles have an estimated higher contribution because of their greater length and due to the fact that proceedings might also comprise short papers, posters, and workshop contributions. Hence, distinguishing between journal articles from proceedings papers hints at the quality of the potential contributions of the publications. The temporal distribution of publications indicates academic age and level of experience, which is not only relevant for recruiting (S_1) but also for finding an expert with sufficient experience (S_2). A special publication is the PhD thesis, where the author independently worked on a topic and achieved a first academic milestone. First-author publications can be considered particularly relevant when hiring early-career researchers (S_1), because these might best express the research interests and abilities of the author (assuming that, like commonly applied in practical computer science, the author sequence reflects contributions and

does not follow alphabetic order).

IN 1 – General Information

IN 1.1 *What is the **number of publications**, overall and discerned by publication type?*

IN 1.2 *What is the publication span and **temporal publication distribution**?*

IN 1.3 *When was the **PhD thesis** published, and what is the ratio of **first-author** publications (for early-career researchers)?*

Secondly, it is important to understand areas of experience on different levels of abstraction for selecting experts (S₂). Here, we discern between research communities (e.g., *visualization*), subfields (e.g., *information visualization*), and focus areas (e.g., *set visualization*). In addition, the temporal evolution of these research areas can be relevant for discerning a researcher's current and past research direction. Research topics can lead to other researchers who have similar expertise. This can be helpful in many cases. For instance, imagine a PhD candidate approached a researcher for the role of supervisor, but the researcher has declined or is unavailable, and now the student needs to find somebody else with a similar research focus and expertise. Frequent co-authors are excluded here, as they naturally share and work on similar research interests; we discuss them in the next information need.

IN 2 – Research Topics

IN 2.1 *What is the main **research community** and its connections to other research communities?*

IN 2.2 *What are **subfields** and focus areas within the main community?*

IN 2.3 *What is the **evolution of topics**?*

IN 2.4 *Who are other authors contributing to **similar topics** (excluding frequent co-authors)?*

Finally, relationships among co-author help to see how a researcher is connected to a research community. This information is important when looking for references of a candidate (S₁), gauging a candidate's influence, experience in supervising other researchers (S₁), or searching for similar experts (S₂). A special kind of relationship is the supervisor (often, one of the last authors) and the supervisee (often, the first author). This can be estimated from author sequence in publications. Former supervisees already supervising other researchers indicate a certain influence of the supervisors. Moreover, the co-authors may form noteworthy subgroups; for instance, an author might frequently publish on a certain topic with a specific subset of co-authors.

Table 4.1: Assessment of fulfillment of information needs in author profile pages of existing digital library systems; degree of fulfillment: no ○ ○ ○, partly ● ○ ○, largely ● ● ○, and yes ● ● ●.

Information Need (IN)	General Information			Research Topics				Collaboration Network			
	1.1	1.2	1.3	2.1	2.2	2.3	2.4	3.1	3.2	3.3	3.4
	number of publications	temporal pub. distribution	PhD and first-author	research community	subfields	evolution of topics	similar topics	main collaborators	supervisors	supervisees	subgroups
ACM Digital Library	● ● ○	● ○ ○	○ ○ ○	● ○ ○	● ● ○	○ ○ ○	○ ○ ○	● ○ ○	○ ○ ○	● ● ○	○ ○ ○
AMiner	● ○ ○	● ○ ○	○ ○ ○	● ● ●	● ● ●	● ● ●	● ● ●	● ● ○	● ● ●	● ● ●	○ ○ ○
DBLP	● ● ●	● ○ ○	● ● ○	● ● ○	○ ○ ○	○ ○ ○	○ ○ ○	● ● ○	○ ○ ○	○ ○ ○	● ● ○
Google Scholar	● ○ ○	● ○ ○	○ ○ ○	● ● ○	● ● ○	○ ○ ○	● ○ ○	● ● ○	○ ○ ○	○ ○ ○	○ ○ ○
ResearchGate	● ● ●	● ○ ○	● ● ○	● ● ●	● ● ●	○ ○ ○	● ○ ○	● ● ○	○ ○ ○	○ ○ ○	○ ○ ○
Scopus	● ○ ○	● ● ●	○ ○ ○	● ● ●	● ○ ○	○ ○ ○	○ ○ ○	● ● ○	○ ○ ○	○ ○ ○	○ ○ ○
Semantic Scholar	● ● ●	● ● ●	● ● ○	● ● ○	● ● ○	● ● ○	● ○ ○	● ● ●	● ○ ○	● ○ ○	● ○ ○

IN 3 – Collaboration Network

IN 3.1 *Who are the **main collaborators** and what is the temporal distribution of joint work?*

IN 3.2 *Who are or were **supervisors** and are the collaborations still ongoing?*

IN 3.3 *Who are or were **supervisees**, are the collaborations still ongoing, and are the former supervisees already supervising?*

IN 3.4 *What are **subgroups** of co-authors who have frequently worked together on certain topics?*

4.2.2 Existing Systems and Author Visualizations

Several digital library systems already allow for the exploration of publication records of scientific authors. This section evaluates these systems against the information needs and demonstrates that they do not yet sufficiently satisfy all needs. Likewise, author-centric visualizations provide an explorable representation of the same data but focus a lot more on the collaboration networks of researchers.

ASSESSMENT OF EXISTING DIGITAL LIBRARY SYSTEMS

The existing digital library systems can be broadly categorized into two types: (i) *publication-centric* systems for exploring scientific literature in general, and (ii) *author-centric* systems for exploring the research profile of a specific researcher. We focus our discussion on the latter systems because these are more closely related to the scope of VAP. We exclude systems that do not have a dedicated author profile page (e.g., *IEEE Xplore*).

Google Scholar is probably the most widely used system for online searching of scientific literature. It has a profile page for each (registered) author comprising information on affiliation, research topics, a list of publications, co-authors, and citations. Likewise, *Microsoft Academic* is a competing system and has a similar layout for author profiles. Unlike these two systems, the author profiles of *ACM Digital Library*, *DBLP*, *ResearchGate*, and *Semantic Scholar* employ faceted browsing¹⁸⁸ to subselect the research articles along certain facets including the publication type, research topic, co-author, or publication venue. *Google Scholar*, *Scopus*, and *Semantic Scholar* show citations and publication frequency on a timeline as a bar chart. *AMiner* provides other visualizations, including a stream graph of research topics, a Kiviat diagram of several publication metrics, and an ego-centric (simplified) co-author network. In addition to faceted article search, *Semantic Scholar* features an impact visualization to showcase researchers who influenced (and who were influenced by) a specific researcher.

We evaluate the existing systems with respect to how well, if at all, they already fulfill the information needs (Figure 4.2.1). To conduct an objective evaluation, we consider only the availability of features, and do not take into account their usability or data quality issues. This enables an unbiased and reproducible comparison while rating the quality of features would necessarily be considered subjective. Table 4.1 summarizes the outcomes of the evaluation; an extended interactive version of the table is part of the supplemental material of the corresponding paper⁹⁵; it provides brief explanations for every rating.

- **IN 1** – All the systems show a list of publications in chronological order. However, not every system distinguishes between publication types (**IN 1.1**) or shows the aggregated number of publications on a timeline (**IN 1.2**). While no system highlights publications of a researcher as the first author, PhD theses—though rarely contained in the data sets—can be retrieved in those systems that allow for filtering records by publication type (**IN 1.3**).
- **IN 2** – In most cases, research communities (**IN 2.1**) can be indirectly derived from aggregated venue information (e.g., *DBLP*) or through author-selected keywords (e.g., *Google Scholar*). In some systems, research communities can be directly identified as automatically mined subject areas (e.g., *Scopus*). Similarly, more detailed information on subfields is indirectly or directly available in most systems (**IN 2.2**). On the contrary, the analysis of the evolution of research topics is only supported in *AMiner* and *Semantic Scholar*; both systems provide a timeline (**IN 2.3**) of multiple research topics using a stream graph. Finally, *AMiner* is the only system that indicates a list of other researchers that are working on similar research topics (**IN 2.4**).
- **IN 3** – Most of the existing systems list frequent collaborators as list, but—except for *Semantic Scholar*—without temporal distribution of research publications (**IN 3.1**) as a result of these collaborations. Unlike other systems, *AMiner* is the only system that explicitly highlights supervisors and supervisees of an author (**IN 3.2** and **IN 3.3**). None of the existing systems—except for *DBLP*—considerably supports the identification of frequent collaboration groups of an author (**IN 3.4**).

To summarize, on the one hand, there are systems like *DBLP* and *Google Scholar* that have an easy-to-use interface but they are restricted to only fulfill a smaller fraction of the information needs. On the other hand, systems like *AMiner* and *Semantic Scholar* already satisfy a good proportion of the information needs. Although these systems can be extended toward fulfilling the remaining information needs, they are already overloaded with many views including lists, tables, and visualizations. The addition of further information would clutter their display and consequently make them overwhelming. *VIS Author Profiles* suggests the use of natural language to describe a large variety of information in an easy-to-understand and compact way. It aims at fully supporting all the information needs.

AUTHOR VISUALIZATIONS

VIS Author Profiles focuses on individual authors and aim to summarize their publication activity and co-author collaboration networks. From the social network analysis perspective, this can be considered as an ego-centric perspective and existing research in this regard is relevant that we briefly describe in the following.

Some ego-centric network visualizations have been studied for understanding the temporal evolution of co-author relationships. One such approach encodes years as rings around a researcher profile that are subdivided by color-coded co-authors.⁶⁷ Another technique uses links in a node-link diagram to encode temporal distributions of the joint publications.¹³⁹ While these two approaches focus on the aggregated data, other existing works also extend such representations to include temporal information. For instance, Shi et al.¹⁵⁴ extend the nodes in an ego-centric visualization to a timeline and connect co-authors to points in time when a jointly authored publication appeared. Similarly, *EgoNetCloud*¹⁰⁵ uses a single timeline to which the co-authors are, then, arranged in independent and individual node-link components along the timeline of the profile author. Another very similar system, *egoSlider*¹⁸⁵, also shows author-specific timelines, where clusters of co-authors vertically interact at points in time when the joint research happened. Going beyond static graph visualizations, *MENA*⁵⁹ represents the ego-centric co-author network as a dynamic graph using small multiples. Fung et al.⁴⁶ draw inspiration from a botanical tree visualization and suggest summarizing collaborations in each branch of a time span with co-authors encoded as leaves.

Besides ego-centric author visualizations, several other visualization approaches include other relevant information in addition to co-author graphs. The collaboration relationships among a group of researchers are often demonstrated in the form of a co-author graph.^{5,61,89} These systems often integrate additional information, for instance on research topics, in the coloring of the nodes⁸⁹, or as a special topic-collaboration nodes⁵. *CiteWiz*⁴¹ combines co-author networks with keyword co-occurrence networks and citation impact visualizations. Similarly, *PivotSlice*¹⁹³ uses faceted exploration to investigate an author's publication records based on keywords, citation, and publication venue information. Another very similar system, *PivotPaths*³⁸, links author nodes, paper nodes, and keyword nodes to support interactive exploration. *SurVis*¹⁵ associates word clouds of keywords and authors in a faceted browsing approach for effectively managing and searching literature collection; though *SurVis* is a publication-centric system rather than an author-centric one. For learning about further approaches on the visualization of scientific bibliographic datasets, interested readers can refer

to the survey of Federico and others.⁴²

VIS Author Profiles includes a visualization related to the aforementioned ego-centric visualizations to show the collaboration network of an author. However, it is only a small part of the interactive document—VAP also integrates other related versatile information as textual explanations to the profile description, for instance, covering research topics and a summary of general author information. In general, this is the most distinguishing factor, and we are not aware of any approach that augments such visualizations with generated textual descriptions.

4.2.3 Generation Pipeline

The first step in our generation pipeline is the pre-processing of *DBLP* publication records. We integrate this data with the *Visualization Publication Data Collection* and enrich it with keywords as discussed in Section 4.2.1. All pre-processing is done in Java. The front end is written in HTML and JavaScript. For producing the visualizations, we use Scalable Vector Graphics (SVG) and D₃. The text generation templates are implemented as part of the front-end code in JavaScript.

To build the templates, we followed an informal iterative approach. In every iteration, we drafted a text fragment based on an author’s publication record. We implemented a base version of it as a template and then kept on refining and fine-tuning the template by testing over many random authors belonging to various personas and special cases. We continued the iterations until the text covered all the information needs discussed in Section 4.2.1. With this approach, we received quick results and continuously tested the generated text. Step by step, we also integrated interactions and visualizations in a similar fashion.

Directed acyclic decision graphs (Figure 4.3 gives an example) generate text from the parameterized templates. An author’s profile consists of three fixed paragraphs (one for each group of information needs) and we define a decision graph per paragraph. The sequence of text fragments (usually, a text fragment represents a sentence or phrase) within a paragraph is fixed. In the decision graph, *start* and *stop* vertices mark the beginning and end of the text generation process, *text* vertices (rectangular nodes) add a new text fragment to the paragraph when traversed, and finally *decision* vertices (rounded rectangular nodes) determine the path based on conditional statements. The path is deterministic, and any traversal from *start* to *stop* vertex results in a paragraph. Hence, the text fragments need to be designed to form well-formed sentences regarding all possible paths.

Our approach is flexible and produces grammatically correct sentences if all conditions are carefully checked. In the templates, we take into account the already generated text and connect it to previous sentences with appropriate conjunctions. The use of numerals (e.g., one, two) in place of numbers (if a paragraph only contains small numbers less than 10) and rounding down larger numbers to the nearest fifties make the text more natural to read. We use adjectives to characterize the objects we are describing (e.g., *long-lasting collaboration*) and consider different tenses for a correct referral to time spans.

4.2.4 The System

VIS Author Profiles (VAP) is a Web-based visual analytics tool that generates profiles for authors of the visualization community describing their publication record. It is designed to fulfill the information needs discussed in Section 4.2.1 (**IN 1** – General Information, **IN 2** – Research Topics, and **IN 3** – Collaboration Network).

Figure 4.1 shows the user interface of VAP. It uses a two-panel layout. The central panel displays the textual description of an author profile and is divided into three paragraphs describing (i) general information, (ii) research topics, and (iii) the co-author network. The *co-author publication timeline* visualization at the bottom of this panel provides insights into the joint work by presenting all co-authors and their yearly publications. The right sidebar is reserved for displaying details on demand, such as additional explanations and publication records. The *ego publication timeline* at the top right provides temporal distribution of individual, joint, and topic-filtered publication records with respect to the selected profile author. Enlarged versions of the word-sized visualizations are also displayed in this bar chart. The text produced in boldface characters is interactive and allows for exploring the underlying publication records by presenting them in the sidebar. Author names in the publication list and anywhere on the page are links to their profiles. Please note that the authors, not available in the *Visualization Publication Data Collection*, are not explorable through our tool and are marked with an asterisk (*). An info icon ⓘ in the text indicates that users can explore additional information by clicking and loading this information in the sidebar.

To provide a quick overview of an author’s experience, VAP uses digital badges and displays them next to the author’s name in the header as shown in Figure 4.1. It is a concept applied in computer games and is often used for *gamification* (i.e., to make a non-game interface or tasks more enjoyable or increase the motivation of users through integrating game aspects). These badges are indicators of accomplishments and skills. VAP awards gold, silver, and bronze badges based on various levels of experience in terms of the number of published papers, length of active publication time, and supervision of other researchers.



For instance, the golden sup-supervisor badge (third from left) indicates that the supervisees of the author have also started supervising (i.e., is following an academic career) and the silver article badge (fourth from left) highlights an accomplishment of publishing thirty or more research papers. Badges are intended to roughly match the author personas (cf. Section 4.2.1): *students* can realistically earn bronze badges, but as soon as authors have a first silver badge, they can be considered *researchers*, and *senior researchers* for a first golden badge. Only *occasional contributors* are harder to link to the badges, as they might be active for a long time but do not publish many papers.

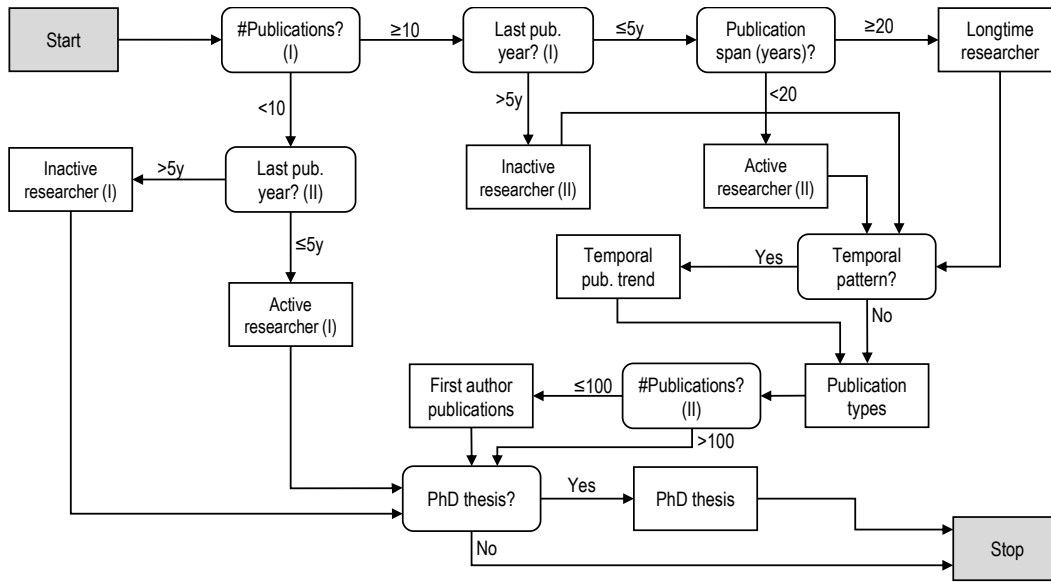


Figure 4.3: Decision graph explaining the flow of text generation for the first paragraph of a profile. Rectangular nodes represent text vertices, whereas nodes with rounded corners are decision vertices. The traversal of any path from start to end node produces a meaningful paragraph.

GENERAL INFORMATION

The first paragraph of the main text gives an overview of the publication statistics of the author and aims at fulfilling the general information needs (IN 1). Figure 4.4 shows the text for author *Benjamin Bach* as an example of a *researcher*. It is generated by traversing the decision graph shown in Figure 4.3 according to the following path:

Start → #Publications? (I) → Last pub. year? (I) → Publication span (years)? → Active researcher (II) → Temporal pattern? → Temporal pub. trend → Publication types → #Publications? (II) → First author publications → PhD thesis? → PhD thesis → Stop.

In general, the first sentence starts by reporting the total number of publications (IN 1.1) and highlights the current status as active if the last publication appeared no earlier than five years ago (nodes *Active researcher (I)* and *(II)* in Figure 4.3) or span of research if the researcher is not active anymore (nodes *Inactive researcher (I)* and *(II)*). Active longtime contribution (≥ 20 years), like for author *Ben Schneiderman* (cf. Figure 4.1), is also indicated (IN 1.2; node *Longtime researcher*). The adjacent sparkline shows the temporal distribution of publications in more detail (IN 1.2 – publication span).

We manually analyzed the publication behavior of a set of authors across all personas to extract the most commonly occurring temporal patterns, which were then implemented and detected automati-





Benjamin Bach is an active researcher since 2010 and has published **25 research papers** , where most contributions appeared since 2015 (14 publications). The publications include **12 journal articles**  and **13 proceedings papers** . Out of the total 25 publications, the author published 14 articles as first author . The author received a PhD degree from University of Paris-Sud, Orsay, France with the dissertation published in 2014 and titled “Connections, changes, and cubes : unfolding dynamic networks for visual exploration”.

Figure 4.4: Excerpt (general information) from Benjamin Bach's profile.

cally. For each detected pattern (**IN 1.2** – temporal publication distribution), we add a clause (node *Temporal pub. trend*). As two of the most frequent and notable patterns, we currently highlight if authors published more than half the papers in any third of their publication history and exceptional peak years greater than twice the second maximum value in the time series of yearly publications. Author *Bach* has a clearly growing publication rate, and hence most publications appeared in the last third.

Next, we discern the publications as journal articles and proceedings papers (node *Publication types*) with respective sparklines attached (**IN 1.1** – publication types).

Considering the importance of publications as the first author for early-career researchers, the third sentence states the number and temporal distribution of such publications as text and in a sparkline respectively (**IN 1.3** – first author publications; node *First author publications*). For senior researchers, this number is not as important anymore and hence skipped (cf. Figure 4.1).

Finally, information about the author’s PhD (dissertation title, institution, and year of publication) is described in the last sentence of this paragraph (**IN 1.3** – PhD thesis; node *PhD thesis*)—unfortunately, this data is only available for a fraction of the authors in DBLP. Profile authors with one or only a few publications have a largely reduced version of this paragraph in their profile.

RESEARCH TOPICS

The following second paragraph aims at satisfying the information needs corresponding to the research communities and evolution of topics (**IN 2**). We analyze the publications enriched with venue-specific keywords and author-specified keywords as described in Section 4.2.1. Figure 4.5 shows an excerpt of the profile of *senior researcher Daniel Weiskopf*. Since we restrict ourselves to the *visualization* community, this paragraph starts with the description of authors’ status within the visualization community (**IN 2.1** – research community). We discern between core member, member, and contributor depending on the number of research papers that are classified under the *visualization* keyword. We describe the author as *active* if the most recent publication appeared within the last two years. Next, we discuss relevant subfields for the visualization community: *information visualization*, *scientific visualization*, and *visual analytics*, which appear in the order of frequency in that they

Weiskopf is a current core member of the **visualization** community. Weiskopf has contributed to all **information visualization**, **visual analytics**, and **scientific visualization**. Focus areas of the author are **volume visualization**, **graph visualization**, **flow visualization**, **multimedia visualization**, and the evaluation of visualization. Other current research areas of the author are **computer graphics**, **software engineering**, and **bioinformatics**. Researchers with similar areas of expertise[Ⓞ] are **Quang Vinh Nguyen**, **Jarke J. van Wijk**, **Jonathan C. Roberts**, **Heidrun Schumann**, and **Mikael Jern**.

Figure 4.5: Excerpt (research topics) from Daniel Weiskopf's profile.

are assigned to the author's publications (■ IN 2.2 – subfields). The focus areas within visualization research are reported in the following sentence (■ IN 2.2 – focus areas).




Then, other research areas are listed the author has contributed to, again in the order of keyword frequency (■ IN 2.1 – connection to other communities). We discern current and past research topics (since the authors of the examples in Figure 4.1 and 4.5 are still actively contributing to all listed communities, this part is skipped here). In both cases, we limit the list to three to six keywords according to Equation 4.1. The communities are highlighted in blue, subfields in light blue, and research topics in light gray. Those keywords marked in bold font are linked with publications and are listed in the sidebar on click. Their temporal evolution is shown as an overlay on the *ego publication timeline* (■ IN 2.3 – evolution of topics). A comparison of the evolution of research topics with the collaboration timeline provides further insights into the impacts of collaborators on the research interests of the profile author.

For the identification of authors having similar research interests (■ IN 2.4), we compare the research interests of the profile author with all other authors within the visualization community. We compute their cosine similarity based on keyword frequency vectors. Since keyword coverage is not optimal in our data, we restrict the similarity computation to authors with at least 30 publications to avoid marking less similar authors as similar. Since frequent collaborators are likely to share most research interests, we also exclude co-authors that share more than two publications with the profile author from the comparison. We use again Equation 4.1 to cut short the list to the three to five most similar authors in the last sentence of this paragraph. A longer list of similar authors is available on demand (cf. Figure 4.1, bottom right).

CO-AUTHOR NETWORK

To analyze the collaboration network formed within the co-authors (■ IN 3), we explore pairwise collaborations as well as groups of researchers working together. Figure 4.6 shows the textual description of the co-author network of *researcher Katherine E. Isaacs* along with the *co-author publication timeline*.

This paragraph first describes the pairwise collaborations among the most frequent co-authors

Isaacs’s most frequent co-author^① is **Abhinav Bhatele** . It is ongoing collaboration since 2011 with **12 publications**. **Todd Gamblin**  is the second most frequent co-author, an ongoing collaboration with **11 publications** since 2011. Together with **Peer-Timo Bremer** , the author published **11 research papers** since 2011. Regarding collaboration subgroups^①, Isaacs has worked with **Martin Schulz**, **Peer-Timo Bremer**, **Abhinav Bhatele**, and **Todd Gamblin** on high performance computing, parallel computing, and information visualization and produced **9 research papers**. Another notable group is with **Peer-Timo Bremer** and **Bernd Hamann** resulting in **8 publications**.

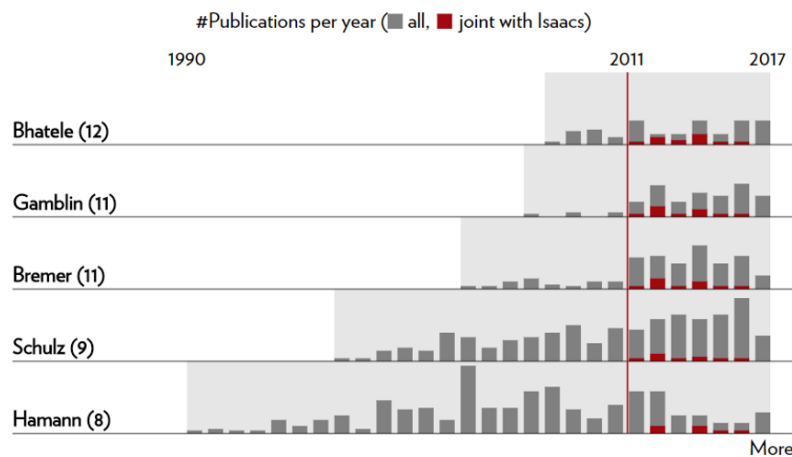


Figure 4.6: Excerpt (co-author network) from Katherine E. Isaacs's profile including the *co-author publication timeline*. Gray bars indicate the number of publications per year of the co-author, while red bars show joint publications with Isaacs.

(**IN 3.1** – main collaborators). The frequent three to eight co-authors are selected using Equation 4.1 from the list of all co-authors based on the number of joint publications. The most frequent collaborations are described along with their current status (e.g., still ongoing, ended). The use of adjectives such as *prolonged* and *long-lasting* highlights whether the span of collaboration is more than 15 years. If the collaboration has ended, we mark it with the total span of years along with the number of joint publications. Adjacent sparklines show the temporal distribution of joint work. Users can load the comparative view of joint publications as a proportion of the profile author’s overall publications in the *ego publication timeline* as shown in Figure 4.1. Further exploration of the joint publications is possible by interacting with this chart—the publications of a selected year appear in the right sidebar, sorted in alphabetical order of their titles.

In the scientific community, two roles of an author are of significant importance; i.e., supervisor and supervisee (**IN 3.2** and **IN 3.3**). To identify these roles, we use a heuristic method with two assumptions: (i) the name of the supervisor appears in the last positions in an author sequence of a

publication and (ii) the name of supervisee appears at the first position. All co-authors of the profile author who started publishing at least five years prior to the profile author are added to a list of potential supervisors. Then, the potential supervisors who appear more frequently before the profile author in their joint publications than after are discarded. Finally, a potential supervisor is identified as a supervisor if, in fifty percent of their joint publications, the potential supervisor appears at the last position. Analogously, supervisees are computed. We highlight the author roles while describing their collaborations with adjacent sparklines as introduced above. For *senior researchers* like author *Ben Shneiderman* (cf. Figure 4.1), the list of supervisees is quite long. We, therefore, list more supervisees in the following sentence after cutting down the list to a maximum of five further supervisees by applying Equation 4.1 and using their number of joint publications with the profile author as a rating. Sparklines again provide details on the temporal distribution of joint work. To find if the supervisees of the profile author are already assuming the role of supervisors (■ IN 3.3), we look for the supervisees of the supervisees and describe this in the fifth sentence. Since our approach could not detect a clear supervisor or supervisees for *Katherine E. Isaacs*, her profile text does not comment on this (cf. Figure 4.6).

For identifying meaningful subgroups among the co-authors who have frequently worked together with the profile author on the same publications (■ IN 3.4), VAP employs *formal concept analysis* (FCA).^{182,47} FCA provides paired sets of co-authors and joint publications (*formal concepts*) that are maximal both regarding co-authors and publications; that means, there is no other co-author that has also contributed to all identified publications and there is no other publication that has been co-authored by all identified co-authors. To focus on higher-order groups of co-authors and substantial collaboration, we discard formal concepts with less than two co-authors ($n < 2$) and less than three publications ($m < 3$). For all other formal concepts, we compute a score $\sqrt{n} \cdot m$ that rates both larger groups and groups with more publications higher. We apply a mitigating transform on the number of co-authors \sqrt{n} because otherwise large groups of co-authors with only a few publications might dominate the scores. This ranking results in an ordered list of groups of co-authors. However, groups in the list might significantly overlap with respect to co-authors, for instance, one being a subgroup of the other. Since we want to avoid listing similar groups, we sweep through the list and discard all groups that share a high similarity with any of the previously listed groups (Jaccard coefficient of $\geq 1/3$ comparing the two sets of co-authors). Among the list of groups sorted based on score, we select the top two groups according to Equation 4.1 for presenting in the profile text along with the number of joint publications and research topics. A longer list of collaboration groups and group publications is available on demand and is presented in the sidebar.

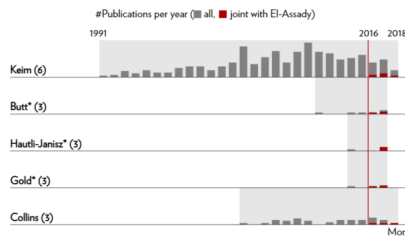
Whereas only the most frequent collaborations are described in the text, the diagram showing the *co-author publication timeline* below the text (cf. Figure 4.6) allows for the exploration of all collaborations in detail. It displays distributions of the co-author's publications per year (gray) as well as those joint with the profile author (red) on a timeline. The timeline begins with the starting year of the most senior author in the visible co-authors, with a vertical red line pointing at the starting year of the profile author. The active spans of co-authors are marked with a light gray background. The *More* button allows for expanding the list. The number in parentheses after the co-author's name shows the number of joint publications. The gray and red bars are selectable, and the respective publications are

Mennatallah El-Assady

Mennatallah El-Assady has published seven research papers since 2016, including five journal articles and two proceedings papers.

El-Assady made contributions to the **visualization** research community.

El-Assady worked with **Daniel A. Keim**, Miriam Butt*, Annette Hautli-Janisz*, Valentin Gold*, and Christopher Collins.



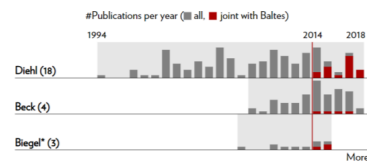
Sebastian Baltes

Indices 10+ Active 5+

Sebastian Baltes has published 22 research papers since 2014, including 10 journal articles and 12 proceedings papers. Out of the total 22 publications, the author published 18 articles as first author.

Baltes is a current contributor of the **visualization** research community. The author is also contributing to the area of **software engineering**.

Baltes's most frequent co-author and supervisor is **Stephan Diehl**. It is ongoing collaboration since 2014 with 18 publications. **Fabian Beck** is the second most frequent co-author, an ongoing collaboration with 4 publications since 2014. Together with Benjamin Biegel*, the author published 3 research papers since 2014.



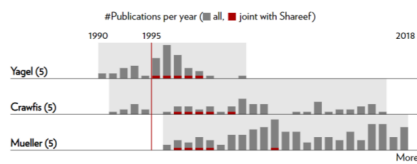
Naeem Shareef

Indices 10+ Active 20+

Naeem Shareef is a longtime research contributor since 1995 and has published 11 research papers, where most contributions appeared until 2003 (7 publications). The publications include 3 journal articles and 8 proceedings papers. Out of the total 11 publications, the author published 4 articles as first author.

Shareef is a current contributor of the **visualization** research community. The author is also contributing to the area of **computer graphics**.

Shareef's most frequent co-author and past supervisor is **Roni Yagel**. This collaboration produced 5 publications between 1995 and 1999. **Roger Crawfis** is the second most frequent co-author, a collaboration that produced 5 publications between 1997 and 2002. Together with **Klaus Mueller**, the author published 5 research papers between 1997 and 2006. Regarding collaboration subgroups, Shareef has worked with **Klaus Mueller** and **Roger Crawfis** produced 4 research papers.



Caroline Ziemkiewicz

Indices 10+ Active 10+

Caroline Ziemkiewicz is an active researcher since 2006 and has published 28 research papers, including 16 journal articles and 12 proceedings papers. Out of the total 28 publications, the author published 10 articles as first author.

Ziemkiewicz was a member of the **visualization** community. Ziemkiewicz's expertise mainly covers the subfields of **visual analytics** and **information visualization**. A focus area of the author was the evaluation of visualization. The author also worked in the fields of **human-computer interaction** and **geoscience**.

Ziemkiewicz's most frequent co-author and past supervisor is **Remco Chang**. It is ongoing collaboration since 2006 with 16 publications. **William Ribarsky** is the second most frequent co-author, a collaboration that produced 11 publications between 2006 and 2013. Together with **Robert Kosara**, the author published 8 research papers between 2007 and 2010. Regarding collaboration subgroups, Ziemkiewicz has worked with **William Ribarsky** and **Remco Chang** on visual analytics and information visualization and produced 11 research papers. Another notable group is with **Remco Chang**, **R. Jordan Crouser**, and **Alvita Ottley** resulting in 4 publications.

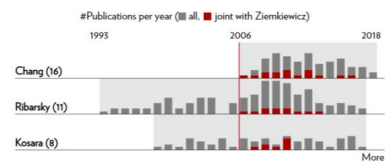


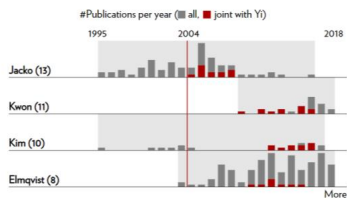
Figure 4.7: Author profiles of several researchers falling into different author personas. For each persona, VIS Author Profiles has successfully generated meaningful profiles. (These example profiles span across two figures; please see the next figure for two more author profiles).

Ji Soo Yi   

Ji Soo Yi is an active researcher since 2004 and has published **41 research papers**, including **26 journal articles** and **15 proceedings papers**. Out of the total 41 publications, the author published 8 articles as first author. The author received a PhD degree from Georgia Institute of Technology, Atlanta, GA, USA with the dissertation published in 2008 and titled "Visualized decision making: development and application of information visualization techniques to improve decision quality of nursing home choice".

Yi was a member of the **visualization** community. Yi's expertise mainly covers the subfields of **information visualization** and **visual analytics**. Focus areas of the author were graph visualization, medical visualization, and the evaluation of visualization. The author also worked in the fields of **human-computer interaction** and **bioinformatics**. Researchers with similar areas of expertise are **Jeffrey Heer**, **Catherine Plaisant**, **Bongshin Lee**, **Stuart K. Card**, and **George G. Robertson**.

Yi's most frequent co-author and past supervisor is **Julie A. Jacko**. This collaboration produced **13 publications** between 2004 and 2008. **Bum Chul Kwon** is the second most frequent co-author, an ongoing collaboration with **11 publications** since 2009 and Yi was acting as a supervisor. Together with **Sung-Hee Kim**, the author published **10 research papers** since 2012. Regarding collaboration subgroups, Yi has worked with Young Sang Choi* and **Julie A. Jacko** on human-computer interaction and produced **5 research papers**. Another considerably large group is with Leon Barnard*, François Sainfort*, Thitima Kongnakorn*, **Julie A. Jacko**, Paula J. Edwards*, and Kevin P. Moloney* resulting in **3 publications**.



Daniel A. Keim   

Daniel A. Keim is an active and longtime research contributor with more than **300 publications** since 1991. Keim's published work includes **130 journal articles** and **210 proceedings papers**. The author received a PhD degree from Ludwig Maximilian University of Munich, Germany with the dissertation published in 1995 and titled "Visual support for query specification and data mining".

Keim is a current core member of the **visualization** community. Keim has contributed to all **visual analytics**, **information visualization**, and **scientific visualization**. Other current research areas of the author are **data management** and **machine learning**. Researchers with similar areas of expertise are **Mark A. Whiting**, **Tatiana von Landesberger**, **Ross Maciejewski**, **Wolfgang Aigner**, and **Giuseppe Santucci**.

Keim's most frequent co-author and past supervisee is **Tobias Schreck**. It is ongoing collaboration since 2004 with **53 publications**. **Ming C. Hao** is the second most frequent co-author, a collaboration that produced **36 publications** between 2001 and 2014. Together with **Umeshwar Dayal**, the author published **36 research papers** between 2001 and 2014. In addition to **Schreck**, **Mansmann**, and **Schneidewind**, further supervisees of Keim with considerable amount of publications are **Daniela Oelke**, **Christian Rohrdantz**, **Haldór Janetzko**, **Sebastian Mittelstädt**, and **Peter Bak**. Keim's supervisee **Tobias Schreck** is already supervising other researchers. Regarding collaboration subgroups, Keim has worked with **Ming C. Hao** and **Umeshwar Dayal** on visual analytics and information visualization and produced **36 research papers**. Another notable group is with **Christian Panse** and **Mike Sips** resulting in **13 publications**.

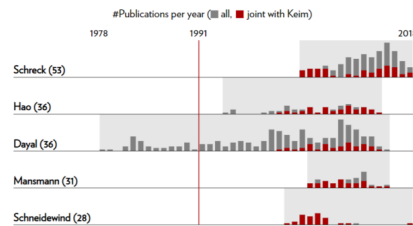


Figure 4.8: (Continuation of Figure 4.7) Two author profiles for the *senior researcher* persona.

displayed in the sidebar. This visualization provides an easy way to explore temporal variations in the collaborations, possible shifts of collaborators from one co-author to another, and the span of joint work.

4.2.5 Author-Profile Gallery for Different Personas

To demonstrate the flexibility of the generation approach, Figure 4.7 shows six author profiles of researchers with varying levels of experience and expertise. VIS Author Profiles does not automatically determine a researcher's persona. Instead, researchers are indirectly associated with personas based on their experience, such as the number of publications, active years in research, and whether they have supervised other young researchers.*

El-Assady is probably a *student* working toward her PhD in the visualization community. She has not earned any badges yet, and there is not enough data to comment on her research topics and collab-

*It is important to note that publication records in VAP are included only until August 2018. Hence, the author profiles might already be outdated. Since an author's profile text is automatically generated, there might be some errors.

oration groups. Yet, her co-author publication timeline reveals other researchers she is working with, all seniors to her. Shareef may relate to the persona *occasional contributor* as he is active in research for a very long time (20+ years) but have only a few (11) publications in the visualization community. He has also collaborated in different time spans with different researchers (e.g., with Yagel during 1995–1999, with Mueller between 1997–2006), that might be another indication of him being an *occasional contributor*. Baltes and Ziemkiewicz are early career *researchers* who already earned bronze or silver badges. They may not yet be supervising other researchers but are contributing actively to the visualization community. Since the data is not enough in both of these cases, VAP has not listed the similar authors to Baltes and Ziemkiewicz. Yi and Keim are *senior researchers* as evident from the silver and gold badges on their profiles. They have many publications and as a result, VAP was able to identify more insights about them: For instance, see the second paragraph of their profiles to learn about the evolution of research topics and similar researchers, and look at the third paragraph to know more about their collaboration groups. While Yi has supervised other researchers (supervisor badge), Keim has some supervisees who are already supervising other researchers (sup-supervisor badge).

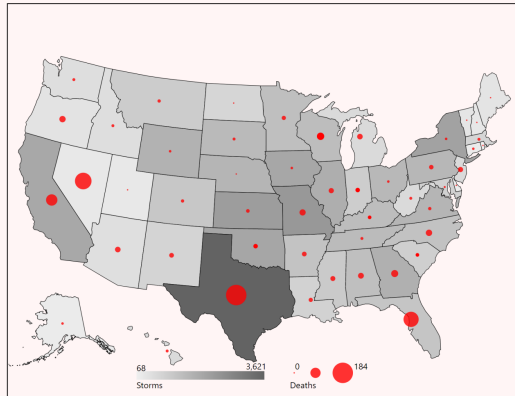
Through these six author profiles, it becomes clear that VAP can produce meaningful interactive author profiles for researchers having any level of experience and expertise. Moreover, mirroring the two initial motivating scenarios (cf. Section 4.2.1), the corresponding research paper⁹⁵ also demonstrates the usability and application of the approach in a realistic setting; interested readers can consult the last section of the paper.

4.3 Interactive Map Reports: Bivariate Geographical Data

The interplay of two variables reveals how one potentially influences the other. In a geographic context, this influence may depend on the geography of the region. For instance, storms might cause more fatalities in densely populated areas. Thematic maps are mostly used to visualize variations in the values of a variable across geographical space. The variable is mostly encoded as color²³, size, and shape of the geographical regions (cartograms), or by overlaying specific symbols on top of the map.⁴⁵ To reveal their relationship, however, both variables need to be visualized simultaneously.

Prior research has generalized the concept of map visualization to bivariate^{66,22} and multivariate geographic data.⁸⁴ The work of Elmer⁴⁰ presents a taxonomy of bivariate map visualizations. This taxonomy is based on various combinations of visual characteristics and is adapted from the work of Nelson¹²⁷ and MacEachren.¹⁰⁸ According to Elmer⁴⁰, although there are more than eleven different kinds of bivariate maps—identified from six cartography books—, only two have been generally employed in the existing literature. These two types include bivariate choropleth maps and choropleths with overlaid graduated symbols (similar to the ones shown in Figure 4.9). The bivariate map visualizations, by construction, are visually more complex and harder to comprehend in comparison to their univariate counterparts. Especially inexperienced users might face problems in effectively interpreting the bivariate visualization. And even experienced users might find it hard to detect spatial patterns and spot outliers. Hence, there is a need to make bivariate geo-statistical visualizations more self-explaining and guide users through the data analysis.

Fatalities caused by storms, USA, 2017



The map shows the number of deaths, encoded as the area of the dots (●), and storms ■ across the different states of the United States of America during 2017.

The average number of deaths per state was 14.9 ●, and it varies from no instances, for example, in Delaware ◊ to 184 ● in Texas. Other states showing high number of deaths are Nevada (121 ●), Florida (101 ●), and California (56 ●) ◊. The states Missouri, Nevada, Texas, and Wisconsin are different from their respective neighboring states as they suffered comparatively a lot more casualties ◊. Southern states experienced higher number of deaths and storms compared to the other states.

A higher number of deaths are associated with a higher number of storms among Southern states. In comparison to the other states, Texas experienced high number of deaths as a result of a high number of storms. Despite having a relatively small number of storms (272), Nevada shows a high number of deaths (121 ●).



Wisconsin

Wisconsin experienced an above average number of deaths (23 ●) and an above average number of storms (1346 ■) compared to the other states. It ranks 5th among all regions with respect to deaths. Compared to its neighbors (Illinois, Iowa, Michigan, and Minnesota), Wisconsin experienced substantially more deaths.



Comparison of Georgia and South Carolina

Georgia experienced a higher number of deaths (21 ●) and a higher number of storms (1679 ■) compared to South Carolina (5 ● deaths, 912 storms).

Figure 4.9: A map report describing loss of lives due to storms in the USA during 2017. The map visualization uses two different encodings to visualize a *focus* and a *context* variable. The narrative (right column) provides an overview of the data analysis. Graphics in the text help establish linking between the two representations. Users can get additional details on a selected region or on a comparison of two selected regions (dashed rectangles).

To facilitate the analysis of relationships among variables in a geographical setting, additional visualization can be linked with bivariate maps. For instance, Monmonier¹²³ presents visual statistical summaries of variables as a scatterplot matrix alongside a bivariate map. However, though the scatterplot matrix reveals relationships among statistical variables, once again, it suffers from a certain complexity that may not be suitable for inexperienced visualization users. The use of textual explanations for bivariate maps can increase the self-explainability and understandability of visually cluttered visualizations—multiple encodings may render them less effective.⁴⁴

Augmenting a bivariate map visualization with a textual explanation and interactively linking both representations can significantly improve users' abilities to understand the data. Interactive Map Reports (IMR)* is a Web-based tool that automatically generates a textual narrative alongside a map visualization to describe the analysis results for bivariate geo-statistical data. Figure 4.9 shows an example; it explains fatalities caused by storm events in the USA, 2017. The reports summarize noteworthy patterns and relationships among the variables. In addition, they provide explanations on selected regions and the ability to compare any two regions of interest. The color and shape encoding of variables

*The interactive system is available at: <https://mrshahidlatif.github.io/interactive-map-reports>.

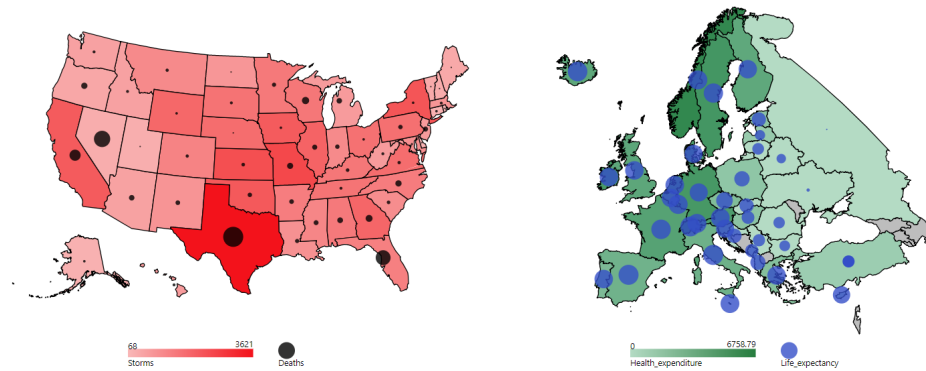


Figure 4.10: Bivariate map visualizations used in the explorative study. (Left, P1) Deaths caused by storms in various states of the USA. (Right, P2) Average life expectancy and health expenditures across Europe.

in the textual narrative help establish quick linking of respective regions across both representations.

4.3.1 Scope, Data, and Content Determination

The generation process begins with the selection and prioritization of the content that is to be presented. In this regard, an explorative study conducted with two visualization experts provides an overview—the results align with the findings of empirical studies (cf. Section 3.7.2)—and serves as the basis of *Interactive Map Reports*. Before delving into the details of the explorative study, we first introduce the scope and dataset.

SCOPE AND DATA

IMR focuses on the analysis of bivariate geo-statistical data—measurements of two numerical variables for a geographic region. Particularly relevant are those scenarios where one variable potentially influences the other; for example, life expectancy may depend on the amount of health expenditure. Similarly, the intensity and number of storms can influence the number of lives lost. In the following, we refer to the variable that potentially depends on the other as the **focus** variable and the other as the **context** variable. If causality can be assumed (e.g., because it is obvious or there exists a reasonable explanation for it), it points from *context* to the *focus* variable. In our analysis, we use three levels of geography, namely regions (e.g., USA), groups of subregions (e.g., group of states), and subregions (e.g., individual states). We use the storm–death dataset as our running example.

EXPLORATIVE STUDY

To get an overview of what aspects to consider while describing bivariate geo-statistical data, we conducted an explorative study with two participants (P1, P2). Both participants were PhD students working in the field of visualization but were not involved in this project. They were presented with

an interactive version of a bivariate map visualization as shown in Figure 4.10. In these visualizations, the *focus* variable was encoded in the radius of filled circles placed on top of the choropleth map showing the *context* variable. The participants were asked to summarize the visualization (Task I), describe one particular subregion (Task II), and provide a comparative view of two given subregions (Task III). They had the possibility to write as much text as they wanted and there was no time limit. Below are the findings of the explorative study organized per task:

Task I – Summary: Both participants began with the description of the subregions having minimum and maximum values of the *focus* variable, followed by the explanation of outlying regions. P1 included information on a possible correlation between the variables. P2 described the spatial trend of the *context* variable. Finally, both participants noticed and described abrupt changes in values between neighboring subregions.

Task II – Region-specific description: P1 described the values of both variables for the given subregion, followed by naming the other regions that show similar behavior, whereas P2 provided a comparison with the mean values. Further, P2 highlighted a specific subregion that has higher values of the *context* variable compared to its direct neighbors.

Task III – Comparison: Both participants compared the values of each variable for two regions and described them in a single sentence. P1 included more details about one subregion, as one of the subregions presented to P1 for comparison was an outlier.

The results show that the prominent aspects are the reporting of outliers (univariate and bivariate), comparisons of regions with their neighbors, variations of variable values across space, and subregions that show similar behavior. In addition, a correlation and variations of values across different sub-regional levels (e.g., parts of Europe) might reveal interesting patterns and are worth reporting. For instance, in Figure 4.10 (left), the correlation is much stronger in the Southern states ($\rho = 0.753$) compared to the overall correlation for the whole country ($\rho = 0.400$).

4.3.2 Analyzing Bivariate Geographic Data

Next, we discuss statistical approaches to automatically identify the content that will be a part of the narrative. In contrast to basic information such as statistical ranges, correlation, and extrema; the detection of univariate outliers, bivariate outliers, and regional differences requires sophisticated data analysis approaches.

UNIVARIATE OUTLIERS

The importance of extreme values (minimum and maximum) in a dataset varies depending on the distribution of variables. A Tukey's boxplot¹⁷⁴ uses measures namely, the first quartile ($Q1$), median ($Q2$), third quartile ($Q3$), and interquartile range ($IQR = Q3 - Q1$) to describe a univariate distribution. Based on Tukey's boxplot, Hoaglin and others⁶³ categorize the observations smaller than

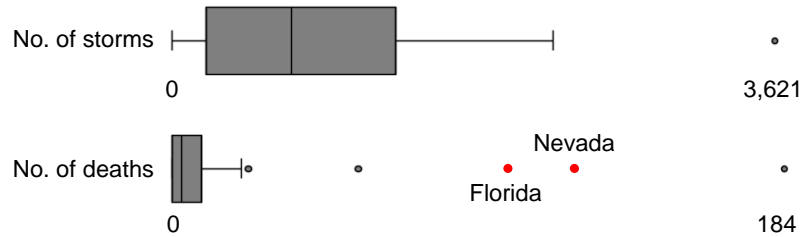


Figure 4.11: Box plots showing the distribution of *deaths* caused by *storms* in the USA during 2017. The dataset contains univariate outliers in both variables.

$Q1 - 1.5 \cdot IQR$ or larger than $Q3 + 1.5 \cdot IQR$ as the potential candidates for outliers. Although somewhat arbitrary, this threshold for detecting outliers works well based on their experience with many datasets.

We analyze each variable individually and identify the univariate outliers, i.e., the points lying outside the range: $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$. Figure 4.11 shows the distribution and outliers corresponding to each of the two variables in our exemplary dataset.

BIVARIATE OUTLIERS

We are also interested in subregions that demonstrate different behavior compared to the rest of the subregions based on the values of both variables. Such a bivariate outlier may not necessarily be an outlier in both univariate variables. For instance, although the states of Nevada and Florida in Figure 4.11 (marked with red dots) are not outliers in variable *storms*, they are bivariate outliers as shown by the bagplot (Figure 4.12). A bagplot¹⁴⁴ is a bivariate generalization of a boxplot and visualizes the distribution, spread, and outliers jointly for both variables. Three main components of a bagplot are: the *bag* containing 50% of the observations, the *fence* usually obtained by inflating the bag by a factor of 3 separating inliers from outliers, and the *loop*, that is the convex hull of the points lying between the bag and the fence.

The detection of bivariate outliers depends on the shape or distribution of the data, which is often characterized by a covariance matrix. To identify outliers, we use a well-known distance measure—the Mahalanobis distance—which takes into account the covariance matrix and is defined as the distance between an observation and a multivariate (here, bivariate) distribution. Mathematically, this distance is specified as:

$$d = \sqrt{(x - \mu)^T S^{-1} (x - \mu)} \tag{4.2}$$

where $\mathbf{x} = (x_1, x_2)$ is the vector of variables, $\boldsymbol{\mu} = (\mu_1, \mu_2)$ is the vector of means, and S is a two-dimensional symmetric covariance matrix. The resulting value d represents the Mahalanobis distance of the point x from the mean $\boldsymbol{\mu}$ of the distribution. For a constant value of d , Equation 4.2 defines a

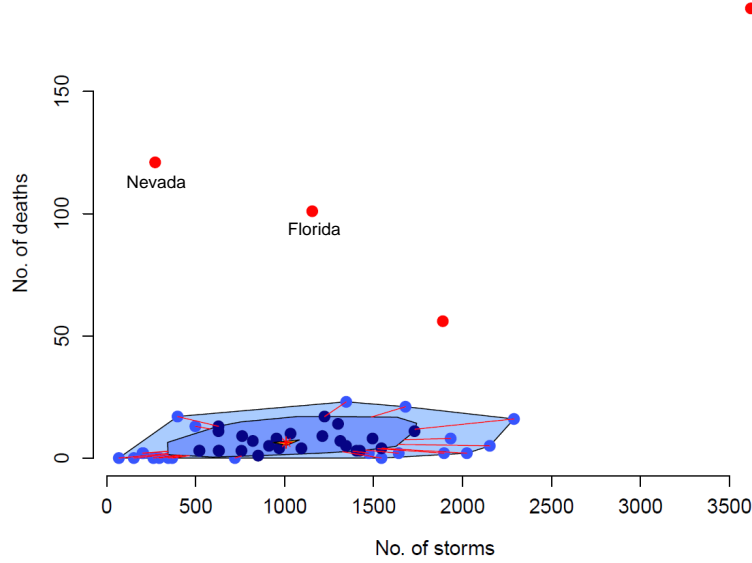


Figure 4.12: Bagplot of *deaths* caused by *storms* in the USA during 2017. The *bag* (blue) contains almost 50% of the data points, and the *loop* (light blue) includes points outside the bag but inside the fence. Bivariate outliers are marked as red dots.

two-dimensional ellipsoid centered at μ . The probability of ellipsoid follows a χ^2 distribution with p degrees of freedom.⁵⁷ Therefore, the ellipsoid satisfying

$$(x - \mu)^T S^{-1} (x - \mu) \leq \chi_p^2(\alpha) \quad (4.3)$$

has a probability of $1 - \alpha$. Hence, for $p = 2$ (bivariate case) and $\alpha = 0.5 \Rightarrow \chi^2 = 5.99$. Equation 4.3 states that any observation is considered a bivariate outlier for which the squared Mahalanobis distance is greater than 5.99.

GEOSPATIAL TRENDS

The behavior of any statistical variable can vary considerably depending on the geographical subregion. For instance, Figure 4.9 shows that the coastal states of the USA have experienced a higher number of storms, and consequently suffered more casualties. To identify this behavior, we take a regional subdivision of the overall geographic region under consideration. The United Nations geoscheme provides a classification of the countries of the world into groups. For instance, European countries are grouped into Eastern, Western, Northern, and Southern countries. Similarly, the regional classification of the USA discerns West, Midwest, Northeast, and South. Using this grouping (or other externally provided groupings), we can look for differences between these groups of subregions. In particular, we detect if there is a strong positive or negative correlation between *focus* and *context* variable in one or more of these groups. Besides the bivariate outliers, an identification of subregions that show different behavior compared to the adjacent subregions can be of interest. For instance, Fig-

ure 4.9 shows that the state of Nevada has different statistics with respect to both variables compared to its neighboring states: Arizona, California, Idaho, Oregon, and Utah. To this end, we compare the values of each variable for every subregion with its neighbors to identify the regions showing different statistics.

4.3.3 Adaptable Text Generation

Just like VIS Author Profiles, a directed acyclic decision graph guides the text generation flow and produce text from pre-written templates. Figure 4.13 shows the decision graph that is responsible for generating the main part of our map reports. To achieve flexibility in narration and to make the templates adaptable to different datasets (of the same bivariate geographic type), we leverage user-defined parameters that describe the metadata about a scenario. Through these parameters, we add semantics and domain-specific vocabulary that cannot be automatically detected from the raw data. The list of parameters along with short descriptions and possible values is shown in Table 4.2.

The parameters *Region* and *Subregion Level* define the name of the region and the name of the level of detail for regions, respectively. The parameters *Focus* and *Context Type* describe the type of both variables that can be selected from a list of predefined values. The choice of adjectives, quantifiers, and verbs depends on the variable types. For instance, for the type *casualties*, possible phrases are: “*X suffered several casualties*”, “*X reported a large number of deaths*”, or “*X lost many lives*”. Similarly, for the variable type *monetary*, a possible phrase is “*X spent a large amount on Y*” or “*X spent less on Y*”. Depending on the variable type, we pick verbs from a list of synonymous verbs to make the text more interesting and natural to read.

In addition to the quantifiers and verbs, the choice of adverbs (e.g., better, worst) depends on the context or situation under consideration. We describe three possible *situations*:

- **Positive:** Situations where higher values of the *focus* variable are desirable. For instance, higher values of average life expectancy are commonly considered to be desirable.
- **Negative:** Situations that favor lower values of the *focus* variable. For example, cities reporting less number of fatalities occurring in road accidents would be considered as better.
- **Neutral:** Situations that do not clearly favor small or large values of the *focus* variable. For instance, only depending on a country’s situation (e.g., aging society or unemployment of young people), lower or higher birth rates are desirable.

Combining the variable type with the *situation*, we can now use more expressive and specific phrases to describe the results. For *Focus Type* \leftarrow *demographic-indicator* and *Situation* \leftarrow *positive*, a possible phrase could be: “*X reports better values of life expectancy compared to Y.*” Similarly, the *Context Type* \leftarrow *incidents*, but *situation* \leftarrow *negative* could result in: “*X was the safest subregion due to the least number of accidents.*”

Another consideration is that the presence of a strong correlation may wrongly be interpreted as causal. Correlation, however, does not always imply causality and it is not possible to automatically

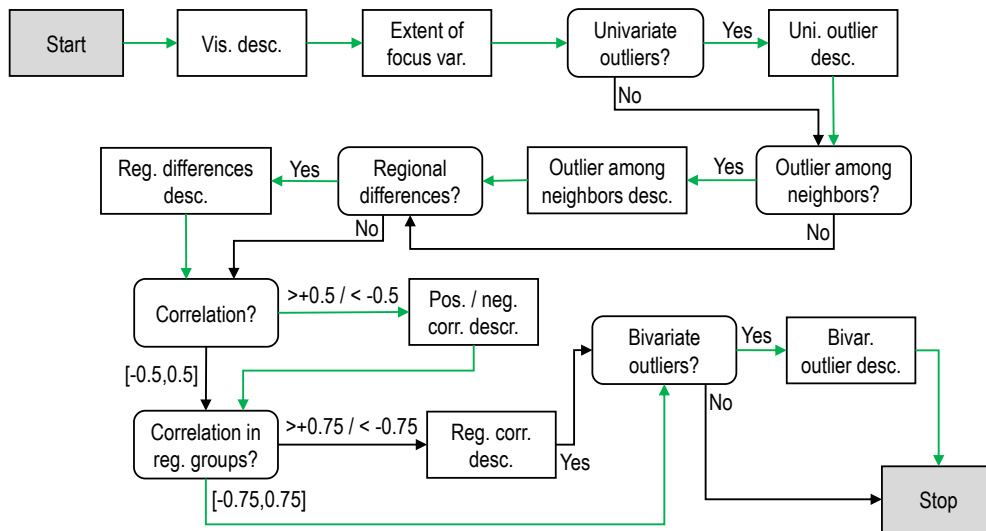


Figure 4.13: Decision graph that shows the text generation process. Round-rectangular decision nodes control the path, while rectangular text nodes add a text fragment when visited. The green path marks the narrative generation for the example in Figure 4.9.

extract causality from the numerical data. Therefore, the parameter *Causality* helps in avoiding wrong interpretations about causality.

4.3.4 The System

Interactive Map Reports is a Web-based system that generates visual analysis reports for bivariate geo-statistical data. Figure 4.9 shows the interface of our tool and the components of the generated report. A map visualization on the left visualizes two variables. The right column presents the generated narrative, consisting of an overview and additional details on the selected subregion or a comparison of any two selected subregions (shown below the map in Figure 4.9 for space efficiency). The small info icon ⓘ indicates the availability of additional explanations—for instance, a complete list of regions with their respective variable values or details on the analysis methods. The use of word-sized graphics—circles for the *focus* and color coding for the *context* variable—in text supports the quick reference and comparison of regions while reading the text. The subregion names are produced in bold-faced characters and are interactive—when clicked, the system highlights the respective subregion on the map. Besides, a tooltip presents the exact numerical values of both variables when hovering over the subregions.

Table 4.2: User-defined parameters for configuring the map reports.

Parameter	Description	Values
<i>Region</i>	Name of the region for which map is displayed	String value, e.g., <i>World, Europe, Germany</i>
<i>Subregion Level</i>	Name of the type of regions the map is subdivided in	String value, e.g., <i>countries, states, cities</i>
<i>Focus/Context Type</i>	Variable types according to predefined categories	<i>incidents, casualties, demographic-indicator, quantitative, percentage, monetary, or indicator</i>
<i>Situation</i>	Type of situation with respect to <i>focus</i> variable	<i>positive, negative, or neutral</i>
<i>Causality</i>	If causality can be assumed from <i>context</i> to <i>focus</i> variable	<i>yes or no</i>

BIVARIATE MAP VISUALIZATION

For visualizing two geo-statistical variables on a map, IMR employs a standard technique; it uses two kinds of encoding, one for each variable. The *context* variable is visualized as a choropleth map based on a single-colored linear brightness gradient. The values of the *focus* variable are encoded in the radii of filled circles and are overlaid on top of the choropleth map. These circles are positioned at the centroid of the respective subregion. We pick this visualization because it was found to perform better when comparing with other variants of bivariate map visualization.⁴⁰

The choice of color for encoding the *focus* variable depends on the specified *situation*, i.e., *positive* → green, *negative* → red, and *neutral* → orange. The choice is based on the fact that green color is generally associated with positive sentiments and safe situations, while red is considered to be a sign of warning or danger. However, the choice of orange color for *neutral* situation is somewhat arbitrary and has been chosen for better visibility as it must be overlaid on top of the gray color. For the *context* variable, we always use the same neutral gradient (light gray to dark gray) irrespective of the *situation*. as the situation depends only on the *focus* variable.

TEXTUAL SUMMARY

The first section of the generated narrative provides an overview. It is divided into three paragraphs; the structure and ordering of the paragraphs are fixed, but the sentences change considerably depending on the dataset and scenario. In Figure 4.9, the overview is generated by traversing the green path in the decision graph of Figure 4.13.

Start → *Vis. desc.* → *Extent of focus var.* → *Univariate outliers?* → *Uni. outlier desc.* → *Outlier among neighbors?* → *Outlier among neighbors desc.* → *Regional differences?* → *Reg. differences desc.* → *Correlation?* → *Pos./neg. corr. desc.* → *Correlation in reg. groups?* → *Reg. corr. desc.* → *Bivariate outliers?* → *Bivar. outlier desc.* → **Stop.**

The opening paragraph consists of a single sentence that introduces the dataset and the visual encoding with the help of in-line legends. The second paragraph summarizes the results of the univariate

analysis for the *focus* variable. It begins by stating the average values followed by its statistical range (text node *Vis. desc.*). In case of multiple subregions having the same minimum (or maximum) value, it names one subregion according to the alphabetic ordering followed by the info ⓘ icon. The complete list of these regions can be retrieved by interacting with the info icon. The next sentence lists the regions that are univariate outliers according to the *focus* variable (text node *Uni. outlier desc.*). This and all other similar lists of subregions are restricted to show only 2 to 4 subregions according to the dynamic selection method (Equation 4.1) with the possibility to view the complete list on demand.

In the second paragraph, the text node *Outlier among neighbors desc.* describes regions that exhibit substantially different values compared to their adjacent or neighboring subregions. IMR employs Tukey’s fences (Section 4.3.2) for identifying such local outliers. The method works well if the number of adjacent subregions are larger (e.g., for Missouri, Nevada, Texas, Wisconsin). However, in case of few adjacent subregions (e.g., Florida has only two adjacent states), it cannot detect meaningful outliers. For this particular case, even Dixon’s Q test³⁵—an efficient method for detecting outliers in a small number of observations—failed to detect Florida as an outlier. Since these situations are challenging to identify, IMR takes a conservative decision and exclude all subregions that have less than three neighbors from the list of potential local outliers. The last sentence of this paragraph describes regional differences in the values of both variables (text node *Reg. differences desc.*). Depending on the regional classification of the geographical region, IMR describes the subregional groups that show distinct behavior. For instance, Figure 4.9 depicts that Southern states lost more lives to storms in comparison to other states.

The last paragraph highlights the relationship between the *context* and the *focus* variable, as well as bivariate outliers. First, it reports a positive or negative correlation (text node *Pos./neg. correlation*). In case of *causality* set to *yes*, a different phrasing and vocabulary is applied to highlight causal relationship. For instance, Figure 4.9 (third paragraph) it is stated that “*Texas experienced a high number of deaths as a result of a high number of storms.*” The choice of the phrase “*as a result of*” is specific to causality. The first sentence of this paragraph is left out (e.g., in Figure 4.9) when the value of correlation is below the threshold value; Figure 4.14 shows an example of this sentence. The presence of a strong positive or negative correlation among one or more subregional groups is stated in the next sentence (text node *Reg. corr. desc.*). Then follows the description of regions that are bivariate outliers. For instance, Texas and Nevada in Figure 4.9; Texas has maximum values for both variables, whereas Nevada suffered a very high number of casualties in a relatively small number of storms.

ON-DEMAND EXPLANATION

The overview section provides a high-level summary of analysis results and does not include a description of every subregion. Therefore, in addition to the tooltips revealing exact values, IMR generates additional descriptions for every subregion. Users can click any subregion to acquire these details, which are then displayed below the overview section (in the right panel).

Here, the first sentence compares the values of the *focus* and *context* variable for the selected subregion with the respective average values across all subregions. If the selected subregion is among one of the extreme cases, it is stated by using quantifiers such as “*highest*”, “*lowest*”, and “*most*”. For instance,

Table 4.3: Parameter configuration for shown examples

Fig.	Title	Region	Subregion	Focus		Context		Situation	Causality
				Name	Type	Name	Type		
4.9	Fatalities caused by storms, USA, 2017	USA	states	deaths	casualties	storms	incidents	negative	yes
4.14	Average life expectancy and spending on health, Europe, 2018	Europe	countries	average life expectancy	demographic-health indicators	health expenditure	monetary	positive	no
4.15	Adolescent birth rates and use of Internet, World, 2015	World	countries	adolescent birth rate	demographic-indicators	Internet users	percentage	neutral	no
4.16	Obesity and consumption of alcohol, World, 2010	World	countries	obese people	percentage	alcohol consumption	indicator	negative	no

in the case of Texas, this sentence reads: *“Texas experienced the highest number of deaths (184) and highest number of storms (3,621) among all states of the United States of America.”* The next sentence states the statistical ranking of the selected subregion with respect to the *focus* variable. The last sentence provides a comparison of the selected subregion with its neighboring regions for highlighting similar or dissimilar statistics. For instance, the state of Utah is the only state among its neighboring states that does not report any casualties.

Besides the explanations on one subregion, users can compare any two subregions by simultaneously selecting them on the map. Here, the generated text consists of a single sentence that contrasts both regions based on the values of both variables. For instance, Figure 4.15 and Figure 4.16 presents two different instances of comparison texts.

4.3.5 Application Examples

This section presents several examples to demonstrate the usefulness of IMR. Through these examples, it is shown that IMR (i) detects outliers, regional differences, and prominent patterns reliably for various datasets, (ii) produces meaningful textual descriptions about the analysis results, and (iii) adapts to different variable types and different levels of subregional granularity.

We demonstrate map reports for three different *regions*: the world (Figure 4.15 and Figure 4.16), continent (Figure 4.14), and country (Figure 4.9); and two different *subregional Levels*: countries and states. Table 4.3 shows the values of the user-defined parameters for the presented examples. At the *world* level, the report describes the group of countries showing distinct behavior. For instance, European countries have higher numbers of internet users and lower adolescent birth rates* in comparison to the rest of the world. On the continent level, Figure 4.14 reports the differences among various parts of Europe—countries in Southern Europe have better average life expectancy despite spending less on health. At the country level, in addition to describing the differences across various states of the country, the report also highlights the states showing dissimilar behavior in contrast to their adjacent

*“The adolescent birth rate measures the annual number of births to women 15 to 19 years of age per 1,000 women in that age group. It is also referred to as the age-specific fertility rate for women aged 15-19.” – definition by United Nations

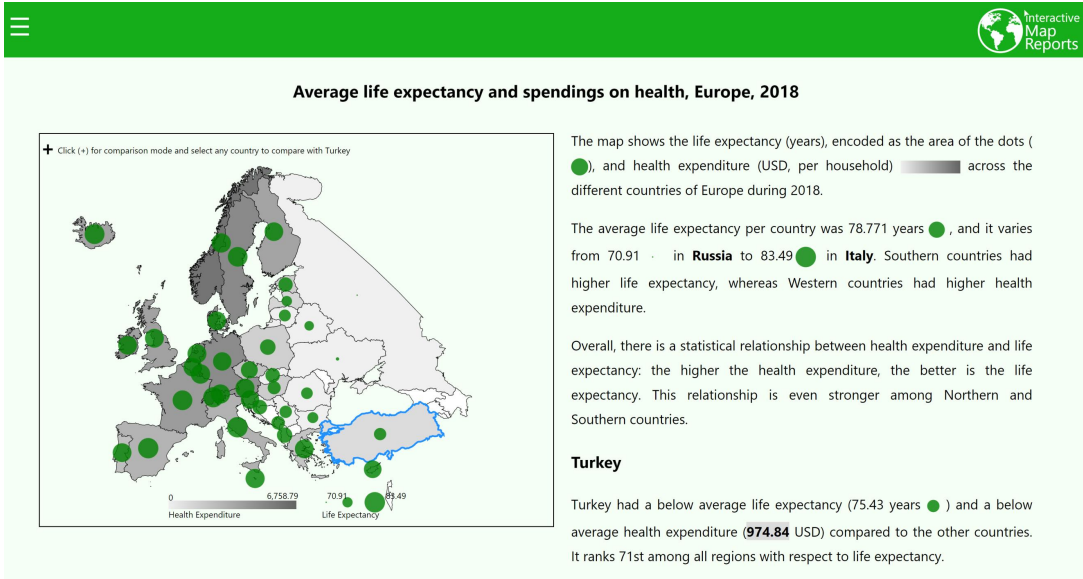


Figure 4.14: An interactive map report describing average life expectancy and health expenditure across Europe in 2018.

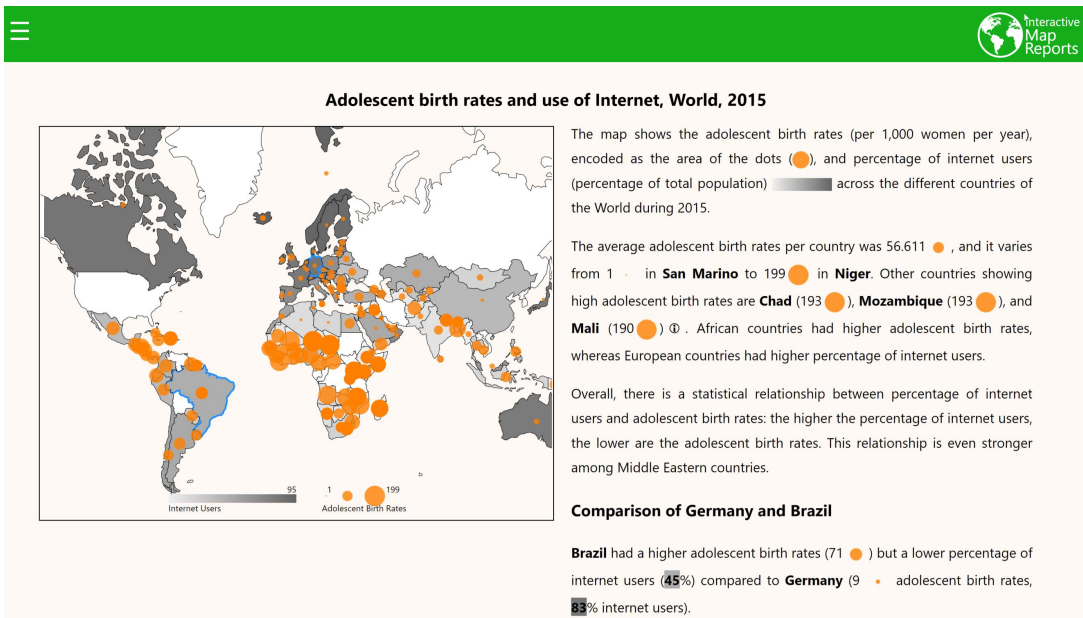


Figure 4.15: An interactive map report showing the possible relationship between adolescent birth rates and the percentage of Internet users across countries of the world in 2015.

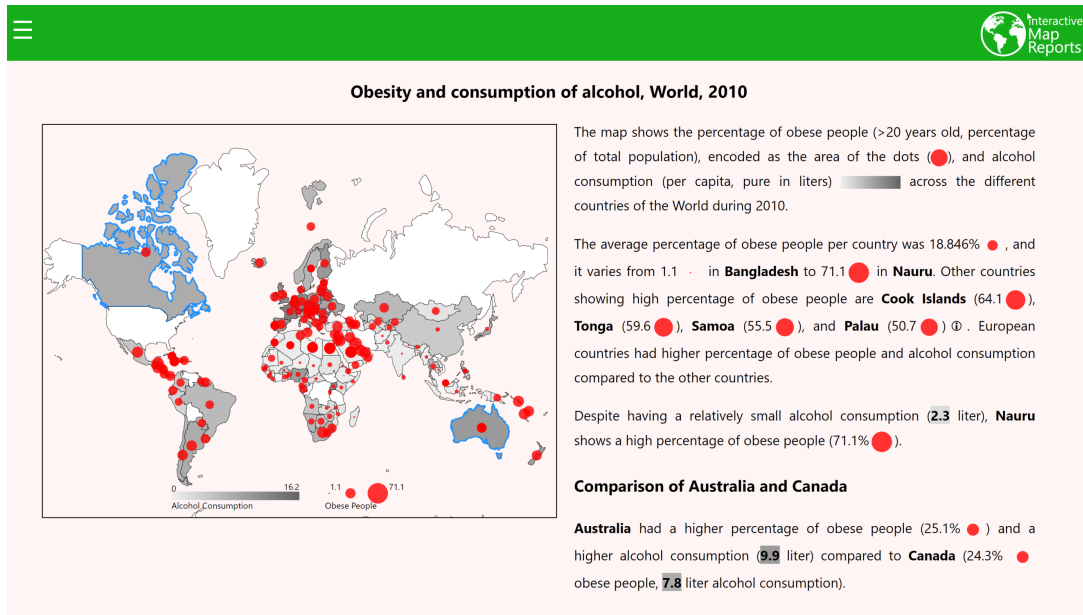


Figure 4.16: An interactive map report describing the percentage of obese people and alcohol consumption in the world during 2010.

states. For instance, Figure 4.9 shows that the states of Missouri, Nevada, and Wisconsin suffered a lot more deaths than their neighbors.

To show the adaptability of the generated text to various situations, we showcase examples for each type of *situation*. Figure 4.15 highlights the relationship between adolescent birth rates and a number of people who have access to the internet. The adolescent birth rate (*focus*) is neither clearly positive nor negative, thus this report is generated according to the **neutral** situation. The map reports shown in Figure 4.9 and Figure 4.16 are produced with the **negative** value of the *situation*. For instance, the phrase “*suffered a lot more casualties*” reflects the negativity of the situation (Figure 4.9). Although both examples share the same value of the *situation*, the narrative differs considerably based on the variable types and the presence of correlations. The former example highlights the presence of a positive correlation among Southern states, while there was no considerable correlation for the entire country. In contrast, there is no paragraph about correlation in Figure 4.16 as the value is not large enough. Figure 4.14 presents the average life expectancy and the money per household spent on health. Here, higher values of life expectancy are favorable, so the *situation* is **positive**. The phrase, “*better is the life expectancy*”, reflects the positive character of the situation while describing the higher values of the *focus* variable.

Besides the *situation* parameter, the choice of quantifiers and verbs also depends on the variable type. For instance, Figure 4.9 uses the *casualties* as the type of the *focus* variable and hence the phrase “*number of*” is used. Similar is the case with variable type *percentage* in Figure 4.15 (“*percentage of internet users*”) and Figure 4.16 (“*percentage of obese people*”). Referring to Figure 4.14, the choice of quantifier “*values of*” for the variable *health expenditure* is based on the variable type *monetary*.

However, the variable type *demographic-indicators* does not need any phrase as seen in the examples of Figure 4.14 and Figure 4.15 (“*African countries showed higher adolescent birth rates*”). In Figure 4.9, the verbs “*suffered*”, “*experienced*”, and “*faced*” correspond to *Focus Type* ← *casualties*. Interested readers are invited to explore more examples in the online tool at <https://mrshahidlatif.github.io/interactive-map-reports>.

4.4 Discussion

We have demonstrated the application of our automatic generation approach in two diverse scenarios, each resulting in an interactive system. Through several examples in each case, it was shown that both VAP and IMR fulfill the required information needs and guide through the analysis insights, as well as provide additional explanations for interpreting the data. Reflecting on and thinking beyond the two presented systems, this section discusses the strengths and limitations of the approach, as well as makes suggestions for the application and generalization of the approaches to other domains.

4.4.1 Quality and Accuracy of Information

One of the biggest challenges while applying the automatic generation approach to a specific application is to ensure the quality and accuracy of generated information. For VAP and IMR, we tried to assure the correctness by only showing information that is available in sufficient quality and quantity. For instance, VAP does not speculate on the research topics of an author if keyword information is only available for a single publication (see the profile of El-Assady in Figure 4.7). Likewise, IMR leaves out the correlation if its value is not statistically significant. Of course, such checks can also be included into other approaches. However, in a tabular or pure visual representation, missing information may cause confusion to some users. In text, it seems more natural to only focus on the things that are most relevant and reliable. A counterargument might be that this selection takes away the user’s control, but we counterbalance it by providing explanations and making the underlying data available on demand. This way, users are not only restricted to what the narrative tells them, but they can also explore.

Another problem with auto-generated reports like ours is that the possibility of incorrect information being generated cannot be excluded. However, one might argue that the same problem may apply to and endanger visualizations to some extent as well, especially when the mapping between data and visual elements is complex. The problem is more severe with auto-generated text as it is more explicit. Like for any complex software system development, one possible countermeasure is to apply thorough testing to avoid the generation of false information. Since incorrect information (in text) mostly stems from heuristic-based algorithms (e.g., Catherine Plaisant was falsely classified as a supervisee of Ben Schneiderman; see the third paragraph in Figure 4.1), lucid communication on how these heuristic-based algorithms work helps in making the representation transparent. Another possible solution could be to communicate uncertainty in the generated information, for instance, including visual hints to convey how confident the approach is on a certain finding; we leave this for future work.

4.4.2 Usefulness and Usability

Both visual and textual representations have their own advantages and augment each other when interactively blended together. Prior research already provides evidence that a bimodal representation can be beneficial for understanding and interpreting the presented information: Gkatzia et al.⁵⁰ demonstrate better decision-making under uncertainty using a task-based study. Similarly, Sripada and Gao¹⁶² claim that divers find the bimodal representation more comprehensive while judging the safety of a deep dive. However, their results are based on specific datasets and may not generalize. Although we have not evaluated our approach in a user study with actual target users, the usefulness of VAP and IMR is shown through several application examples. These examples showcase that both systems produce meaningful reports and fulfill initial information needs. Using VAP, we also simulate two realistic scenarios (S1 – Recruiting and S2 – Identifying Experts) to demonstrate how an analyst would benefit from the system; see details in paper⁹⁵.

Therefore, we cannot yet claim that our approach performs better than a pure visualization-based representation. For such comparison, we shall need to first develop the other representations showing equivalent information and optimize each as we have optimized the presented data documents (VAP and IMR). Only then, we can perform a user study comparing the three representations: pure visualization, pure textual, and bimodal (our) representation. While a quantitative study could answer which of the representations is best in terms of accuracy and answer times, a qualitative approach could also reflect on how intuitive and self-explaining the representations are and how end users work with them. Furthermore, textual explanations can influence the way users interpret visualizations. Whereas a recent study⁸⁶ investigates the effect of diagram titles on the interpretation of visualizations, the impact of longer textual explanations on the accompanying visualizations remains yet to be explored.

4.4.3 Generalizability

In contrast to most previous systems (e.g., GALIWeather¹³⁵, WIP¹⁷⁸, Q2Q¹²⁶, and others discussed in Chapter 2) which generate textual descriptions for a very specific scenario, our focus is on producing integrated visual and textual representations of comparatively complex data (e.g., bibliographic data compared to a simple time series weather data). While the approach greatly depends on an application scenario for which it is to be applied, the design principles and different components are generalizable to many application domains and scenarios; for instance, the selection of important data items (Equation 4.1), and *document integration* are applicable to any dataset. Although text generation requires thorough rework for each application scenario; IMR provides some flexibility in text generation within a specific scenario or for somewhat similar scenarios.

For instance, in the VAP system, despite some tailoring for the visualization community, most of the generated text would still work for other DBLP publication data. Increasing with the difference in publication culture, more adaptation would become necessary when switching to other fields. For instance, some research communities might use the alphabetic ordering of authors that renders our current algorithm unable to identify supervisor relationships. Still, the seniority check in our approach (supervisor is at least five years senior to the supervisee) holds across many publication cultures

and might convey this information to some extent. Whereas developing a system uniting various sciences might be challenging, tailored solutions for specific fields are straightforward to derive. Only the phrasing of the text and calibration of algorithms would need to be adapted. Similarly, we can also build profile pages for actors, musicians, software developers, or other people. The only requirement is that these people work together and contribute to specific work items.

The use of a small set of configuration parameters in IMR gives sufficient flexibility to adapt the same text generation process for different datasets of the same type. Hence, our approach can be considered as lying between the fully automated text generation systems (which are tailored to a narrow scenario) and tools that allow for building fully customizable templates. However, our approach does not support manual refining or extending the text templates beyond the configurations that can be specified through the parameters, as opposed to commercial tools like *Wordsmith* and *Arria NLG Studio*.

4.4.4 Applicability and Extension

The automatic generation approach greatly depends on the scenario and dataset. The approach requires substantial effort to tailor it for a specific application scenario. However, once tailored, the approach gives enough flexibility for modification and extensions. Unlike pure visualization approaches, where adding more data aspects could result in a more cluttered representation, the extension of a textual description usually does not reduce how self-explaining or understandable it is. For instance, VAP did not include related information such as affiliations, citation information, or awards due to the non-availability of such data reliably. Adding paragraphs for this information to the author profiles would be a natural and easy extension. Likewise, IMR has chosen a specific geographic visualization to encode the bivariate data, it would be relatively easy to replace the visualization with a different one or even make it customizable. An option to achieve it could be the use of the comprehensive declarative model for producing visualizations.⁷⁵

In our approach, we use the generated textual representation as a primary entity to demonstrate the capabilities of this rarely leveraged bimodal representation. However, we do not argue that our representation needs to replace conventional visualization or table-based approaches. On the contrary, the generated explanations could simply be made part of the existing visualization systems. For instance, as an introductory text of an author profile in a digital library system, as a detailed description of an author node in a co-author network visualization, or as a description of bivariate outliers in a map visualization. Following this idea, our technique could enrich existing approaches without the threat that the additional information would clutter the interface as text grows linearly and does not lose its intuitiveness, unlike visualizations that can become overloaded if too much information is added. Moreover, the generated texts are rather likely to make a formal representation more self-explaining and understandable.

5

Explorantation – Exploration and Explanation

Explorantation is a new science communication paradigm introduced by Ynnerman, Löwgren, and Tibell.¹⁸⁹ The term stems from **exploration** and **explanation**, which used to be two clearly distinguishable objectives of a visualization. While the former enables efficient and rapid visual data analysis (e.g., traditional visual analytics systems), the latter (data-driven storytelling¹⁴³ or narrative visualization¹⁵¹) focuses on communicating the analysis findings to the general public. In the context of scientific communication, Ynnerman et al. showcase a variety of explorantative examples; for instance, to show visitors of a museum a visualization of an ancient mummy, then guide them to interesting aspects of this visualization but also let them explore it. A similar earlier work by Weiskopf and others¹⁸¹ combines visual explanation and exploratory and illustrative visualizations for communicating Einstein’s theory of relativity to non-experts.

Akin to the concept of explorantation, Bret Victor¹⁷⁷ presents the idea of **explorable explanations** that argues for *active reading*⁶. According to Victor, an active reader “*asks questions, considers alternatives, questions assumptions, and even questions the trustworthiness of the author. An active reader tries to generalize specific examples, and devise specific examples for generalities. An active reader doesn’t passively sponge up information, but uses the author’s argument as a springboard for critical thought and deep understanding.*”

We adopt the ideas of *explorantation* and *explorable explanations* for visual analytics scenarios. The overarching objective is the broad dissemination of analysis results as an interactive visual representation (interactive data document) that enables explorantation in an accessible way. Orthogonal to Ynnerman et al., we work extensively with multiple media like text and audio to provide explanations (as opposed to other visual views in Ynnerman’s examples¹⁸⁹). Some of their design principles are also applicable to our scenario, in particular, the *explorative microenvironments* blended with *sign-posted narratives* results in an intriguing representation. Accordingly, we interweave explorative parts

through annotations, textual explanations, and visualization–text linking into an overall self-directed explorative representation. With respect to the visual representations of data, Victor’s suggestions for *explorable explanations* are even closer to our work, as they also focus on interactive documents. We employ *explorable examples* and *contextual information*, as two of three suggested categories of interactive data representations enabling active user behavior.

This chapter first extends interactive data documents with respect to explorative analysis as a step toward blurring the borderline between explanation and exploration (Section 5.1). In the proposed explorative data representation, users are not restricted to passively consuming explanations, but they are facilitated to actively explore the data. In contrast to traditional visual analytics interfaces, which focus solely on exploration, more guidance is provided to the users in the form of textual explanations while preserving the explorative power of visualization; they can read with a specific focus, break the content apart, and analyze its meaning. While Section 5.2 directly instantiates the concept of explanation to a software engineering application, the next two sections (Section 5.3 and Section 5.4) can be considered as translations of this concept to data representation involving chatbots and virtual reality environments, respectively.

5.1 Explorations in Interactive Data Documents

The overarching objective of the *interactive data documents* is to support active reading⁶ or active user behavior.¹⁷⁷ While *VIS Author Profiles*⁹⁵ and *Interactive Map Reports*⁹⁴ (described in Chapter 4) present interactive representations of data, they have a substantial focus on explanations; the exploration is very restrictive (especially for *Interactive Map Reports*) and mostly originates from the textual content. The objective, here, is to move toward a visual analytics solution that retains its characteristic extensive exploration capability while providing users with explanation and guidance as they explore the data. This section describes the general approach that is applicable (in particular) to multivariate data in a broader sense and might even extend to wider classes of interactive data representations (e.g., virtual reality visualizations). The approach comprises the following three building blocks:

I – TEXTUAL EXPLANATIONS

The approach uses automatically generated text as the main explanatory medium. The textual explanations are categorized into three types: (i) **Data-driven explanations** summarize the results of data analysis (e.g., identification of patterns, clusters, and outliers) and point to remarkable observations as well as give characteristic examples. (ii) **Educational explanations** provide background on the domain concepts reflected in the data that may not be well-known to the target audience. (iii) **Methodological explanations** give details about how the analysis was performed, what heuristics were employed, and the reason why the system came to certain conclusions. Such explanations contribute to increased transparency and allow for the validation of data-driven explanations.

The textual explanation is the main feature that distinguishes our approach from conventional visual analytics solutions. The data-driven explanations are the focus of the explorative document and serve as an independent and connected view, just like other visual views in a visual analytics sys-

tem. The two other kinds of explanations provide important context and are mostly available on demand. Users can obtain them if they require background information for any part of the generated data summary. Chapter 4 already explains the automatic generation methods for generating these textual explanations.

II – EXPLORABLE VISUALIZATIONS

The explorative component of (explorative) interactive data documents is mainly contributed by visualizations. Here we discern two types of visualizations. (i) **Overview visualizations** (*vis*)—as evident from the name—provide an overview of the data and should have a consistent location in the interface. They should be visible to the user at all times. As with the presence of data-driven explanations, it is important that these visualizations do not get scrolled out of the users’ view. The layout presented in Section 4.1 enforces it through the use of coordinated multiple views. Any appropriate visualization of data (e.g., parallel coordinates, scatterplot matrices, or multivariate glyphs for multivariate data) can be employed to provide the overview.

In contrast to overview visualizations, the (ii) **embedded detail visualizations** (*emvis*) enrich the text with further information and just show subsets or different aspects of the data. These may include regular (large) visualizations that scroll with the corresponding text or word-sized graphics embedded inside lines of text. The better these visualizations are integrated with textual explanations, the easier it will be for the users to explore them along reading the text. The exploration process happens visually by users deciding to look at and investigate certain elements of the visualization, followed by interactions to subselect the data and pick out individual elements for further inspection.

III – CONSISTENT LINKING

Similar to a traditional multiple-coordinated-view visual analytics system, a challenge is to maintain a clear and consistent linking between different views. The linking becomes even more difficult in our case as data is described on various levels of abstraction and with different modalities (text or audio and visualization). We apply the concept of interactive (i) **vis–text linking**. This linking refers to providing visual cues (e.g., visual highlighting) to guide users’ attention to the relevant portion of a visualization while interacting with the corresponding text fragments and vice versa. To connect textual and visual descriptions, consistent interactive linking is important. We build on the concept of *vis–text interaction* introduced by Beck and Weiskopf.¹⁸ Figure 5.1 shows an abstract representation of our visualization–text linking model. It envisions an interactive data document consisting of overview visualization (*vis*), embedded word-sized visualization (*emvis*), and textual explanations (*text*) as already explained in Chapter 4. These representations are interactively linked to each other in a bi-directional way. The model discerns between two types of interactions: The local interactions that only locally impact a component (*text–text*, *emvis–emvis*, *vis–vis*). Examples include details on demand or filtering data in a visualization. The global interactions, on the other hand, are intended to link the various visual representations (e.g., *text–emvis*, *vis–text*). Through bi-directional *visualization–text linking*, text and visualizations become a coherent and integrated unit of information and the resulting docu-

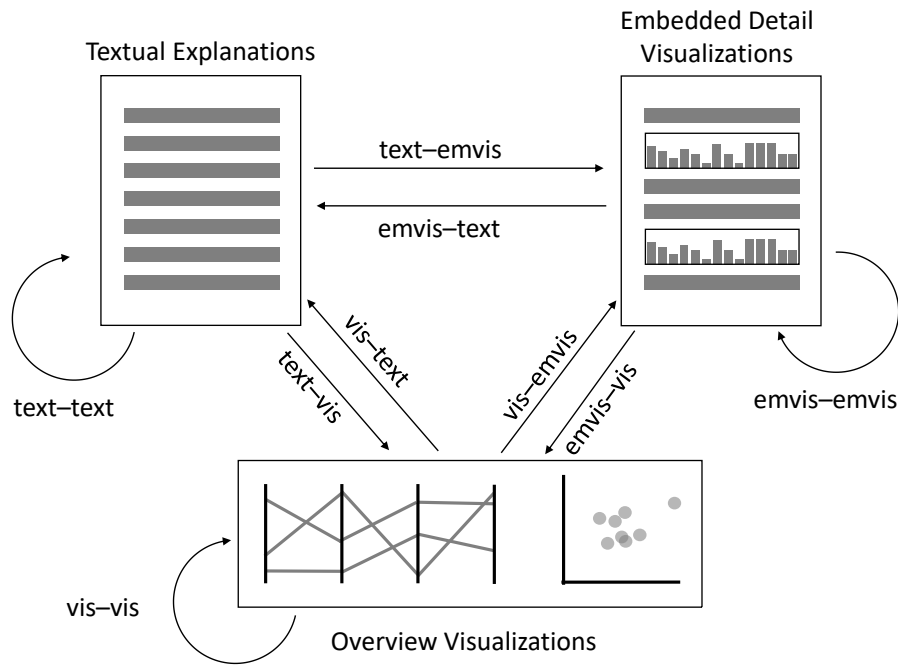


Figure 5.1: Abstract representation of the interaction model. We have bi-directional interactions among text, embedded visualizations (emvis), and overview visualizations (vis).

ment supports easy and guided exploration of the presented content. In standard reading strategy—reading the text first and then exploring the visualizations while going through the text—linking helps in quickly seeing and collecting the *text-vis* references. Besides, it would also offer alternating reading strategies such as exploring the visualization first, and then reading the corresponding parts in the text that are visually highlighted through this linking.

The approach further uses a consistent (ii) **color coding** to clarify relationships between the different textual and visual descriptions of related data. We suggest applying a consistent color coding of the different variables across all representations. For instance, similar variables can be grouped by hue and get assigned a different brightness. Finally, the (iii) **positional linking** places content that is related in close physical proximity. Having fixed panels and a consistent visual platform to hold information helps in seeing global interactions more clearly as everything is in the users’ viewport.

With the aforementioned components of a data document, we aim at supporting active reading and the explanation. This explanative representation can also cater multiple groups of users with varying levels of expertise and experience. Within such documents, at first, users are confronted with a textual summary and associated overview visualizations. One group of readers, especially first-time users, might follow mainly the provided narrative and start reading the explanations from the beginning. Whenever they find something unclear from the high-level summary, they can explore the required background (I.ii and I.iii) on demand. After having read the summary text, they may switch to exploring the data using visualizations. On the contrary, another group of users—who are more

experienced—could directly start the data exploration process. While some information can be directly gained from the visualizations, reading the textual explanations for other insights can provide further support. Likewise, the textual summaries might also point them to interesting findings that they might have missed otherwise.

5.2 Explorations: A Concrete Application Example

This section instantiates the concept of explorative documents for multivariate data, with its application to a concrete and specific example of software engineering. This resulted in a visual analytics solution, namely *Code Quality Documents*. It presents the source code analysis of a software project.*

The Code Quality Documents is a full-fledged system[†] developed for the analysis of the source code quality. It was developed for less technical stakeholders like product owners, project managers, or anyone interested in better understanding the quality of source code as well as for supporting software developers. The system was developed in close collaboration with software engineers and visualization experts. This section skips such extensive details (interested readers can consider reading the paper¹²⁵) and discuss the explorative aspect of the Code Quality Documents. However, the necessary information on the analyzed dataset and code quality metrics is included. Afterward, this section gives an overview of the interface, explains the individual components, and finally describes the interaction model that links the different components.

5.2.1 Code Quality Data and Analysis

Our target user group is broader than just the technical stakeholders or highly experienced developers. That is why integrating guidance and explanations will provide valuable support for most of the users and allows them to draw actionable conclusions. However, simply presenting a static report would not suffice because the prepared summaries and explanations can only be a starting point for investigating a detected problem in detail.

Data – The system utilizes 11 software metrics defined at class-level in object-oriented software projects (Table 5.1). These metrics provide significant information on code quality and are employed to measure and quantify different code quality aspects.^{114,12,29} *Code Quality Documents* focuses on four quality aspects: complexity, coupling, cohesion, and inheritance. Table 5.1 shows the associated metrics for each quality attribute along with their acronyms and brief descriptions. The metrics listed in the “Other” category reflect general properties like the size of a class or the history of bugs in a class. Some metrics in this category are required to detect code smells. Code smells provide information on implementation decisions or choices that might degrade code quality.¹⁹²

*The research project¹²⁵ was led by Haris Mumtaz (the first author), a PhD student working in the field of software visualization at VISUS, University of Stuttgart. I was involved in the project as the second author and was mainly responsible for the presentation and communication aspect of the analysis results through the integration of automatically generated explanations and visualizations to support exploration. While Haris conducted the code quality analysis (briefly described in Section 5.2.1, I focused on summarizing analysis results as automatically generated textual explanations, conceptualizing and implementing the interactive linking model, and extensively worked to make the representation explorative.

[†]The system is Web-based and available online at: <https://mrshahidlatif.github.io/code-quality-reports>.

Table 5.1: Eleven class-level software metrics (name and acronym) used for code quality analysis, grouped by quality attributes.

Quality Attribute	Software Metric	Acronym	Description
Complexity	Weighted methods per class	wmc	The sum of all method complexity values for a class.
	Maximum cyclomatic complexity	max_cc	The maximum of all the method-level complexity values of a class.
Coupling	Afferent coupling	ca	The number of other classes that depend on a class (incoming dependencies).
	Efferent coupling	ce	The number of other classes on which a class depends (outgoing dependencies).
Cohesion	Lack of cohesion of methods	lcom3	It checks whether the methods access the same set of variables of a class.
Inheritance	Depth of inheritance	dit	The inheritance levels for a class.
	Number of children	noc	The number of immediate descendants of a class.
Other	Average method complexity	amc	The average size of the methods in a class.
	Lines of code	loc	The total lines of code present in a class.
	Number of public methods	npm	The number of methods declared as public in a class.
	Number of bugs	bug	The number of bugs that have been associated with a class.

Analysis – The system uses thresholds (similar to the work of Filó et al.⁴³) to rate code quality (as good, regular, or bad) with respect to four quality attributes and highlight the severity level of the problem (high, medium, or low). Again based on related work¹³², we detect four types of common class-level code smells: *large class*, *functional decomposition*, *spaghetti code*, and *lazy class* using class-level metrics. A *large class* is one that has many fields and methods, resulting in many lines of code. A class with many private fields and methods is associated with *functional decomposition*. A class with *spaghetti code* has long methods without proper structure. A class with little to no functionality is a *lazy class*. Since we have class-level metrics, it is possible to compute these code smells using predefined thresholds.

Based on the metrics and the analysis, the content of the code quality document comprises three parts: first, quality attributes covering coupling, complexity, cohesion, and inheritance; second, code smells in terms of *large classes*, *functional decomposition*, *spaghetti code*, and *lazy classes*; and third, information about bug history.

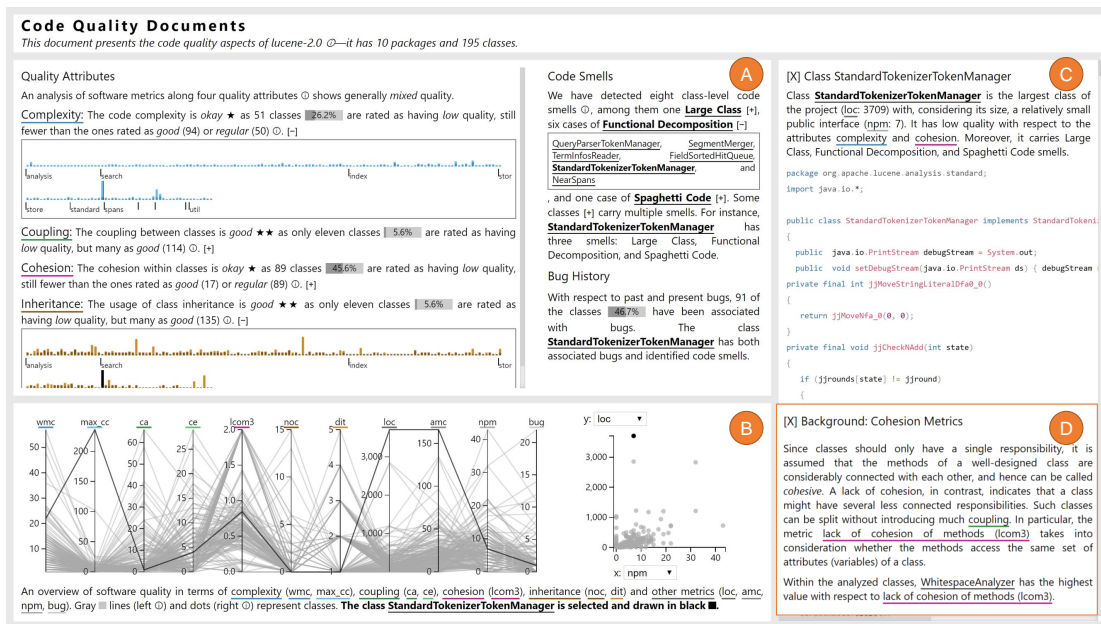


Figure 5.2: Explorative code quality document for *Lucene 2.0*. (A) Textual overview in terms of quality attributes, code smells, and bugs, which includes embedded visualizations. (B) Overview visualizations: parallel coordinates plot and scatterplot. (C) The source code of a class is provided in the details view. (D) Description of a quality attribute alternatively presented in the details view.

5.2.2 The System

The system, *Code Quality Documents*, is designed as a multi-view interface with three different panels: two for overview descriptions (summary text and visualizations) and one for details. The system maps the building blocks described in Section 5.1 to the different panels of the document structure; Figure 5.2 shows a specific example. The summary panel presents the main data-driven textual explanations (I.i) summarizing the results of code quality analysis. Besides, it contains embedded visualizations (II.i) and methodological explanations (I.iii) that are retrievable on demand. The panel for the overview visualizations (II.i) contains a parallel coordinates plot and a scatterplot. The details view provides educational explanations (I.ii) or class details (e.g., source code) on selection.

TEXTUAL EXPLANATIONS

To automatically produce textual descriptions, we employed the same text generation process as already described in Section 4.1.2. The data-driven text (I.i) is split into three categories (Figure 5.2 A) and describes code quality along quality attributes, code smells, and bugs. The overview text also contains detailed embedded visualizations and a list of classes corresponding to specific categories of code smells. These details are not expanded in this panel by default. A plus icon [+] indicates their presence and allows for expanding. Likewise, an info icon ⓘ hints at the availability of methodological explanations (I.iii). Hovering over this icon presents a tooltip that describes the exact methodology

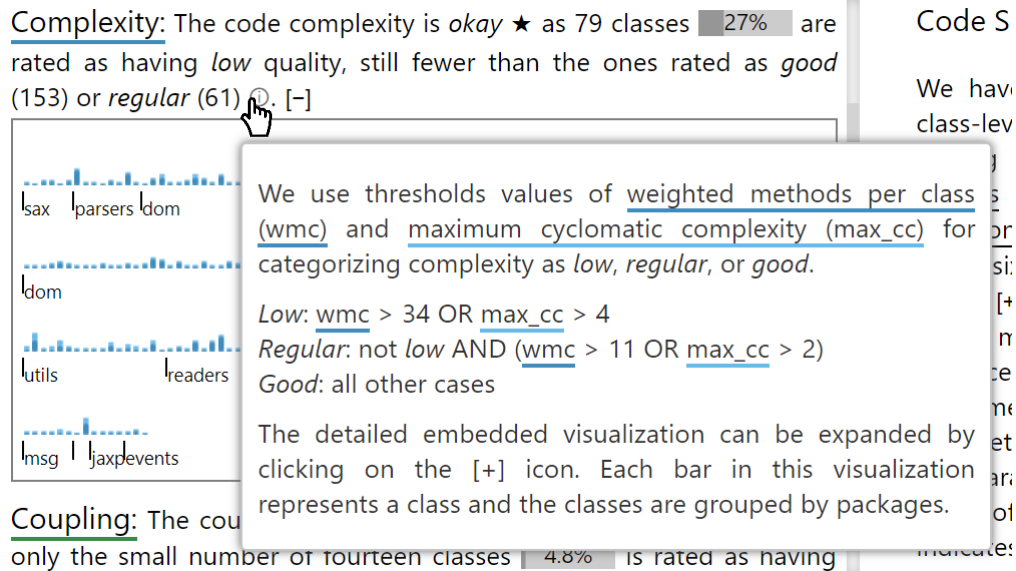


Figure 5.3: Methodological explanation of classifying the complexity of a source code project in terms of low, regular, or good.

that was employed to come up with the corresponding detail. Figure 5.3 shows an example displaying the software metrics and their thresholds used to classify low, regular, or good complexity. The reporting of different quality issues along with methodological explanations can assist users in quickly understanding the core issue and making decisions to improve the quality.

The names of the quality attributes (e.g., complexity) and the software metrics carry a thick colored underlining at the bottom (e.g., wmc, max_cc) to provide a quick reference about which metrics and quality attributes are related. Educational explanations (I.ii) provide background information on these quality attributes and code smells. Interacting with these terms brings up educational explanations in the details panel. Beside standard definitions of the terminologies, the description includes project-specific examples to better communicate the concept in the current context. For instance, Figure 5.2 D shows the educational explanation for the quality attribute cohesion; WhitespaceAnalyzer is provided as an example of a class having the highest value of lack of cohesion of methods (lcom3). Likewise, explanations on code smells can be accessed. Clicking on a class anywhere in the system opens the source code of that class in the details view preceded by a class-specific explanation providing a short summary of problems in the class, if there are any; Figure 5.2 C gives an example.

The system makes extensive use of conditionals to describe different cases and provide reasons for different analysis results. For instance, it not only lists the number of classes that have low quality with respect to any of the four quality attributes but also explains reasons for the specific rating. The same rating can even have different reasons (Figure 5.5: “The code complexity is okay as 79 classes are rated as having low quality, still fewer than the ones rated as good (153) or regular (61).” and “The usage of class inheritance is okay. Although not a high number of classes (13) are rated as having low quality,

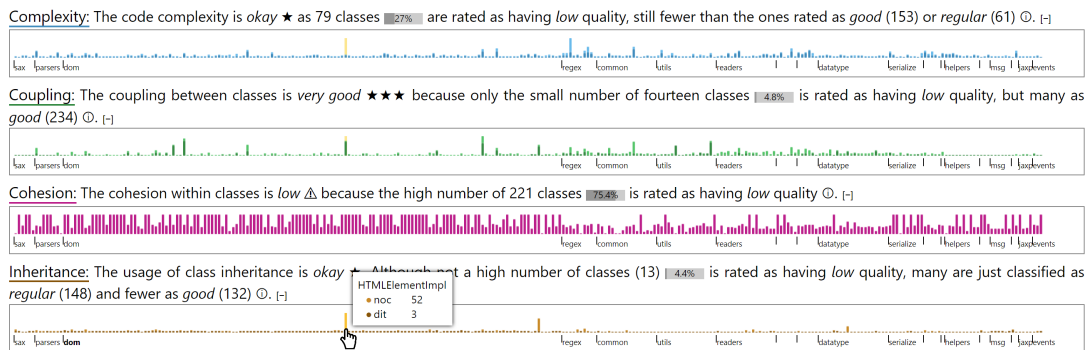


Figure 5.4: Low quality in `HTMLElementImpl` found through active exploration of the embedded detail visualizations of coupling, cohesion, and inheritance.

many are just classified as *regular* (148) and fewer as *good* (132).” The system leaves out sentences if no results are available and produces a special text in such cases. For instance, a special case for this paragraph is when no code smells were found: “We have not detected any class-level code smells in the project—congratulations!”

EXPLORABLE VISUALIZATIONS

To offer data exploration and provide context behind the textual descriptions, the system provides two overview visualizations (II.i): a parallel coordinates plot and a scatterplot (Figure 5.2 B). These visualizations are useful in discerning important patterns and relationships between code quality metrics.¹⁵⁰ Users find these visualizations useful in better understanding the code quality and comparing the properties of different classes across all (parallel coordinates plot) or a subset of metrics (scatterplot). To obtain an overview of all the metrics, the parallel coordinates plot is helpful, whereas the scatterplot supports the identification of relationships between two metrics and seeing if there are outliers.

Besides these overview visualizations, the embedded detail visualizations (II.ii) complement the data-driven text generated in the document (Figure 5.2 A). The small bar charts are employed to represent the metric values of each quality attribute for all classes. The classes are structured with respect to the packages they belong to (Figure 5.4). Furthermore, the word-sized bar charts in line with sentences (e.g., 27%) indicate percentage values to give a quick impression of the quality. These values always refer to problematic cases (e.g., low quality or bug-prone classes) and are given relative to the overall number of classes. Likewise, small star ratings on a scale of 1–3 (e.g., ★★★) and warning symbols provide a quick hint of the respective rating for each quality attribute.

LINKING AND INTERACTIONS

To connect textual and visual descriptions, consistent interactive linking (III.i) is important. The components or views in the system—textual explanations (*text*), overview visualizations (*vis*), and embed-

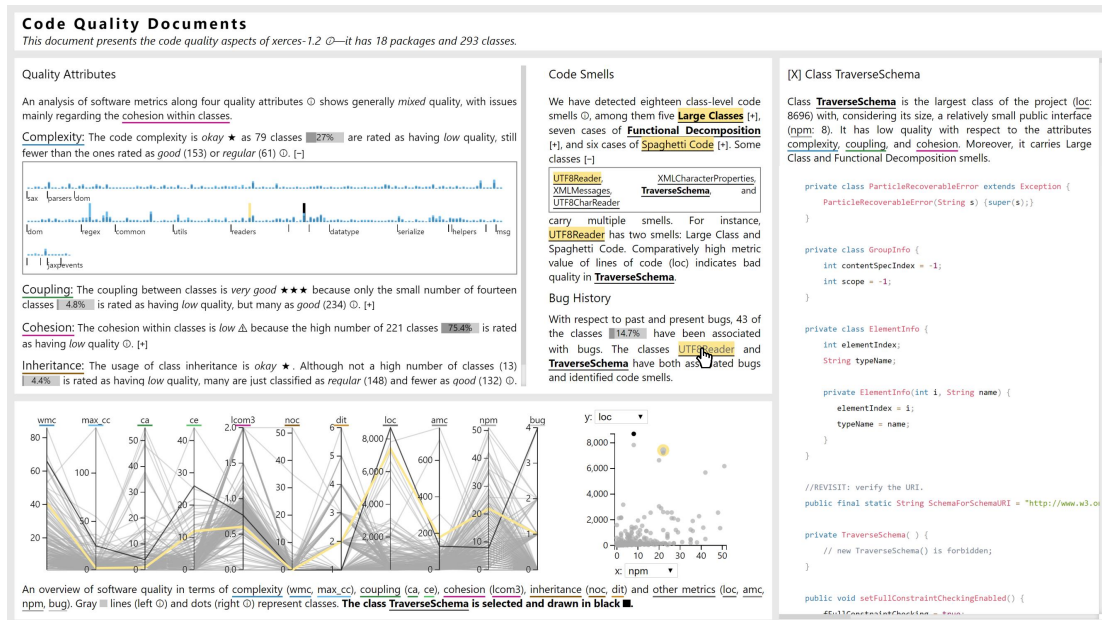


Figure 5.5: Various instances of vis-text interactions. A persistent highlighting (click on *TraverseSchema*) marks the related elements with bold font in the text (text-text), a black line in the parallel coordinates, a black dot in the scatterplot (text-vis), and black background in embedded detail visualizations (text-emvis). Similarly, a non-persistent highlighting (hover on *UTF8Reader*) marks the corresponding elements in yellow. The details panel shows the class-specific description and source code of the persistently selected class, *TraverseSchema*.

ded visualizations (*emvis*)—are interactively linked to each other in a bi-directional way as shown in Figure 5.1.

As these components contain different representations of the same class-level multivariate data, an essential interaction is the brushing and linking of data points across all representations. Hovering over a class name anywhere in the text triggers a transient (non-persistent) selection. It highlights the corresponding polyline in the parallel coordinates plot (*text-vis*), the dot in the scatterplot (*text-vis*), the bars in the embedded visualizations (*text-emvis*), and other occurrences of the class name in the text (*text-text*). Figure 5.5 shows the effect of hovering over the class name *UTF8Reader*; the linked parts are highlighted with yellow color. Apart from the other instances of the hovered class in text, we also mark the corresponding code smells that the selected class possesses. The *large class* and *spaghetti code* are highlighted, since *UTF8Reader* contains both code smells (Figure 5.5). Hovering over a bar in the embedded visualization and a dot in the scatterplot has a similar effect and triggers *emvis-vis/vis-vis*, *emvis-text/vis-text*, and *emvis-emvis/vis-emvis* interactions.

The transient selection shows as long as the interactive element is hovered and provides a quick way of cross-referencing different representations. To make the highlighting persistent, the interactive elements can be clicked; the parts related to the clicked element are highlighted with black color in the visualizations and with boldfaced font in the text. For instance, Figure 5.5 shows the persistent

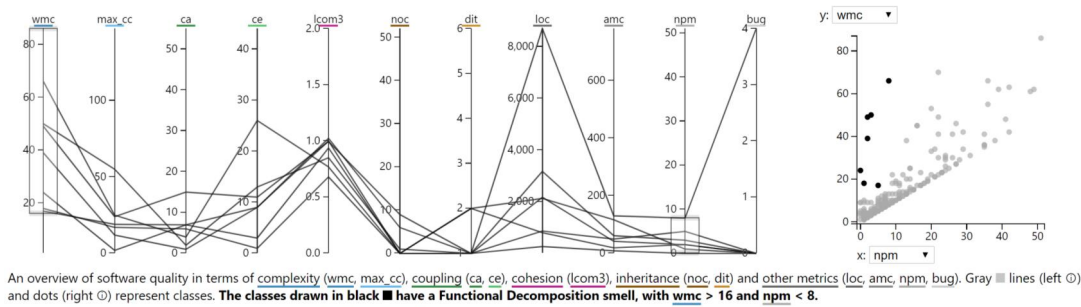


Figure 5.6: A selection of the classes in Xerces 1.2 that have a *Functional Decomposition* smell; they are highlighted in the parallel coordinates plot and the scatterplot. The caption of the visualizations adapts to describe the selection.

selection corresponding to the class `TraverseSchema`. This helps in getting a comparative overview of two different classes with respect to various quality aspects; one glance at Figure 5.5 is sufficient to tell that `UTF8Reader` and `TraverseSchema` have one code smell (*large class*) in common. In addition, users can quickly observe that `UTF8Reader` has less complexity (embedded visualization for complexity), fewer *lines of code* (scatterplot), and fewer bugs (parallel coordinates plot) than `TraverseSchema`.

Clicking on a code smell name, aside from showing an educational explanation in the details view, highlights the set of classes that contain that code smell in the parallel coordinates plot and scatterplot (*text-vis*)—Figure 5.6 shows the result of clicking *Functional Decomposition*. This helps in understanding the pattern of metric values for the classes having different code smells. Since the scatterplot illustrates the relationship between any two software metrics, we update the dimensions of the scatterplot on persistent (click) interactions according to the context. For example, clicking on *Functional Decomposition* will update the scatterplot dimensions to weighted methods per class (`wmc`) and the number of public methods (`npm`) as these metrics are used to identify the smell (see Figure 5.6). Moreover, users can explore the relationships between other metrics.

Since lines are hard to select in a parallel coordinates plot (they are thin and often occlude each other), we provide a persistent range selection on the axes (brushing interaction with mouse press and hold). On every persistent selection, the caption of the figures adapts accordingly to describe the selected elements (see Figure 5.6). In contrast to legends, the textual captions allow for the inclusion of contextual and methodological information (e.g., data filtering criterion), which helps in making the interactive visualizations more self-explanatory.

The system uses consistent color coding to couple the visualized metrics (III.ii). For instance, in Figure 5.5, the colors of complexity metrics in the caption of the parallel coordinates plot match the color coding of the complexity bar chart in the quality attributes section. The metrics are grouped in terms of the quality attributes—two metrics of the same group are associated with the same hue but a different brightness.

5.2.3 Usage Scenario – A User’s Perspective

We demonstrate how explorative Code Quality Documents assist a user in analyzing and understanding the quality of a source code project. This demonstration implicitly reflects on how users can work with our interactive document and the support they will receive from it in performing various analysis tasks. Let us assume that a hypothetical user—Alex, who is a junior software developer—is interested to explore *Xerces 1.2* (an XML handling framework)^{3,111}. In particular, this example demonstrates how textual explanations point Alex to a specific issue in the code, and how exploration helps her in analyzing the problems.

Alex loads the *Xerces 1.2* project data in Code Quality Documents as shown in Figure 5.5. The system identifies 18 code smells and issues, mainly with respect to cohesion within classes. From the textual explanation, Alex reads that the class `UTF8Reader` suffers from the two code smells: *large class* and *spaghetti code*. The class `UTF8Reader` becomes more interesting to her when she realizes that this class has had many bugs in the past. Intrigued to know the root cause of the problem, she looks into the source code of this class. Just scrolling down a little, she finds confirmation of these issues self-admitted by the developers in the class description comment, also providing a reason: “*Some blatant v[io]lation of good object-oriented programming rules, ignoring boundaries of modularity, etc., in the name of good performance.*” This is an example that shows that our approach helps the user focus on the most problematic classes.

Another class, `TraverseSchema`, that contributes to low quality catches her attention while reading the bug history. This class carries two code smells (*large class* and *functional decomposition*). In the class description, she reads that `TraverseSchema` is the largest class in the system and its quality is low with respect to complexity, coupling, and cohesion (see Figure 5.5, details panel). Exploration through consistent linking helps her locate this class in the parallel coordinates plot and the scatterplot (see Figure 5.5, black line and dot); it shows a very similar pattern as `UTF8Reader` (see Figure 5.5, yellow line and dot). She moves on to the source code of `TraverseSchema`, to confirm the existence of a *Functional Decomposition* smell because it has several private but few public methods.

Alex then switches to the quality attribute section to see whether there are any more problematic classes. During an exploration of the embedded bar charts in the quality attributes section, she finds that `HTML_ElementImpl` has a complex inheritance structure. This class has an extremely high number of children (noc) (Figure 5.4; 52 children, while the threshold of low quality is only more than three children). She then heads on to explore the quality of `HTML_ElementImpl` from other quality attributes to see whether there is a pattern. She notices that the related metrics of coupling and cohesion are also high (meaning low quality in these attributes) while the complexity metrics are in the acceptable range (see Figure 5.4).

Finally, moving away from the textual explanations, Alex uses the parallel coordinates plot and the scatterplot to freely explore the interplay of various metrics. She observes an overall positive correlation between the weighted method per class (wmc) and the number of public methods (npm) in the scatterplot. However, a few classes form interesting outliers with high wmc values but low npm values. Unsure but interested in knowing what might have caused this behavior, she selects those classes (points on the scatterplot) and sees the captions adapting to her selections and revealing that these un-

usual characteristics correspond with the *Functional Decomposition* smell (highlighted as black dots in the scatterplot in Figure 5.6, right). As the other metric values for this set of classes are shown accordingly in the parallel coordinates plot (Figure 5.6, left), she observes one of the classes associated with a file size (loc) and a high number of bugs. To her surprise, it is the same *TraverseSchema* class.

5.2.4 Key Takeaways

Finally, we summarize the experience gained in designing the system as a few key takeaways. These takeaways focus on aspects that generalize beyond the specific application of software engineering. Rather, they are applicable to interfaces producing interactive bimodal data representations consisting of generated textual and exploratory visualizations.

Overview always – Shneiderman’s information-seeking mantra¹⁵⁶ emphasizes on overview first and details later for visualization systems. In slight adaptation to this mantra for interactive documents containing text and graphics, we observe that overview visualizations should always stay in view and not scroll with the text. Likewise, longer textual explanations that provide a high-level summary act as anchors for the users to and from the visualization; these should also always stay visible. These overview elements are required for interactive *vis-text linking* and highlighting to work properly: hovering an element anywhere in the interface, related items in the overview representations get visually highlighted. The always visible overview representations along with a stable layout offers a reliable skeleton for the user and allows other content to change dynamically without confusing the user’s mental map.

Consider brushing text – From a strict visualization perspective, textual explanations in an interface appear disconnected and boring. Since they do not belong to a visualization and, therefore, cannot be interacted with. However, our approach has shown that text can be integrated into interactions like any other element of a visualization. Brushing a marked text element, the related parts of visualizations become highlighted and vice versa. It is important to provide text decorations (e.g., boldface or underlining) to discern interactive text fragments from the rest. Opportunities, where such interactions make sense, appear naturally whenever describing a data-related entity, as already described in empirical findings of data-driven stories (Section 3.4 and Section 3.5).

Make captions dynamic – It is surprising to see that many visual analytics systems even lack basic captions for the visualizations shown in different views. The inclusion of captions helps to make the interface more self-explanatory for non-expert users. However, in an interactive system, it turns out that these captions should be adaptive. For example, when something changes or gets highlighted in the visualization based on a user interaction, the caption needs to change accordingly and should always accurately describe what is currently shown (Figure 5.6). Template-based natural language generation provides the means for implementing such adaptive and dynamic captions.

Pointers everywhere – The system discerns three types of explanations: data-driven, educational, and methodological (Section 5.1) according to their role in the data document. However, they should be smoothly blended in the interface. With educational or methodological hints, the data-driven text provides necessary pointers to understand it more easily. Similarly, an educational explanation can profit from examples from the data to better grasp the concept (as we have integrated in the respective

background text; see Figure 5.2 (D)). These hints may also be used as hyperlinks to more detailed explanations. When authoring the texts, we recommend strictly differentiating between the categories of text and also reflecting this categorization in the user interface by a consistent layout. For example, we presented *methodological explanations* only in tooltips—made available by hovering over info icons—and educational explanations were marked with the term *background* in the details view on the right. This consistency facilitates users to quickly learn where to look for a certain type of information.

Learn on the side – Similar to most visual analytics systems, we develop our approach to support users in better understanding and analyzing the specific data. With every instance of usage and performing data analysis, the users gain experience and get insights with respect to the overall analysis procedure or domain. However, in contrast to other interfaces, we enable support for *learning on the side* through methodological and educational explanations that educate users on the domain terminology and concepts. Through activating these explanations, the interface is adapting—not automatically but with only a little extra effort—to the individual information needs of the users; for instance, expert users can simply ignore them as they are unobtrusive.

5.3 Explorantion and Chatbots

An increasing number of visualization systems are considering a natural language interface for constructing, modifying, refining, and interacting with a data visualization.^{191,190,172,152} The objective, here, is to relieve users from worrying about details of visual design and encoding while directly asking questions about the data. Using such an interface, users can simply specify their actions in a simple chat interface; for instance, just by typing instructions like “*show the relationship between life expectancy and health expenditure*” (e.g., the system can respond with a scatterplot), and then “*show the differences among different continents*” (the system can encode continents as colors in the scatterplot) in the context of a dataset. While the research on natural language interfaces is still in the early stages and filled with challenges¹⁷², it is a promising direction to make the (visual) analysis more accessible to a broader audience. Considering a broad audience, natural language interfaces can be the most natural way of exploring information. However, existing systems either provide answers to user queries as a visualization^{191,152} or as text (e.g., search engines like Google). While the former answer may suffer from a lack of intuitiveness, explicitness, and context, the latter has no exploration aspect (other than reading many relevant articles on the subject matter). We believe that providing an answer to users’ queries as an explorantive interactive data document facilitates them in not only knowing what they are looking for, but also in exploring the context and other relevant information.

This section explains the idea of combining a chatbot interface with explorantion in a very specific use case scenario, i.e., searching the required information in a knowledge graph dataset. As a response to the query, the user is presented with an explorantive representation of data (as already demonstrated in previous sections). The section begins with a brief motivation and introduction of the scope and dataset. Then, it goes on to describe the approach, followed by four application examples showcasing the usability. Finally, it concludes with a discussion of the strengths and limitations of the approach.

5.3.1 Motivation and The Problem

Nowadays, many people employ search engines to ask questions and know more about public figures*. The process is fairly straightforward: users ask questions and search engines try to reply precisely by pulling the related excerpts from digital encyclopedias (e.g., Wikipedia). Often, search engines (e.g., Google) employ knowledge graphs to show compact biographic information about historical figures as info boxes aside the search results. However, these simple excerpts from a knowledge graph are typically limited to a single person and do not convey interactions between different public figures. Traditionally, information about connections between historic figures is reported in history books and encyclopedias but is limited to the connections drawn by the authors of those bodies of text. In a historical context, learning about famous people of the past and how they interacted with each other helps in understanding historic developments in world politics, sports, or science. Knowledge graphs store information about these persons in a structured way—for instance, their dates of birth, family ties, achievements, and participation in certain events—and can be leveraged to infer possible connections between arbitrary figures of the same era.

Employing a (natural language) chat interface—like a search engine but very focused on the above-mentioned use case—our system, *VisKconnect* aims at providing a familiar, natural, and accessible starting point to start exploring the knowledge graph data. Figure 5.7 shows the response of an example query asking whether three soccer players (Mats Hummels, Miroslav Klose, and Philipp Lahm) have met. While the chat explicitly answers the question (E), three-colored rectangles in the timeline visualization (A) reveal the shared events. The potential connections between historical figures and their potentially intertwined lives are visualized using different visualizations. Such points of contact could include events such as sports tournaments, award ceremonies, or summit meetings. Through three visualizations—namely, an event timeline, an event map, and a relationship graph—*VisKconnect* provides access to the underlying connections for users to explore.

5.3.2 VisKconnect

Given a natural-language question about two or more persons of public interest, *VisKconnect* returns a textual answer to the question plus visualizations that reveal potential connections between the persons mentioned in the question. As shown in Figure 5.7, these visualizations focus on different facets: A the event timeline represents the persons’ lives over time, B the map view depicts geographic co-locations, and C the relationship graph provides a concise overview of shared events. Additionally, E the chat panel serves as a textual question–answer interface. *VisKconnect* is a web-based system; the front end is developed in TypeScript using Angular while the backend is written in Python.[†]

* “A public figure is a person, such as a politician, celebrity, social media personality, or business leader, who has a certain social position within a certain scope and a significant influence and so is often widely of concern to the public, [...] and is closely related to public.” – Wikipedia

[†]The implementation of the *VisKconnect* system was greatly supported by a group of five graduate students who are also coauthors of the corresponding paper.⁹² The two senior doctoral candidates: Shivam Agarwal (research area: visualization) and Simon Gottschalk (research area: knowledge graphs)—also co-authors—participated in feedback rounds about the visual design of the *VisKconnect* system during the iterative development phase.

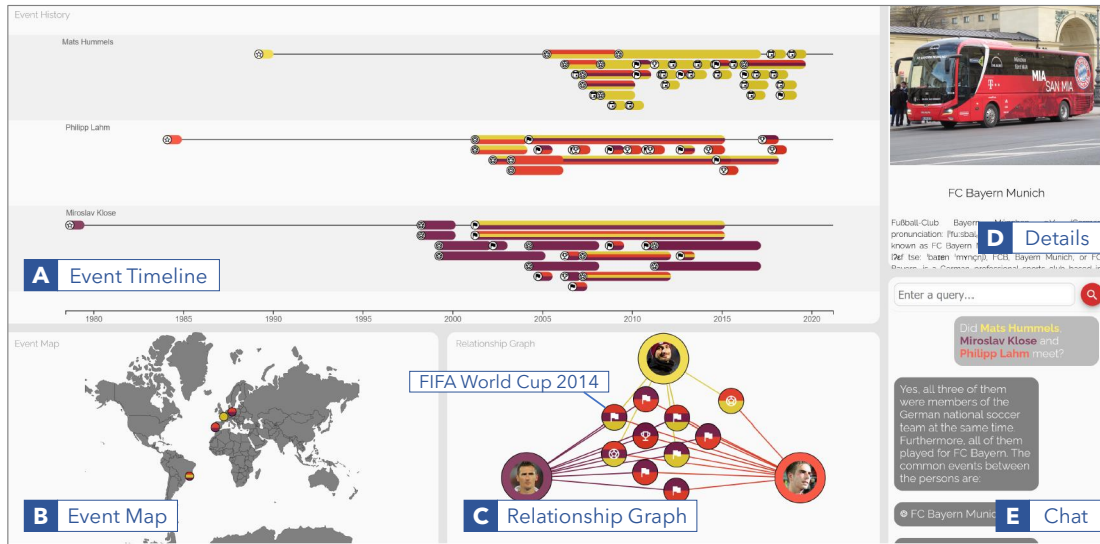


Figure 5.7: VisKconnect interface visualizing the result of a query on soccer players: (A) An event timeline shows individual (un-colored rectangles) and shared (multi-colored rectangles) life events of all historical figures in a user query. (B) An event map provides a geographic perspective of those events. (C) A relationship graph provides an overview of all shared events. (D) Clicking an event brings up a related article from Wikipedia. (E) A chat interface enables typing in a query and generates a short textual answer.

EVENTKG AND DATA RETRIEVAL

The approach requires structured information about real-world entities, specifically persons, their relationships, and the events they have been involved in. A natural source of this information is knowledge graphs. A knowledge graph $G = (N, R)$ consists of a set of nodes (N) representing real-world entities (e.g., *Mats Hummels*, the *FIFA World Cup 2014*) and a set of edges (R) denoting relationships between entities.⁶⁵ The relationships are represented as triples consisting of a subject, a predicate, and an object (e.g., $\langle \text{Mats Hummels, participated in, FIFA World Cup 2014} \rangle$ and $\langle \text{Mats Hummels, type, Person} \rangle$).

EventKG⁵⁴ is a specialized knowledge graph that incorporates event-centric and temporal information extracted from several other knowledge graphs and semi-structured sources such as Wikipedia and DBpedia. Therefore, VisKconnect uses EventKG as the main source of data. EventKG consists of several types of events, temporal relationships, and an event class ontology that provides a basis for the visualization components of the system. Figure 5.8 shows an excerpt of the EventKG with triples about the soccer players Mats Hummels and Miroslav Klose that indicate a shared event; despite the absence of a direct relation between the two footballers, it is still possible to infer a connection as they both participated in the FIFA World Cup 2014.

As stated earlier, VisKconnect has a specific focus on connections between historical figures. Consequently, it retrieves (i) the relation of people to associated events (e.g., Mats Hummels’ participation in the FIFA World Cup 2014) and (ii) the temporal associations between different persons (e.g., Marie Curie was married to Pierre Curie from 1895 to 1906). We define a connection between two people if

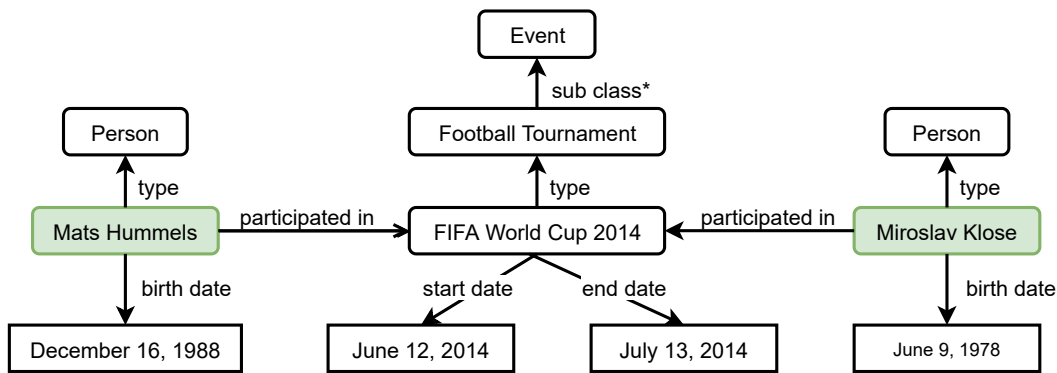


Figure 5.8: An excerpt of EventKG about the relationship between Mats Hummels and Miroslav Klose; *sub class** denotes transitivity.

there exists an undirected path between them via an event. We use parts-of-speech tagging and named entity recognition using pre-trained models of *spaCy* to identify person names. These names are then matched against EventKG through ElasticSearch², and a SPARQL^{*} query is formulated to retrieve information about people of interest along with their individual and shared events from the EventKG.

The *intent* of the question relates to the kind of relationship between persons in the user-asked query and is required to generate a textual response. VisKconnect restricts to three kinds of relationships between persons and events: *professional*, *personal*, and *general* relations. Professional relations describe working alliances between people like receiving an award or war alliances. Personal relations consist of friendship between persons or family ties identified by events like marriage or childbirth. Any other relation falls under the general category (e.g., if a person has influenced the other person). The system uses a rule-based approach to detect the intent. These rules are identified through the analysis of a set of 500 questions that are relevant for EventKG¹⁵⁸; each relationship type is associated with a set of pre-defined words. For instance, words like “collaborate”, “work”, and “ally” define professional connections. To classify the user’s query as one of the relationship type, we use semantic similarity between characteristic words of a relationship type and important words in the user query (excluding nouns and stop words). The system uses a pre-trained *word2vec* model¹²⁰ for computing semantic similarity. It is important to note that the intent is only relevant for generating textual answers and has no impact on visualizations.

THE INTERFACE

The interface consists of five components containing four visual and one textual view (Figure 5.7). The views are connected through brushing-and-linking and consistent color coding. The inspiration for visualizations—in particular, the event timeline—have been taken from Joseph Priestley’s “Chart of Biography”¹³³ and other related work on biography visualizations.^{79,103,128}

^{*}SPARQL is a semantic query language for databases. It is able to retrieve and manipulate data stored in the Resource Description Framework (RDF) format that EventKG employs

- A Event Timeline** – The event timeline provides an overview of the overlapping events for all persons in a user query. We use a distinct color for each person and vertically separate the persons into individual timelines. Each timeline begins with the person’s date of birth and ends at either the date of death or the last relevant event (in case the person is still alive or received awards posthumously). Events are represented by thick colored lines, with their duration encoded in the length. We use exoteric icons and place them at the start of the line to communicate the type of events, for instance, a star icon for birth, a cross icon for death, and a flag icon for sports tournaments. Many events can take place at the same time in a person’s lifetime, therefore the overlapping events for the same person are placed on separate stacked rows. The shared events are visualized as multicolored lines, with colors corresponding to the persons involved.
- B Event Map** – The event map gives a geographical perspective of places where the events took place. For instance, Figure 5.7 **B** shows that all three soccer players participated in an event in Brazil (FIFA World Cup 2014). The events are displayed as colored circles, which are horizontally divided into different colors if the events are shared among two or more people. Since not all events have a geographical location in EventKG, fewer events might appear on the map compared to other views.
- C Relationship Graph** – It shows the connectivity of queried persons through shared events as a force-directed graph. The persons are displayed with their picture and corresponding background color as assigned in the event timeline (Figure 5.7 **A**). The events are connected to corresponding persons via colored edges. Nodes representing shared events use multiple colors similar to the event map; events only assigned to a single person are not shown to reduce clutter. A desired consequence of the force-directed layout is that the more connected event nodes appear toward the center, while the person nodes float in the periphery.
- D Details** – The details panel is reserved for showing the relevant Wikipedia article about an entity or event on demand.
- E Chat** – The chat serves as the entry point to the system. It greets users with a welcome message and a few sample queries. Users can freely type in a query. Figure 5.7 **E** shows an example query and the generated response. Answering the user’s question is done either using the GPT-3—an autoregressive language model that uses deep learning to produce human-like text²⁴—or through the identified shared events with pre-written text templates. GPT-3 receives the user’s question as input and tries to generate an answer based on its 499 billion tokens of pre-trained data. Since these tokens come from a variety of sources, such as Wikipedia or Common Crawl, GPT-3 answers are often more precise than the information displayed by the template. If GPT-3 fails to answer the question, VisKconnect generates a response from the text templates. This response is based on the number of identified events and the intent. When there is no identified event, VisKconnect reports that no event was found with temporal overlap between the queried persons. When shared events are detected, the answer reflects this, and the identified events are displayed below. Figure 5.9 shows two textual answers generated by GPT-3 and the text templates, respectively.

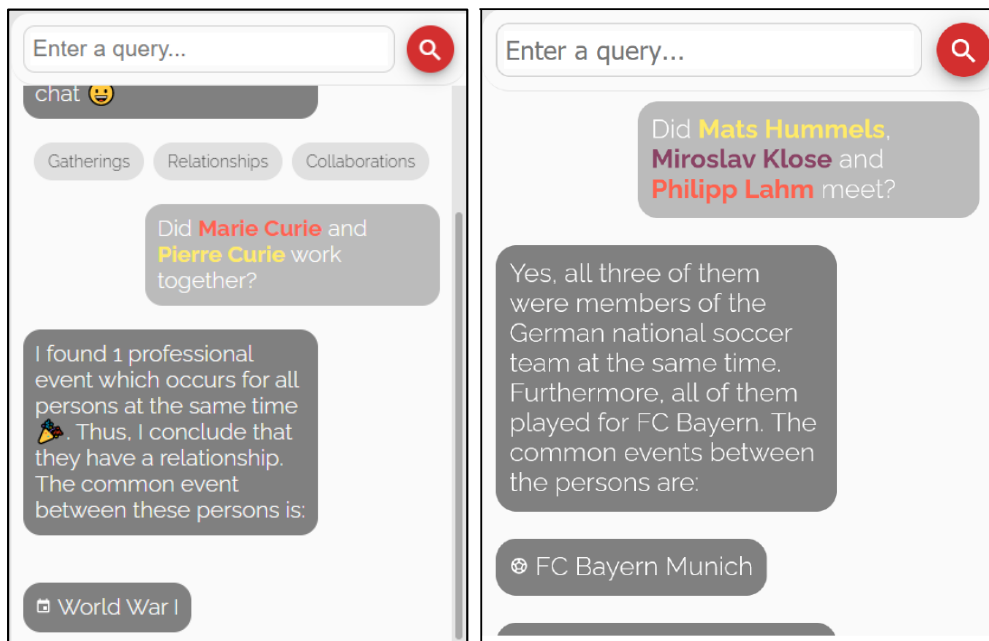


Figure 5.9: Chat responses based on templates (left) and GPT-3 (right).

VisKconnect uses consistent colors and icons across all views to support quickly moving from one view to another. Furthermore, interactions link the views. For instance, hovering over an event in any component highlights the related events in other views. Users can click any event anywhere to pull up the related Wikipedia article in a dedicated panel (Figure 5.7 D). Besides, zooming and panning in the event timeline, event map, and relationship graph helps in exploring the details.

5.3.3 Sample User Queries and Responses

To evaluate the expressiveness and usefulness of VisKconnect, we tested the system by asking various queries. Here, we first present a few examples and then analyze the system’s response to highlight benefits and potential problems.

1 – WHEN DID PIERRE CURIE AND MARIE CURIE MARRY?

The system (GPT-3) correctly answers the question as: “They married in July 1895” followed by listing two individual events “*spouse: Marie Curie*”, and “*spouse: Pierre Curie*”. Clicking the ring icon in their lifelines, it is found that the marriage lasted from 1895 to 1906; Pierre died in 1906 (Figure 5.10 I). Their shared events include “*Nobel Prize in Physics (1903)*”, “*Davy Medal (1903)*”, and “*World War I*”. Pierre is also connected to events after his death. For instance, he posthumously received the “*Elliot Cresson Medal (1909)*”.

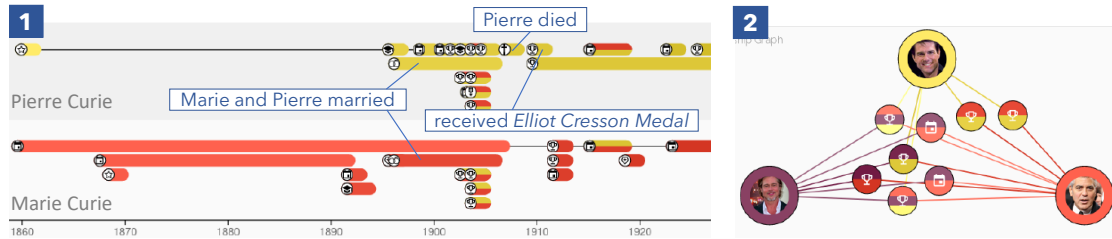


Figure 5.10: Cut-outs of two responses generated by VisKonnect. (1) Event timeline for Example 2. (2) Relationship graph for Example 3.



Figure 5.11: Excerpt of a question about Einstein and Schrödinger (Example 4). The template-based answer identifies the Solvay conference as a shared event where the two scientists were photographed together.

2 - DID BRAD PITT, GEORGE CLOONEY, AND TOM CRUISE ALL RECEIVE MOVIE AWARDS?

The system (GPT-3) answers that they have all received the Golden Globe Award and were all nominated for an Oscar award. Two events appear in the chat: The “*Golden Globe Award*” and “*Academy Awards*” (Oscar) supporting the claim generated by GPT-3. Searching for the award symbol in the other views (e.g., relationship graph – Figure 5.10 2) and looking up details further asserts the answer.

3 – DID ERWIN SCHRÖDINGER MEET ALBERT EINSTEIN?

Figure 5.11 shows a cut-out of the generated response. The template-based answer describes that four shared events were identified for the two scientists; they can also be seen in the relationship graph on the right. Among these, the “*Solvay Conference*” seems the most likely event for a true meeting—clicking it actually brings up a photograph where both men can be seen together.

VisKconnect (GPT-3) states that they all were members of the German national soccer team at the same time, and they all played for FC Bayern. The chat lists two shared events: “*FIFA World Cup 2014*” and “*FC Bayern Munich*” (Figure 5.7 [E](#)). This fact also becomes evident by looking at two three-colored events in the relationship graph [C](#). The event timeline [A](#) further reveals that the three soccer players participated in other sports events together between 2002 and 2020.

VisKconnect could answer most questions correctly or partially correctly as text; the answers generated by GPT-3 were more precise than the template-based answers. The accompanying explorable visualizations helped verify the answer, provided context, and assisted in navigating to the answer in case VisKconnect failed to provide a textual answer. On the downside, we observed that, sometimes, GPT-3 answers did not fully align with the visualized events. Sometimes, events are very general and overestimate who might have met or is related (e.g., *World War I* is assigned to many persons). As some events are prolonged or assignment to an event does not necessarily imply the physical presence of the person. Some events may seem misplaced in the event timeline and start either before birth (e.g., before Marie Curie’s birth in Figure 5.10 [I](#)) or after death (e.g., after Pierre Curie’s death in the same figure). VisKconnect also had problems recognizing the names of some historical people, especially the ones having titles, prefixes, or suffixes (e.g., Queen Elizabeth II).

5.3.4 Strengths, Limitations, and Challenges

The system is a proof of concept that a natural-language interface integrated with knowledge graphs and visualizations can support lay users in answering non-trivial questions about historical figures. Still, there are various open issues and interesting research challenges. In the following, we discuss these issues and challenges along with the strengths of the approach.

Verification and explanation of results – Knowledge graphs and language models are two distinct approaches and have recently been combined for answer generation.⁷⁷ VisKconnect demonstrates how this combination—through linked visualizations and textual answers—can provide immediate benefit to the user. With VisKconnect, on the one hand, a user can verify the answer generated by the language model through visualizations of related events. On the other hand, the generated answer helps in explaining the information conveyed through the visualization of knowledge graph data.

Exploration of intertwined lives – VisKconnect showcases the biographies of queried people. The linked visualizations enable exploration of their lives not only from temporal and spatial perspectives but also highlight the shared events to show how their lives were intertwined. The consistent color encoding across the entire interface allows easy navigation. This ability to explore relationships in the lives of historical figures can be applied in education, as demonstrated by the usage scenarios of the *HisVA* system.⁵⁶

Information selection – Knowledge graphs typically contain millions of nodes and edges (for instance, EventKG has more than two million relations connecting persons to events). Consequently, the selection of relevant pieces of information that are shown to the user poses an important challenge for any visualization built on top of knowledge graphs. For the selection of relevant events,

VisKconnect relies on the number of mentions of the respective events on Wikipedia. As a result, events that are of large historic significance such as the First or Second World War play an important role in VisKconnect, independent of the context, i.e., the question and the persons involved. For a more focused exploration of the persons' lives, there is a need for context-dependent relevance metrics and for providing interaction possibilities that allow users to deselect specific or less relevant pieces of information. Another problem relates to the ambiguity in entity names (e.g., whether the user is interested in Winston Churchill, the British prime minister, or the American novelist). Having human interventions in case of such ambiguities through interactive widgets (similar to Eviza¹⁵²) may resolve the problem to some extent.

Visual scalability – Taking inspiration from *TimeSets*¹²⁸, we used unique colors for each queried person and represented the shared events by filling the respective colors in the glyphs (e.g., lines in the timeline and nodes in the relationship graph). As a result, it enables the users to infer the shared events among queried historical figures. However, since it may become difficult to differentiate between more than five colors, the visual scalability of the interface is limited to about five people in a single query. Besides, the event timeline can quickly become visually cluttered when a person is linked to many events that temporally overlap.

Templates or GPT-3? – While the use of GPT-3 in the chat allows precise answers (see Figure 5.9), it is hard to control, predict what it would generate, and therefore integrate with the other visual views, which use EventKG as underlying data. Potential differences in textual and visual answers might confuse the users. In contrast, answers generated by text templates can always be made consistent to the visual answer. However, it requires bigger effort to make them sound natural and handle all possible cases, given the user has all the freedom to ask questions. While VisKconnect leaves it up to the user to verify whether GPT-3 produced the right answer, a future step is to automatically analyze GPT-3 response to flag potential misinformation or conflicting information.

Chatbot: expectations and benefits – Chatbot systems often cause the user to overestimate what the system can do¹⁷², as they can be usually perceived as capable as a human chat partner. Overestimating the system's capabilities leads to user inputs that the system cannot handle, therefore resulting in frustration for the user. However, we think that they can be a way of introducing a visualization system to general users who often lack visualization literacy. For instance, first answering a question and then pointing the user to related events (with a simple click, coherent colors and icons) which can take the user to visual views with the context still in mind. At present, VisKconnect does not support follow-up questions. A natural extension would involve making the system context aware and using the chat interface to apply filters. For instance, a user query "Did x meet y?" could be followed-up by "Were they married?" without repeating the names and using co-reference analysis. Similarly, follow-up queries like "Show me events related to scientific awards" could apply filtering to visualization.

5.4 Explorations in Virtual Reality Visualizations

With the advancement in environments like CAVE*, virtual, and augmented reality, researchers have started to investigate more immersive data visualizations.⁷³ As originally defined by Ynnerman et al.¹⁸⁹, the concept of exploration is also applicable to immersive environments. The active involvement of the users¹⁷⁷ as discussed previously for 2D visualizations—especially in a virtual reality environment—aligns with the broader definition of *immersion* presented by Isenberg et al.⁷³ as “*the engagement or involvement someone feels as the result of looking, exploring, or analyzing a visual data representation.*”

On the one hand, ideas of 2D exploration—especially with respect to what content needs to be communicated—are extensible to virtual reality applications. The presentation of this content in virtual reality, on the other hand, requires an altogether different approach. For instance, textual explanations—which are mostly used in 2D data documents—are no longer a good option. Longer textual explanations are not only difficult to read in virtual reality but may also occlude the display and produce visual clutter. Besides, if not properly designed, they cause motion sickness, fatigue, and discomfort.¹⁵⁵ Hence, audio narration can be an alternative to communicate the story instead of text. Audio has always been a powerful medium as an alternative to text when it comes to explaining. It has already been used in various virtual reality applications such as games, movies, reconstruction of historical events, and virtual museums. Creation of such applications involves prerecording and inclusion of audio commentary either at various stages of the story (e.g., in a game) or triggered by user interactions at predefined locations (e.g., in a virtual museum). Since such applications provide limited flexibility in terms of what users can interact with and change, the adaptability of aural content with user interactions is not a concern as everything can be scripted beforehand. In contrast, this adaptability becomes a challenge in interactive data documents where explanations need to be adapted for different datasets and user input, thereby demanding a more flexible approach.

This section aims at supporting exploration in immersive virtual reality visualizations by extending the idea of interactive data documents in virtual reality environments. The proposed multi-sensory representation—mapping to sensory channels such as vision and hearing among others—of data contributes to increased immersion⁷³. With our concept, Talking Realities, we target an adequate balance between an active exploration of data visualization and an explanation of findings through an audio narrative for providing an engaging and immersive experience.

5.4.1 Talking Realities

Talking Realities combines a data-driven audio narrative with an immersive virtual reality visualization. The audio narrative is automatically generated and adapts to data selections and user interactions. While the narrative guides users through identified insights, the interactive visualization allows free exploration. The concept is first discussed independent of any specific application, and then it is instantiated on two realistic application scenarios. The concept is applicable to virtual reality sce-

*“A Cave Automatic Virtual Environment (CAVE) is an immersive virtual reality environment where projectors are directed to between three and six of the walls of a room-sized cube” – Wikipedia

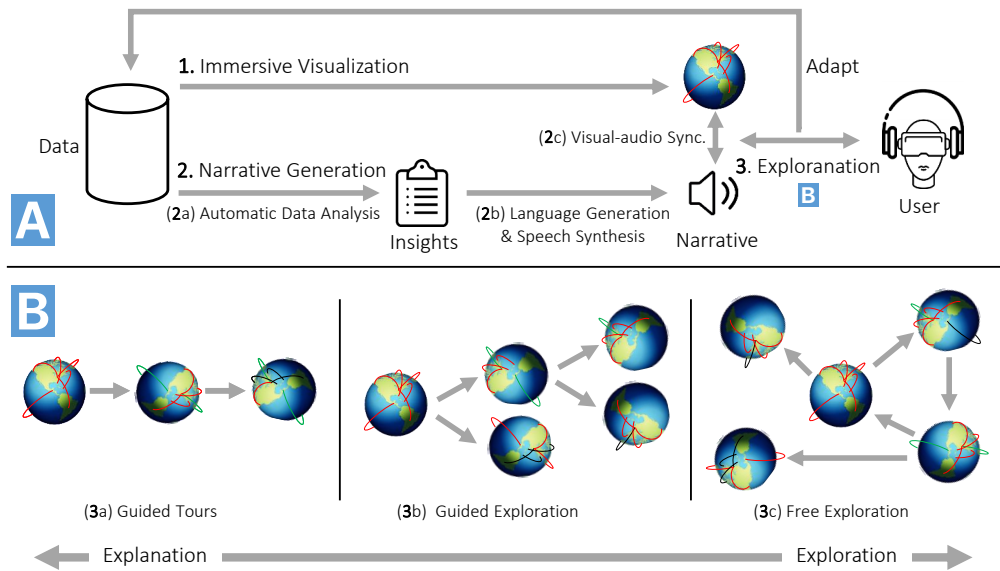


Figure 5.12: Abstract representation of Talking Realities—a concept for producing data-driven narratives in virtual reality. (A) The interplay of three aspects (I) *Immersive Visualization*, (II) *Narrative Generation*, and (III) *Explorantion*. (B) Explorantion offers varying levels of guidance: *Guided Tours* walk the users through a predefined sequence of events, *Guided Exploration* provides hints at various possible perspectives to explore, and *Free Exploration* enables the users to freely examine the dataset.

narios where data is represented as a single 3D visualization. We restrict the discussion to single-user applications and assume the visualization is viewed with a state-of-the-art head-mounted display. In general, we do not target an audience of experts but rather a broad group of users and do not assume specific previous knowledge about visual analysis or the data domain. For example, one such scenario is the visualization of long-distance air traffic projected onto a virtual 3D globe as it could be shown in an aircraft exhibition.

Figure 5.12 describes Talking Realities on an abstract level. The process begins with the visualization and analysis of a given dataset. While an immersive visualization provides an overview of the data and offers exploration, automatic data analysis results in insights that are then converted to audio narrative using natural language generation and speech synthesis (Figure 5.12 A – Narrative Generation). The representation is produced by synchronously integrating the audio narrative with the immersive visualization (Figure 5.13). Finally, users can choose from three different levels of explanation and exploration, ranging from fully guided tours to a free exploration of the visualization (Figure 5.12 B). We rely on existing techniques and off-the-shelf tools for accomplishing various tasks in our generation pipeline (Figure 5.12 A) as described below:

I – IMMERSIVE VISUALIZATION

A visualization in virtual reality provides an overview of the data and serves as an anchor point throughout exploration. To use the full potential of the three-dimensional virtual world, it should have a

meaningful third dimension that adds value to the analysis. Such visualizations can be borrowed from existing literature on virtual reality immersive analytics for information visualizations as well as scientific visualizations.²⁷ To support the exploration, the *immersive visualization* enables interactions. We discern between two types of interactions. The first type includes zooming, rotating, and panning. Since these interactions allow users to adjust the orientation of the scene, we refer to them as *visual navigation interactions*. They are not associated with any audio explanation and do not interrupt the audio playback either. This way, users may change the orientation of the visualization during playback to get a peek from a different angle. The second type of interaction allows users to select and manipulate data items that are encoded by visual elements in the visualization. These interactions directly relate to the underlying representation of the data and are referred to as *data interactions*. Users should be able to select data points and get details on demand, which can be presented as audio comments. It should be possible to filter or sub-select the visualized data. Generally, interactions in virtual reality are triggered through hand-held controllers.

2 – NARRATIVE GENERATION

Orthogonal to immersive visualization, narrative generation aims at automatically identifying interesting insights from the data by applying various analysis techniques. The resulting insights are then verbalized. The process consists of the following three steps:

2a – Automatic Data Analysis: The first step is to automatically find interesting analysis insights within the data. In line with the empirical findings (cf. Section 3.3), Table 5.2 provides an incomplete list of different types of analyses that can be performed. General *statistics* on the overall dataset or the most prominent parts of the data can give the users a first overview. *Clusters* of similar data items might be interesting to study as they show the main structure of the data. Specific *examples* of data items can be identified to either illustrate a representative case for a cluster or, in contrast, show noteworthy exceptions or outliers. If the users are interested in specific data items (either by selection or assumed background), a data-driven *comparison* of the items to a set of other items is relevant. Also, *trends* that describe consistent changes across sequential information, such as in a time series are often of particular interest. Each type of analysis may produce a varying number of results, in total, often beyond a number that can be realistically presented to the user. In this case, we need to prioritize the findings, for instance, just listing the most prominent clusters instead of all (Equation 4.1).

2b – Language Generation and Speech Synthesis: The detected insights are then transformed to natural language text. To accomplish this, we use an automatic generation process as already described in Section 4.1.2. Next, to convert the generated text to an audio output, any off-the-shelf speech synthesis API can be used as offered, for instance, by Google or Microsoft. These APIs allow customizing the way a text is read through the Speech Synthesis Markup Language (SSML), an XML-based markup language that controls the pronunciation and prosody of the synthesized speech. In our narrative, we discern between *contextual* and *data-driven* explanations. The former are used to introduce the dataset and provide other background information about the scenario. The data-driven explanations, on the other hand, are the ones that report notable data insights (I.i – Section 5.1).

2c – Visual–Audio Synchronization: Since the content is presented across two different media

Table 5.2: Various types of analysis that can be performed on a dataset.

Analysis Type	Description
Statistics	Statistical properties that summarize (parts of) the data (e.g., average, data ranges, correlations)
Clusters	Data items of similar properties or dense connections
Examples	Single data items that are representative for a group or noteworthy outliers
Comparison	Contrasting data items to a set of other items
Trends	Changes in sequential information

(audio and visual), it becomes important to temporally align these representations. Similar to *vis-text linking* (III.i – Section 5.1) Figure 5.13 illustrates the synchronization of content during the narration. The respective parts of the visualization are either (visually) highlighted or animated in synchronization with the related *data-driven* audio narrative (green and orange blocks). *Contextual* narrative (gray blocks) is usually not directly associated with the visualization and hence cannot be synchronized.

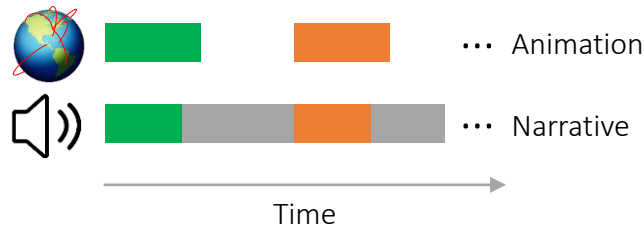


Figure 5.13: The visualization is animated in synchronization with the audio narrative.

3 – EXPLORATION

The data-driven audio explanations provide guidance to the users, while the visualization allows a *Free Exploration* of the data. Considering this as a continuum between explanation and exploration (Figure 5.12 B), we suggest the following three specific usage modes, all three including elements of explanation and exploration but with varying strengths.

3a – Guided Tours: Focusing on *explanation*, *guided tours* correspond to a fully automated story including a predefined sequence of events. They are like self-running presentations.⁸⁸ The main objective is to present the users with a series of insights in adequate detail. A linear sequence of available insights gets automatically selected from the data. For instance, in the case of air traffic data, *Guided Tours* can walk the users through the busiest airports of the world; the airports and sequence might be different depending on the data currently loaded. These tours serve as the starting point and can be used to get familiar with the data and visualization, similar to a tutorial. Users are free to select from a

list of different, but independent tours according to their interests, which provides a minimal degree of *exploration*.

3b – Guided Exploration: In contrast, the *guided exploration* scenario allows users to choose between various story branches at fixed junction points. The users begin by looking at the visualization and hearing an introductory audio narrative. Then, as shown in Figure 5.12 B, at the first junction point, the users are presented with different possible storylines and can interactively choose one. As a third type of interaction, we refer to these as *navigation interactions*, since they determine the subsequent progression of information. The system needs to clarify the available options, for instance, using self-explanatory icon images or explaining available options as part of the audio narration. This is repeated recursively for several levels. In that way, the users navigate through the data in a mix of short *explanations* and *exploration* through choosing the next option. Different starting points should be provided for entering different *guided explorations*.

3c – Free Exploration: *Free exploration* enables users to further investigate details after having viewed a *guided tour* or *guided exploration*, but it can also be considered an alternative to these modes. For instance, users may have some hypotheses in mind that they want to validate. They select data points and activate short *explanations*, which provide details on demand. Various such options for interactions should be provided so that users can steer in the desired direction. By selecting these various possibilities, users create their own stories fitting their current information interests.

Both *data interactions* and *navigation interactions* are associated with the audio narrative. Upon triggering them, the audio starts playing with the relevant parts of the visualization animated. Since the audio comments may take a while to complete, they might interfere with follow-up interactions. While *visual navigation interactions* (such as zoom, pan, rotate, etc.) should not affect the playing audio, *data* and *story navigation interactions* should influence the audio as they clearly indicate that the user is now interested in something else. When the user triggers a *data* or *navigation* interaction while an audio comment is playing, the active playback immediately jumps to the newly triggered comment. Among other *audio control interactions*, the user should further have the option to skip any audio playback at any time, which introduces a fourth type of interaction. Since *guided tours* provide limited exploration, they only cater few *story navigation interactions* for choosing or switching between the available tours. *Guided exploration* consists of both *data* and *navigation interactions*, where the former may also serve as an anchor point to a branching follow-up story (controlled by the latter kind of interactions). Finally, *free exploration* only provides data interactions. However, *visual navigation* and *audio controls* are available in all modes of exploration.

5.4.2 Application Examples

Next, we instantiate the concept of Talking Realities to two different datasets to showcase its applicability. First, the air traffic data is chosen as an example due to its spatiotemporal nature and suitability for a broad audience—nearly anyone, mostly irrespective of age and background, can understand and connect to this scenario. Second, we sketch a mock-up for demonstrating the generalizability of the concept. We pick the example of an information visualization scenario. The example relies on exist-

ing tools¹⁵⁷ for producing immersive visualizations; the audio explanations, however, are not implemented and are solely drafted for mock-up purposes. In contrast to the first example which focuses on *guided tours* and *guided exploration*, the second example goes deeper into the types of automatic data analysis (Table 5.2) and describes the *free exploration* mode. The second example may be employed in educational scenarios to explain, for instance, statistical concepts to pupils.

AIR TRAFFIC DATA

The choice of this scenario is grounded in the fact that the data has a 3-dimensional representation; visualizing it in a virtual reality environment comes naturally and makes sense. Also, due to its complex geo-temporal nature, it can benefit from explicit audio explanations while showing it to the general public. Our target audience includes flight enthusiasts, school or university students, or anyone who wants to explore and get insights from global flight data. The prototype implementation uses *Unity* and *C#*; it is developed for the *Oculus Rift* head-mounted display*. The dataset we use is freely available at *OpenSky Network*. It consists of more than eleven million flights in a single calendar year. Visualizing such a large amount of flight trajectories in virtual reality would result in slow rendering times and cause visual clutter. Also, short flights are not particularly relevant on a global scale. Therefore, we restrict ourselves to only large airports (visited by five million passengers annually) and inter-continental flights longer than at least ten thousand kilometers. This filtering leaves us with about a hundred thousand flights for the year 2018 and 600 airports.

Immersive Visualization: As we target an audience of non-experts, we went for straightforward visualization of 3D trajectories as opposed to more complex visualization techniques.⁷² The inter-continental flights are visualized on a 3D globe of the Earth as colored trajectories starting and ending at airports. Figure 5.14 A shows an overview of all the inter-continental air traffic on August 2, 2018. We visualize airports as blue spheres at their exact locations. We approximate flight trajectories using cubic Bézier curves and visualized them as smooth curved lines. Although it may be more interesting for advanced users to see the exact flight paths, it would probably confuse novices as the exact trajectories result in more visual clutter. To prevent overlapping trajectories, we randomly exaggerate flight altitudes. A color gradient (red–green) marks the departure and arrival point of a flight. To enable exploration, the visualization offers *data* and *visual navigation interactions*. Users can interact using the touch controllers of the *Oculus Rift*. It is possible to pan, rotate, zoom in, and zoom out (*visual navigation interactions*). For instance, users can grab the visualization and then expand or contract to zoom in or out the entire visualization. Users can interact with countries and airports to get details on demand. The calendar enables users to select and load data for different days (*data interactions*).

Narrative Generation: The analysis (*Automatic Data Analysis*) results in interesting insights for all airports and countries. For each airport, we find its international as well as national (statistical) rank, number of daily departing and landing flights, longest flights to and from it, most connected airport (in terms of number of flights), and the busiest hour of the day (Table 5.2 – *Statistics*). Similarly, all these details—except for the last one—are detected and aggregated for each country. The analysis re-

*The implementation of this prototype application was greatly supported by a group of graduate students at the University of Duisburg-Essen.

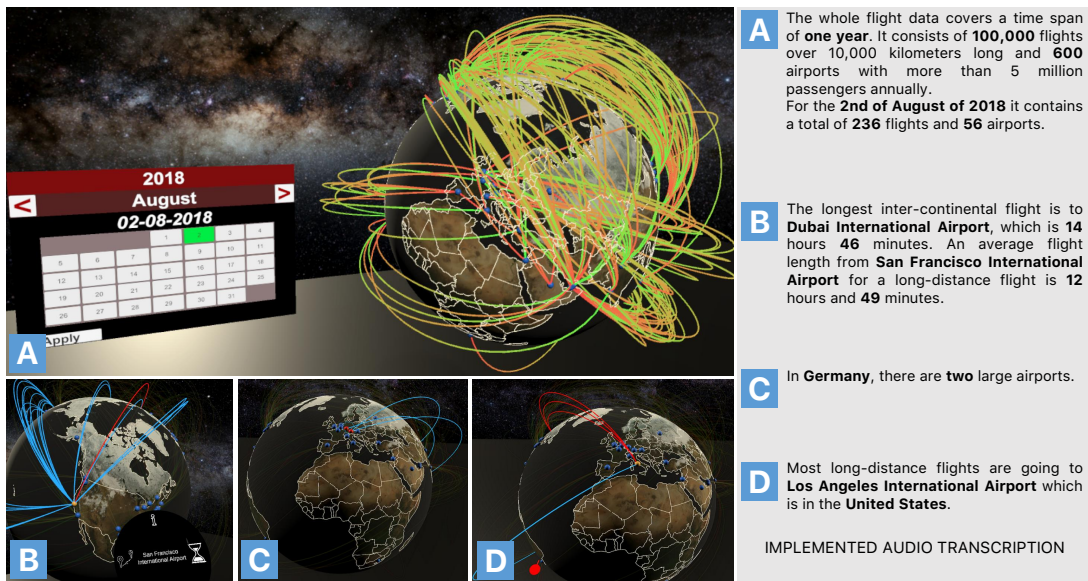


Figure 5.14: Graphical interface of the prototype and various animations. (A) The visualization displays the aggregated inter-continental flights for one regular day. The color gradient (red:departure to green:arrival) denotes the direction of flights. Animations showing (B) the longest flight from an airport, (C) large airports of a country, and (D) most flights to any other airport. The right (gray) box discloses the audio transcription that are played when users see corresponding animations.

runs and updates according to data selections (for instance, when the users select another day of the year). To convert identified insights into audio guides, we first employ text generation (*Language Generation*) as already described in Chapter 4. Afterward, we use a text-to-speech API for transforming text into audio (*Speech Synthesis*). A combination of animations and visual highlighting helps synchronize the visual content with the audio narrative. When the audio narrative talks about a specific insight, the relevant parts of the visualization are highlighted (related airports with red and trajectories with blue color). The rest of the flights and airports are faded out using the opacity channel to create a focus–context effect. A sequence of this highlighting produces an animation effect. Figure 5.14 (B – D) provides examples of audio guides and the visual–audio synchronization. For instance, when the user selects San Francisco International Airport (Figure 5.14 B), the audio plays “*The busiest hour of this day is 1 o’clock, where three inter-continental flights arrive or depart.*” while all these flights get highlighted (not shown in the figure); the narrative then continues to describe the longest flight from this airport as “*The longest flight is to Dubai International Airport, which is 14 hours and 46 minutes.*” and the corresponding flight gets highlighted in red.

Exploration: In the system, users are confronted with the visualization and an introductory (contextual) audio that explains the application scenario and data statistics (Figure 5.14 A). At this point, users can either go for *guided tours*—that are available on a virtual menu—or start exploring the visualization (*guided* or *free exploration*) on their own. Guidance is provided to the users via virtual radial menus (Figure 5.15). These menus hint at possible aspects of data analysis that are available (*guided exploration*). They can be accessed through hand-held controllers. For every country and

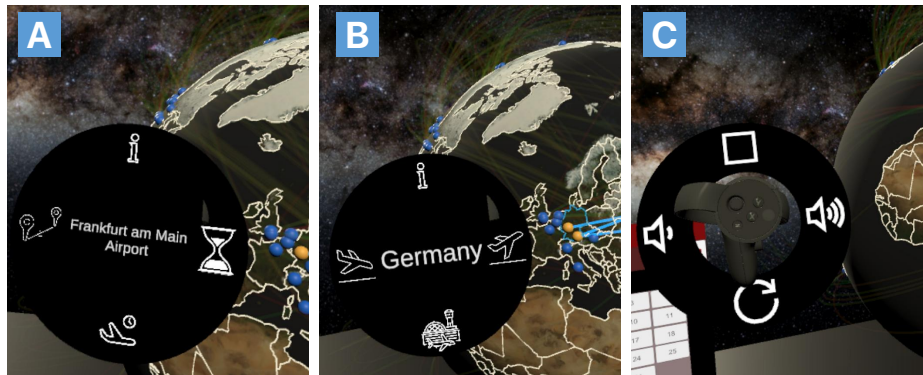


Figure 5.15: Virtual menus providing guidance on the possible aspects of exploration on selecting an airport (A) and a country (B). Audio controls (C) can be accessed at any time to replay, skip the current playback, and to adjust the volume.

airport, detected insights are grouped into four distinct categories. For instance, when users select a country, a menu (Figure 5.15 B) appears showing the name of the country and four possible options (*navigation interactions*). Users can then choose among the *general statistics*, *arriving and departing flights*, or *large airports*. Similarly, for an airport, the radial menu (Figure 5.15 A) contains information on *general statistics*, *temporal information* (e.g., busiest hour), and the *longest flight*. *Audio controls interactions* are provided on a similar radial menu that is available on the left touch controller (Figure 5.15 C). It can be accessed at all times and includes skip, repeat, and volume control options.

The explanations, in the form of audio guides, are presented when users interact with the globe visualization. In *free exploration* mode, interacting with countries or airports brings forward the details-on-demand audio guides. For instance, Figure 5.14 C shows the result of selecting *Germany* on the globe. It highlights that there are two large airports in the country with not a lot of inter-continental traffic. Similarly, Figure 5.14 D shows the state of visualization and audio comment while interacting with *Leonardo da Vinci (Fiumicino) Airport* airport of Rome, Italy. The comment says that this airport is most connected to *Los Angeles International Airport* via long-distance flights; three flights fly daily between the two airports.

MULTIVARIATE DATA

Tabular data where objects are described along multiple variables is a common type of data. For instance, the multivariate dataset *mtcars** contains eleven properties of 406 different car models that were manufactured between the years 1970–1982. It can provide insights into similar car models and relationships between different properties of cars.

Immersive Visualization: A scatterplot is an intuitive and established visualization that can be used for this purpose. As it is not limited to only two dimensions, more properties of cars can be simultaneously visualized as the third dimension (z -axis), color, size, shape, or opacity. Figure 5.16

*<https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/mtcars.html>

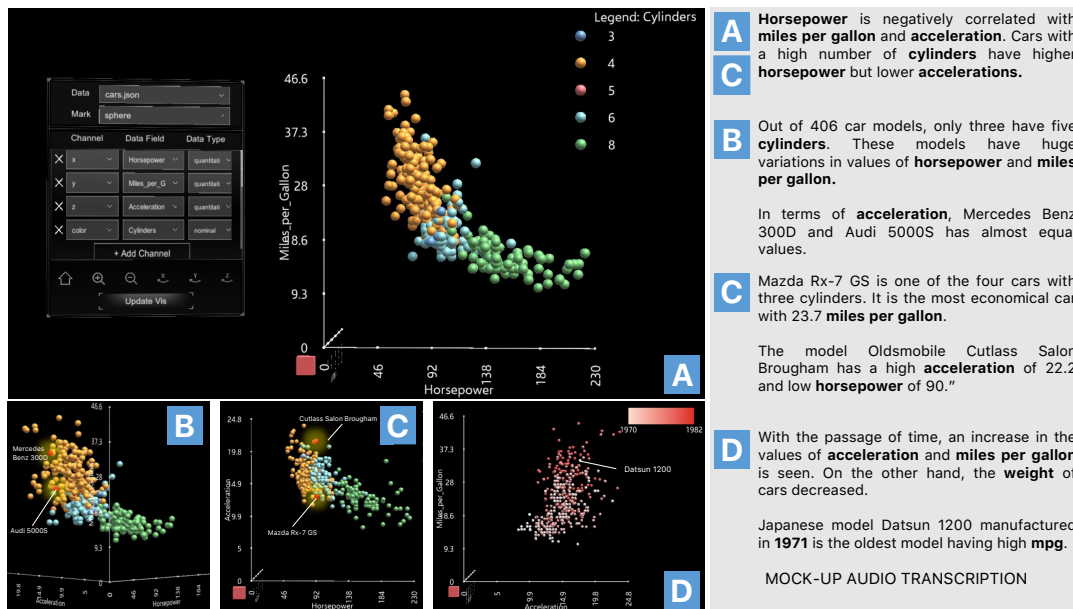


Figure 5.16: A mock-up illustrating the application of Talking Realities to *mtcars* dataset. (A) Scatterplot visualizes horsepower, miles per gallon, acceleration, and cylinders for 406 different car models (1970–1982). The right (gray) box shows the transcription of data-driven audio that summarizes insights related to (A,C) correlations among all visualized variables (B) unique models (outliers) with respect to number of cylinders. (C) details on demand for a selected car model, and (D) changes in the values of different properties over the years. The first text block refers to both sub-figures A and C.

presents a mock-up of a possible implementation for virtual reality. At the center lies an interactive 3D scatterplot showing horsepower (*x*-axis), miles per gallon (*y*-axis), acceleration (*z*-axis), and the number of cylinders (color) of each car model. The visualization is produced in *Unity* with DXR¹⁵⁷ and offers interactions. It is possible to change (add or remove) dimensions using the menu on the left of the scatterplot as seen in Figure 5.16 A.

Narrative Generation: *Automatic data analysis* followed by *language generation and speech synthesis* can describe insights like pairwise correlations between the selected variables (Figure 5.16 A and C), outliers with respect to one or more variables (B), and temporal evolution of car models with respect to the visualized variables (D). The analysis types *Statistics*, *Examples*, and *Trends* have been used here (cf. Table 5.2).

Exploration: To steer the users' attention, related data points on the scatterplot would be highlighted in synchronization with the audio narrative. For instance, in Figure 5.16A, first blue spheres and then green spheres will be highlighted sequentially (fading out the other) while the audio plays: "Cars with a high number of cylinders have a higher horsepower but lower acceleration." Guidance can be provided by offering various analysis types to users, for instance, *clustering* with various types of clustering algorithms. Other users may want to directly explore the data with respect to specific questions. Figure 5.16 outlines an example of a *free exploration* scenario. It begins with getting an overview of the correlations among *horsepower*, *miles per gallon*, and *cylinders* (Figure 5.16 A). Knowing the

relationship between these three properties, now, the user wants to explore more the *cylinder* property; she simply chooses it from a menu on the hand-held controller. It turns out that the 5-cylinder cars are very rare and two of those (*Mercedes Benz 300D* and *Audi 5000S*) have almost equal acceleration values (Figure 5.16 B); since one of these models is occluded by other cars, the user rotates the view to inspect it better (*visual navigation interaction*). Next, she inspects certain car models of her interest; Figure 5.16 C describes the results of interacting with two different car models (red labeled dots). Since *Mazda Rx-7 GS* is an outlier, it has more explanation compared to *Oldsmobile Cutlass Salon Brougham*. Finally, the user decides to remove the variables *cylinders* and *horsepower*, and instead adds *year* and *weight* as variables. The *year* is added as a color gradient to the scatterplot from light red (1970) to dark red (1980) (Figure 5.16 D). Now, the data analysis re-runs and adapts to the new state of the scatterplot. The audio describes the relationship among the three visualized variables (*year*, *acceleration*, and *weight*), followed by highlighting an outlier with respect to *year* and *miles per gallon*.

5.4.3 Limitations and Future Opportunities

In general, the concept Talking Realities has focused on generating interactive data representation for virtual reality environments by integrating audio explanations and visually presented data. It uses similar linking concepts as discussed for 2D representations (Section 5.1) and extends them to virtual reality visualizations. However, our scenario and solution are still limited with respect to various aspects of content presentation and, in particular, human-computer interaction that are altogether different from conventional mouse interactions on a 2D screen. These aspects can be explored as future work and include further relevant research challenges.

Evaluate interactions for explanation – We have made suggestions—as part of the abstract concept as well as specific ones in the application examples—how to design interactions for controlling the progression of information in a given data representation. However, it still remains an open research question whether these interactions already sufficiently support the suggested (three) exploration modes and how to best achieve a smooth information flow. An extensive user evaluation is necessary to address it. Moreover, in the *guided exploration* mode, instead of having fixed exploration paths, the system can provide flexibility, for instance, different exploration can be suggested depending on the previous selections of the user. The main constraints are that the user should be able to go back and forth between different modes of explanation at any viewing location and interactions should be effortless.

Advance the natural feel of interactions – In a virtual reality environment, users usually interact with motion controllers they hold in their hands and by moving their heads around. This input mode supports basic interactions for pointing at objects, directly manipulating these objects, panning the visual view. However, other modes of interaction could feel more natural and increase immersion. In particular, speech input (like Google Assistant or Apple’s Siri)—instead of selecting and tapping objects—can transform the audio-guide into a conversational interface. However, it comes with the caveat that users might get disappointed if the interface is not able to answer arbitrary questions about the presented data.¹⁷² Also, hand gestures could further extend the Talking Realities concept. For ex-

ample, a halt gesture could naturally stop the current audio. Likewise, we can think about extending the output modes, for instance, including tactile feedback synchronized to audio and visual presentation. Such feedback, however, needs to be balanced well as the user might consider it as intrusive if it is too strong and too frequent.

Adapt to users' experience level – Considering varying levels of experience and expertise of end users, and consequently varying the level of explanation and guidance while interacting with the interactive data representation would be useful. We can speculate that *guided tours* and *guided exploration* may particularly educate children and visualization novices in a playful manner while *free exploration* can be interesting to advanced users. Still, also novices might like to play in a free manner, or the experts would profit from the hints on some advanced findings (*guided exploration*). Interesting challenges and future directions include: defining appropriate levels of insights and language for each group of users, testing whether certain interaction modes are preferred by different groups of users, and how an automatic solution can adapt to different user experience levels.

5.5 Conclusion

This chapter has introduced a generic approach of supporting *exploration* in (multimodal) interactive data representations leveraging the benefits of text, audio, and visualizations. Traditional visual analytics systems have a strong focus on exploration, while conventional storytelling methods are centered around providing more explanations in a communication context. Unlike both extremes, we consider our contributions in aiming to blur the hard borderline between explanations (provided by textual or audio content) and visual data exploration. The interaction model introduced to connect visual and textual or audio elements goes beyond previous work and showcases how text generation and brushing-and-linking techniques can play together in a multi-view system. We believe that text or audio, well-integrated with explorable visualizations, can make data analysis more accessible and easier to understand for a wide audience. With such close integration, eventually, text—or any other explanatory medium—and visualization will become only two points in a continuum (with any point in between possible) instead of being treated as two separate modalities.

The resulting approach can be classified as a visual analytics solution that puts a lot more emphasis on presentation and dissemination alongside exploration. Unlike traditional visual analytics systems, it no longer leaves users on their own to derive insights but helps discover them, evaluate them, verify them, and even learn on the side.

A user interface is like a joke. If you have to explain it, it is not that good.

Martin LeBlanc

6

Authoring Interactive Data Documents

Fully automatic solutions to produce interactive data documents—as discussed in Chapter 4 and Chapter 5—are domain-dependent and require a large effort to generalize to other application domains and datasets. Therefore, a natural next step is to equip creators (e.g., digital journalists) of such interactive content with an easy-to-use authoring solution. Although existing authoring solutions provide basic support for data-driven storytelling (see Section 2.5.1), they have poor support for creating interactive data documents—particularly with regard to allowing for interactive linking of the text and visualizations—that would ultimately allow exploration. Therefore, practitioners have to opt for programming frameworks (e.g., D3¹⁹ or *Idyll*³¹) to create such data documents. These programmatic approaches require programming skills and going through complex application code. Content makers (e.g., journalists, visualization designers) may not have such strong programming skills. Therefore, we need an authoring solution that is simple to use and requires little to no programming. In the early phase, this thesis started off with a simple declarative syntax to produce visualization–text linking in web browser-based interactive documents.^{97,98} These solutions, however, are restricted to specific domains and datasets (tabular⁹⁷ and graph data⁹⁸) and are not easy to generalize. On top of that, they still require basic knowledge of web programming (e.g., HTML); hence, these are not discussed in the thesis. Instead, this section describes a more generic and powerful approach leveraging an intelligent graphic user interface powered by natural language processing, resulting in a Web-based tool called *Kori*.

Kori is a mixed-initiative interface system that enables easy authoring of interactive data documents with a specific focus on the synthesis of text and visualizations through interactive references. Figure 6.1 shows the interface and a series of steps for constructing an interactive data representation. *Kori* is grounded in the design space analysis of implicit visualization–text references (Section 3.5) and leverages natural language processing to automatically identify references while users are authoring. It

combines this automation with a minimal and simple-to-use graphical user interface for constructing further references.

6.1 Design Considerations

The main objective of Kori is to facilitate non-technical users in creating customizable and highly interactive data documents. To facilitate this user group, the most natural choice of interface is a graphical user interface. In addition to creating interactive visualization, the system should assist users by detecting and suggesting potential references while they are composing a narrative about these visualizations. Besides, the system should provide an intuitive and easy-to-use interface for the manual construction of interactive references beyond the automatically suggested ones. Kori is developed along the following design rationale:

- D1 Suggest possibilities** – The system should not explicitly create references, but rather suggest them to the user. One reason is that every automatic detection system would inevitably result in false positives as the matter is complex and the linking might be subjective to the user. Another reason is that users may get annoyed by the automatic creation of many references—not all wanted. Therefore, the system will only suggest them, and the user can either accept or reject them.
- D2 Let users create** – Users should not feel restricted to only what is suggested by the system—the automatic suggestion of potential references may not be enough. Users may want to create additional references or combine the suggested ones into a single higher-level reference. The construction of a reference involves the selection of visual marks (data items) on the chart and relating them to the text. To this end, our goal is to support a smooth visual interaction design for creating references. The interaction design should be intuitive, efficient, and generalizable to multiple chart types.
- D3 Assist but do not distract** – In general, the users should not get distracted by the additional features our tool provides. They should be able to focus on creating the content, while assistance for linking text and charts blends in smoothly. It might go unnoticed at first, but becomes a valuable tool when revising and polishing the document. The suggestions and options to create references might even inspire the authors to communicate a deeper analysis of the data as understandable communication is easier to achieve.

6.2 The User Interface and Usage Scenarios

Kori comprises an *editor* and a *viewer*. Accordingly, it discerns two roles of users: *authors*, who create the content within the editor, and *readers*, who consume the content in the viewer. While the tool assists authors in creating interactive references, the readers profit from an improved synthesis of text and charts leveraging those references. In the following, we describe two typical usage scenarios that

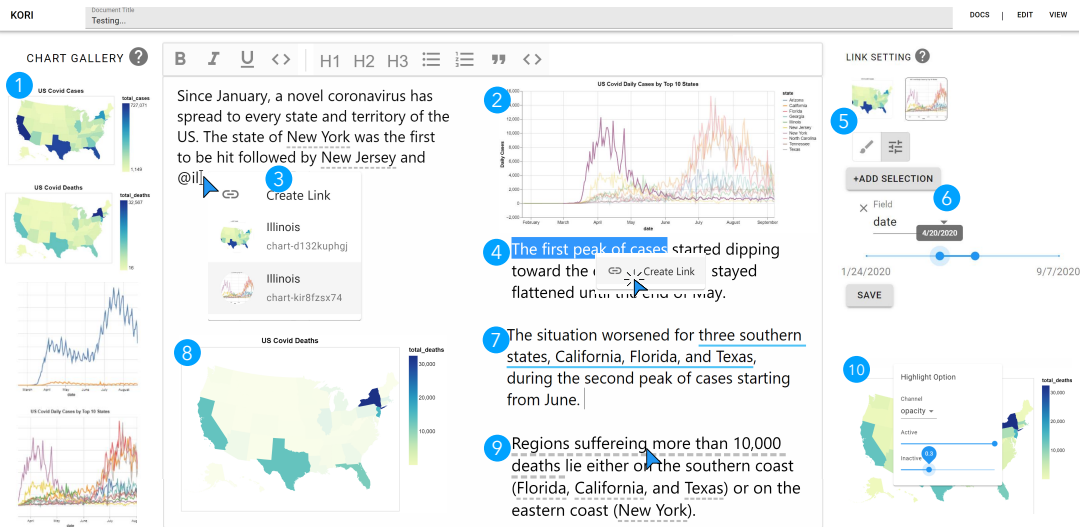


Figure 6.1: The Kori system consists of a chart gallery (left), edit area (middle), and a link setting panel (right). Kori automatically suggests potential references (dotted gray underline) as a user types. Besides, it supports manual creation of links through simple interactions (4–6). The steps 1–10 describe a usage scenario to create an interactive story.

illustrate both perspectives, before Section 6.3 describes the technical details. For the sake of demonstration, we take an example dataset and a collection of charts that describe the Covid-19 cases in the US (on federal and state level) from January to late August 2020. Let us assume that Alice writes an interactive data document for Bob.

6.2.1 Authoring

Figure 6.1 shows the *editor* interface of Kori. It consists of a chart gallery and an editing area. The chart gallery holds a collection of charts that can be inserted into the editor. Kori supports charts based on Vega-Lite syntax¹⁴⁸. Charts can be loaded into the gallery by dragging and dropping their specification files. Alice begins her composition by adding four charts (1), she has already created using Voyager¹⁸⁴—a Tableau-like interface for data analysis that can export Vega-Lite charts. Charts can be dragged from the gallery to the editor. The editor window provides a standard text formatting toolbar at the top.

Alice first wants to give an overview of the temporal development and adds the line chart (state level) to the editor (2) and then starts writing text. While typing, she gets automatic suggestions (e.g., New York). Special decorations—dotted gray underline—notify her about the suggestions. Curious, she hovers over the suggestion “New York” to preview it. As a result, the line representing New York gets highlighted (2). She types the @ symbol to trigger available suggestions while creating a reference for Illinois (3). The small chart avatars in the suggestion panel notify her about to what chart the suggestion corresponds to. Alice observes an interesting pattern about cases dipping toward the end of April and wants to create a reference for that. She does so by first selecting the text phrase and activat-

ing the reference construction mode ④. Then, she selects the line chart in the *link setting* panel ⑤, chooses the filtering mode ⑥, adjusts the value of the interval slider for date, and finalize her reference construction.

Moving forward, she wants to provide a comparison of three states that were hit hard by the pandemic during the second wave. She sees a suggestion for each state name, but she wants to highlight them at once to allow comparison. She simply does so by following similar steps (④ – ⑤) but this time choosing the direct manipulation ⑦ mode (see Figure 6.2 B), and simultaneously selecting three corresponding timelines using a multi-point selection on the chart ⑦. Since these references are manually constructed, they are underlined with blue color to discern them from suggestions.

To give an impression of the severity of the pandemic, she adds a map of total deaths to the editor ⑧. As she describes the states suffering more than 10,000 causalities, she gets a suggestion ⑨. She previews it to see whether it is correct and then accepts it. As she previews it, she realizes the opacity of faded out regions was quite low, and they were hardly visible. She adjusts the inactive opacity as desired ⑩.

6.2.2 Reading

Bob reads the composed article in the viewer. It is a restrictive version of the editor and offers a reading interface where Bob can activate the explicit referencing by interacting with the reference text. Kori uses visual highlighting to make the related parts of a chart stand tall. The default highlight scheme is the opacity channel.

6.3 The Kori System

The implementation of Kori includes the challenges of coming up with a mixed-initiative interaction paradigm that assists human authors (D1) and does so in a passive and non-distracting manner (D3) while still giving full control and freedom to the authors (D2). Kori is a web-based system; the front end is developed in JavaScript, React, Draftjs, and Vega-Lite, while natural language processing tasks have been performed in Python using SpaCy and FastText.¹¹⁹

6.3.1 Data, Charts, and Visualization—Text References

Kori supports a wide range of chart types including but not limited to (stack or group) bar charts, (multi) line charts, scatter plots, distribution plots, heat maps, and choropleth maps. However, advanced visualizations (e.g., network diagrams, tree maps, etc.) as well as charts with coordinated views are not supported. Kori expects Vega-Lite¹⁴⁸ specification—JSON syntax—of a chart. We rely on Vega-Lite for constructing and modifying charts, as it offers an expressive and declarative syntax. Authors can load their data into Voyager¹⁸⁴ or Vega-Lite editor⁴ to construct and export charts. These charts can then be imported to Kori. Although not implemented, the chart gallery can be connected to Voyager¹⁸⁴ to further facilitate data analysis and chart construction inside the tool. Since, at present, Kori is restricted to standard visualizations—that can be created using Vega-Lite or Voyager—we use the term ‘charts’ to refer to them, instead of the broader term ‘visualization’.

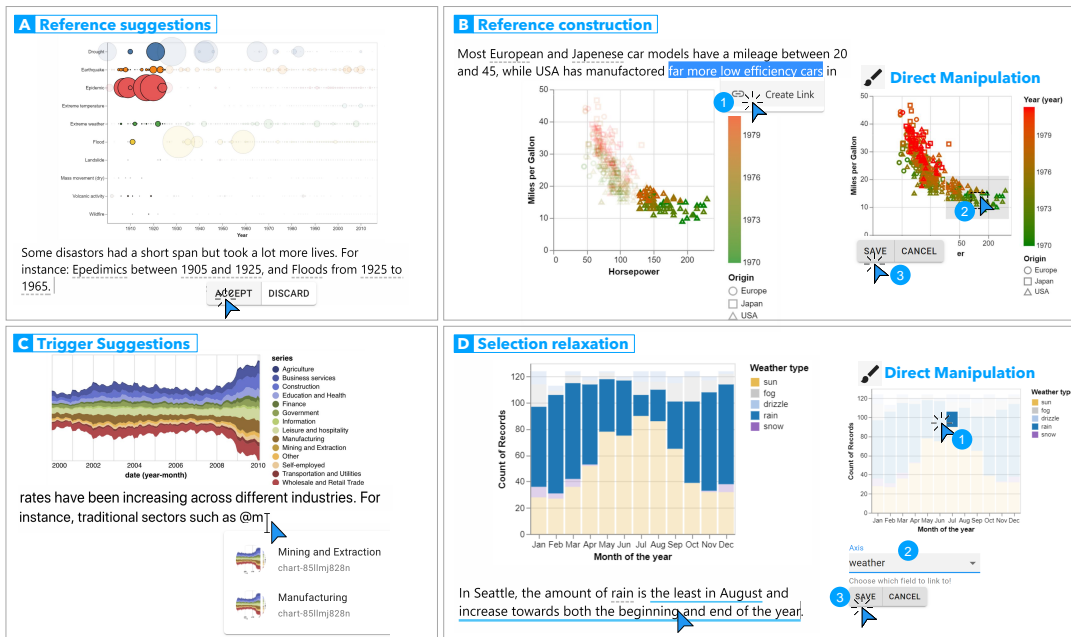


Figure 6.2: Salient features of Kori. (A) It uses natural language processing to suggest potential references between charts and the text while a user types. (B) Users can construct references by directly manipulating the visual marks on the chart. (C) It is possible to manually trigger suggestions. (D) In direct manipulation mode, users can easily expand their current selection to multiple visual marks of the same type. The encircled numbers mark the sequence of interactions for each activity.

In Kori, a visualization–text reference corresponds to an explicit linking of a text phrase to corresponding visual marks in the visualization. Since both representations correspond to the same data items, our visualization–text links are instantiations of conceptual cross-referencing between the two (see Section 2.3.2 for more details). According to the design space of implicit references between text and visualizations (Section 3.5), Kori supports the construction of *point*, *multi-point*, and *interval* references. It also suggests higher-level references but has no user interface to manually group individual references together yet.

6.3.2 Reference Detection

The automatic reference detection speeds up the document composition process. While Kori detects all three types of references, it only combines intervals with the corresponding axis of the chart. Once the references are identified, their presence is communicated to the author via visual cues without interrupting the authoring process (D_1 , D_3); they are underlined in gray. Authors can inspect them in due time or safely ignore them as they would not appear in the viewer mode. Authors can preview them by hovering over before accepting or discarding (Figure 6.2 A). Once accepted, they are shown with blue underline and will appear in the viewer as interactive references. Figure 6.3 shows our automatic reference detection approach that consists of the following four steps:

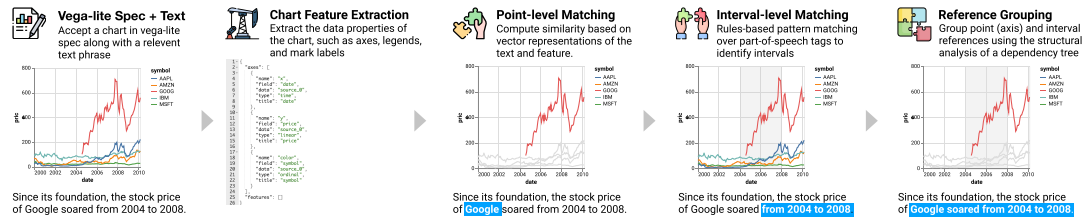


Figure 6.3: Four stages of our automatic reference suggestion pipeline. It accepts text and Vega-Lite specifications as input and begins with extracting features of a chart. These features are then matched against user text to find point references. The third step identifies numerical intervals in the user text. Finally, related references are grouped together to form higher level references.

I – CHART FEATURE EXTRACTION

Charts are composed of different types of encoding that map data values to visual properties (e.g., position, color, shape) of marks. The first step is to extract these data properties (e.g., axes labels, axes values and scales, legends, data labels) of a chart. The expressive JSON syntax of Vega-Lite enables easy access to this information. We loop through all kinds of visual encoding and extract their values in the underlying data. We also extract the axes properties, as this is particularly relevant for interval references. The extracted *features* serve as a knowledge base to match the user-typed text against and for finding potential references. For instance, the extracted features of the scatterplot in Figure 6.2 **B** are: x and y axes (horsepower and miles per gallon), legend categories (Europe, Japan, and the USA), and names of all car models (denoted by each dot).

II – POINT-LEVEL MATCHING

The next step detects the occurrences of chart features in a user-typed text. Our matching process uses vector representations of text obtained from FastText¹¹⁹, a neural network-based approach to obtain text representations that takes into account both word- and subword-level information (and therefore more resistant to noise than word-only approaches such as word2vec¹²⁰). We first use FastText to obtain vector representations of (i) chart features and (ii) n -gram representations in the sentence up to $n = 3$. For each chart feature, we then select the n -gram in the sentence that had the highest cosine similarity to the chart feature and present it as a potential link to the user if the threshold is greater than 0.5. Both the n -gram size and the similarity threshold were selected empirically to maximize the F_1 score against a small set of 55 manually annotated examples (see Figure 6.5). In addition to exact keyword matches, this vector-based matching process can tackle typographical errors, slight variations of the words or phrases (e.g., US, U.S., USA), abbreviations (e.g., EU, European Commission), synonyms (e.g., donating, contributing), and semantically similar words (e.g., Obama, democrat).

III – INTERVAL-LEVEL MATCHING

A rules-based approach on top of words and part-of-speech tags is used to identify numerical intervals in a sentence. We derived heuristics based on our observations in the design space analysis (Sec-

tion 3.5). We observed 116 sentences (16 from our collection, 100 gathered from diverse news articles on the web) that described one or more numerical intervals each and found the following list of frequent patterns (their frequencies in brackets):

more/less/fewer than X (45), X to/through Y (29), between X and Y (15), at least X (11), since/from X (4), below/above X (3)

The symbols X and Y denote either a number, date, or time, which is identified using the parts-of-speech tag `NUM` from Spacy. The combination of words and part-of-speech tags makes it convenient to derive compact rules to capture intervals. For instance, the rule ‘between X and Y ’ is identified with the part-of-speech pattern ‘`NUM CCONJ NUM`’, which detects phrases where two numbers (`NUM`) are joined by a coordinating conjunction (`CCONJ`); a complete list of patterns can be found in the supplemental material of the related research article.¹⁰⁰

IV – REFERENCE GROUPING

The final step is to combine the intervals with the correct axis-reference of the chart. In some cases, it may be possible to infer this by comparing the interval occurrence to chart axis values, for example in simple charts with a single numerical axis. However, if a sentence contains multiple intervals or a chart has multiple axes with overlapping axis extents, inferring the correct interval-axis combination is nontrivial. Surface-level heuristics which (for example) combine an interval with the nearest axis-reference is often inadequate as they do not take into account the syntactic and semantic structure of the sentence. To better account for sentence structure, we use its dependency tree (again obtained from Spacy) to map interval occurrence to their corresponding axis-references. Concretely, we map an interval occurrence to an axis-reference that is closest in the dependency tree distance, where we treat the dependency tree as an undirected graph and use Dijkstra’s algorithm to compute the distance between words. For cases where axis-reference and/or intervals consist of multiple words, we compute the tree distance between the phrase head words.

As an example, in Figure 6.4 (top) the sentence contains three axis names (`Count of Records`, $A_1 = \text{Minimum temperature}$, and $A_2 = \text{Maximum temperature}$) of a scatter plot and two intervals ($I_1 = \text{between 5 and 10}$ and $I_2 = \text{between 10 and 15}$). The shortest distance between A_1 and I_1 is 4 (orange and blue arcs) while A_1 is 5 edges (orange and green arcs) away from I_2 . Similarly, A_2 and I_1 are 5 edges apart, while the distance between A_2 and I_2 is 4. Since we combine a pair of axis and an interval when the shortest distance between them has the minimum value, this results in grouping of A_1 with I_1 and A_2 with I_2 as desired. In the case of ties based on dependency tree distance, we use the distance to the first common ancestor as a tiebreaker (i.e., the interval-axis combination that shares a closer ancestor is grouped together). If this second heuristic results in a tie, then we resort to the surface-level distance as the final tiebreaker. The bottom sentence in Figure 6.4 highlights a failure case where our approach wrongly groups the axis `price` with the interval `between 2006 and 2008` (distance 2 – orange and blue arcs) instead of combining it with `over $400` (distance 3 – orange and green arcs).

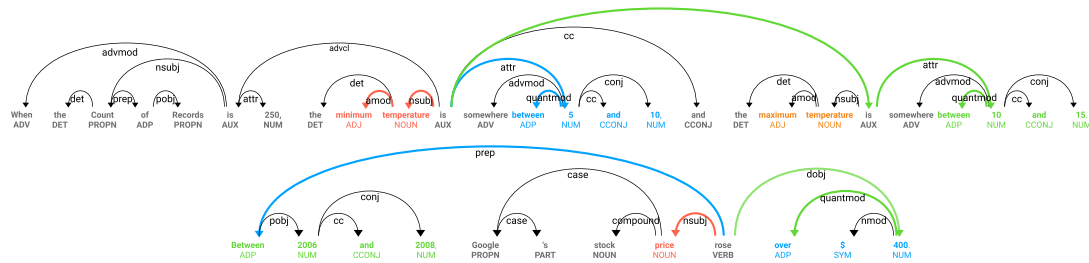


Figure 6.4: Dependency parsing for two sample sentences. (top) Success case: “minimum temperature” is successfully grouped with “between 6 and 10”. (bottom) Failure case: “price” is wrongly grouped with “between 2006 and 2008” as it is closer to “price” than “over \$400.”

6.3.3 Reference Construction

Often, authors already have references in mind while writing about a chart. Thus, they may not have to wait for reference suggestions. Instead, they can explicitly trigger suggestions by typing in the ‘@’ symbol (Figure 6.2 C). It shows a list of features or labels of charts that are currently active in the editor. Each item in the suggestion list is preceded by a thumbnail of the related chart, which helps authors quickly see what feature refers to which chart. The list offers an auto-complete; it keeps on filtering as the user types. The list of suggestions is also triggered by selecting a portion of the text.

The reference suggestions are limited and may not cover the full spectrum of possibilities that an author needs. Authors may want to reference a certain part of the chart, a few distinct visual marks that seem interesting in some context, or combine an arbitrary set of visual marks related to a message they are communicating. Kori offers a smooth interface to accomplish this using two distinct modes as shown in Figure 6.1 4 – 6 and Figure 6.2 B.

The first is the **direct manipulation** (Figure 6.2 B) mode. Authors can select a portion of text 1 and then directly brush visual marks of interest 2. Kori offers point, multi-point, and brushing selections to select visual marks. The system enables only the valid type of selections for a chart (e.g., no rectangular brushing for maps). Having selected the visual marks, authors can finalize the reference construction 3. The selection of visual marks is made efficient using the concept of relaxation of a selection⁶⁰. Relaxing a selection is simply a way of saying “select all items like this [selected] one”. This is particularly useful in situations where users want to reference many visual marks that are of the same type as a single or couple of selected marks. Figure 6.2 D explains an example where an author likes to create a reference to data dimension `Weather = rain`; she can simply do so by selecting any blue rectangle (rain) 1 and relaxing the selection to `Weather` 2 instead of the default dimension that was detected by the system.

While the direct manipulation mode is explicit, it is not flexible, especially in situations where a chart has overlapping marks or when an author wants precise referencing by combining multiple dimensions of a chart. The **filtering** mode (Figure 6.1 4 – 6) enables the addition of (upto) as many filters as the number of dimensions in a chart. Values of the dimensions can be adjusted using a multi-selection search field for categorical variables or an interval slider for numerical ones. The combination of multiple filters corresponds to the creation of higher-level references using *union* or

intersection operations, as discussed in Section 3.5. Kori combines multiple filters using the intersection operation.

Kori uses opacity as the default visual highlighting scheme when previewing the interactive references, as it does not mostly interfere with the existing visual encoding. Each chart is provided with a configuration panel—like the one in Figure 6.1 [10](#)—where the default values of opacity can be modified for each individual chart. Users can choose between two channels (opacity and fill). For the fill channel, we provide a color picker to choose active and inactive colors.

To showcase the flexibility of Kori and evaluate its usability, we conducted a two-fold evaluation. First, we quantitatively evaluated the automatic reference suggestion approach and compare it against a benchmark provided by Kong et al.⁸⁷ Second, we qualitatively evaluated the interface through a user study with 11 participants including visualization experts, novices, and interface designers.

6.4 Algorithmic Evaluation: Reference Detection

To quantitatively evaluate our reference detection approach, we need pairs of Vega-Lite chart specifications and corresponding text. Although we had previously collected 110 text–chart pairs (Section 3.2.1 – Study II), we did not have access to the underlying data for charts in all those examples. Therefore, we curated a dataset for evaluation.

6.4.1 Dataset Curation

We had data for 42 / 110 text–chart pairs and their images (Kong et al.’s⁸⁷ collection). We reconstructed these charts in Vega-Lite editor to get Vega-Lite specifications. However, they were all bar charts and included very few interval references. We needed a diversity of charts as well as more examples. Therefore, we augmented the 42 text–chart pairs with 50 additional pairs as follows: We extracted 116 instances of real sentences—including interval references—about a variety of different charts from our data collection (Section 3.2.1 – Study II). Besides, we collected 34 diverse charts from example galleries of different visualization libraries (e.g., Vega, Vega-Lite, D3, Observables). Then, we mapped these charts to instances of 116 real-sentence collection. A junior researcher (co-author of the corresponding research paper¹⁰⁰) manually rephrased the sentences to match the data on the charts while keeping the essence as close to the original sentences as possible. A senior researcher (also one of the co-author in the paper¹⁰⁰), then, went through these sentences to make sure they are both syntactically and semantically correct. Finally, we annotated each text–chart pair for point, interval, and group references to create the ground truth. The resulting dataset includes 92 chart–text pairs with 15 different types of charts. This dataset was randomly split into a validation (60%, 55 pairs) and a test (40%, 37 pairs) dataset.

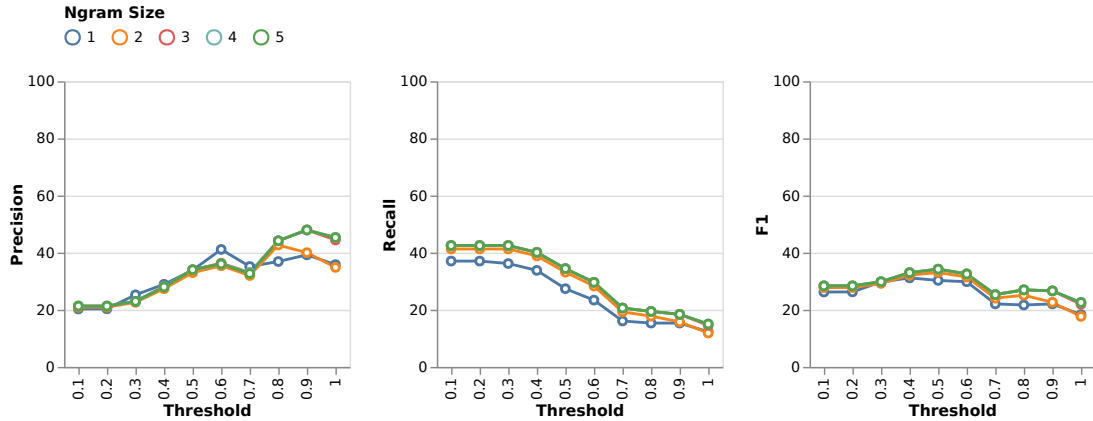


Figure 6.5: Quantitative evaluation of the reference detection approach. Precision, recall, and F_1 at various maximum n -gram sizes for the validation dataset (55 text-chart pairs).

6.4.2 Results

Figure 6.5 shows precision, recall, and F_1 scores* for varying similarity thresholds for various maximum n -gram sizes on the validation dataset. We selected the threshold of 0.5 and the maximum n -gram size of 3 as these values maximized the F_1 score (0.34). The higher values of F_1 are desirable— $F_1 = 1$ would mean that the system correctly suggested all references, while $F_1 = 0$ means no correct suggestion.

For the test dataset, our pipeline correctly identified 57 references (out of 137 true references), produced 90 incorrect references (false positives), and missed 93 references (false negatives). A desirable outcome is to maximize the correctly identified references while keeping the false positives and false negatives low. Kori correctly suggested 42 / 84 point, 9 / 26 interval, and 5 / 27 group references. While our approach worked well for identifying point references ($F_1 = 0.47$), the interval detection ($F_1 = 0.25$), and reference grouping ($F_1 = 0.26$) proved to be more challenging. Since too many incorrect references can be bothersome for users, another option could be to go for higher precision ($n=5$, threshold=0.8). This results in fewer (32) false positives.

When running our approach on Kong et al.’s dataset (42 text-chart pairs), we obtained an average distance ($1 - F_1$; lower values are better) of 0.57 from the gold standard (annotated examples by experts) compared to 0.39 produced by their approach. Although quantitatively, references extracted by Kong et al. are closer to those in the gold standard, we detect references automatically while they rely on user intervention to extract base references (distance to gold standard = 0.54) and then automatically refine them that reduces the distance to 0.39.

Several problems contribute to the lower F_1 scores. One problem with general purpose pre-trained vector representations of text is that it matches words that are dissimilar in the context of a chart (e.g., “Obama” matches both chart categories “Democrats” and “Republicans” with cosine similarity of 0.58 and 0.56 respectively). A similar problem occurs with numbers. Numbers are important in dealing

*The F_1 -score is a measure of a classifier’s accuracy. It combines precision and recall into a single metric by taking their harmonic mean.

with charts and can be described in different ways (e.g., 12, twelve). FastText had problems matching “60” to “sixty” in the sentence “*Most movies have a rating between sixty and 100*”, and instead matched “60” to “100”, presumably because arabic numeral representations of numbers are closer in the vector space. While the parts-of-speech tagger of Spacy NUM could identify numbers with units, it has no support for dates (e.g., months, days); they are tagged as proper nouns (NNP). Therefore, intervals like Apr to Jun could not be detected. Some failure cases in the grouping, like the one in Figure 6.4 (bottom sentence), can be avoided if we consider the extents, scales, and units of numerical axes in the chart in addition to distances in the dependency tree space.

6.5 User Evaluation: The Kori System

We conducted a user study to gain insights into the usefulness and usability of the Kori system. The focus is on evaluating how people interacted with the system to author an interactive data document.

6.5.1 Participants

We recruited 11 participants (P1–P11; five male and six female) with diverse backgrounds ranging from undergrad students (3 – P2, P9, P10) and grad students (3 – P5, P6, P11) to visualization experts (4 – P1, P3, P4, P8) and a user interface designer (P7). All participants had experience using document editing tools like Microsoft Word, Google Docs, or similar. All participants mentioned that they had created charts using data science tool-kits (e.g., R, Python) or programming libraries (e.g., D3, Vega-Lite). All participants except P10 regularly create charts as part of their job or studies. Eight participants mentioned having worked with charts in a word processing tool (e.g., Google Docs, Microsoft Word).

6.5.2 Procedure and Tasks

In sessions lasting about 60 minutes each, the participants used Google Chrome on their personal computers to access the tool in online video call with an experimenter sharing their screens. After collecting the above demographic information, every session began with a brief introduction to the project followed by a short tutorial. We demonstrated the main features of Kori using a variety of chart types.

In the main part of the study, the participants had to complete three tasks. First, to familiarize themselves with the system, we asked them to replicate an example from the tutorial, which contains a bar chart and a paragraph of text with three interactive references. Second, the participants had to reproduce a given but previously unseen example. Two charts (a scatterplot and a heatmap) and two paragraphs of text were provided with nine references. We marked the references in the text, and participants had to transform them into interactive references according to their understanding. We tried to maximize the diversity of references so that participants had to use every feature of Kori to construct them. In the third task, the participants had to design a short story (5 to 6 sentences) on

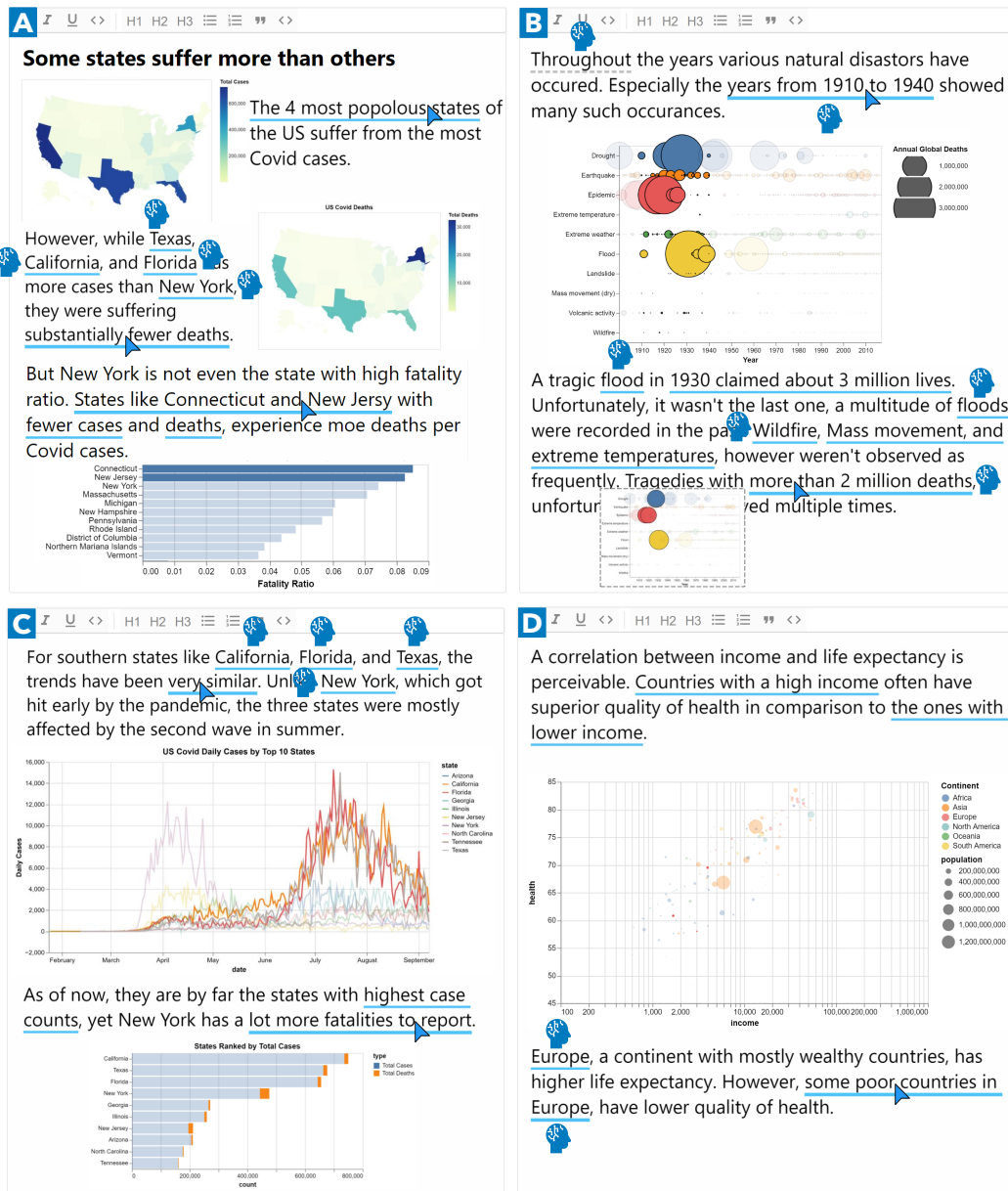


Figure 6.6: Copy-edited excerpts of stories created by participants. The symbol marks the reference suggestions. (A) P3 relied on reference construction using direct manipulation . (B) In contrast, P8 got 9 suggestions (1 wrong). She manually constructed 2 references: "1930 claimed about 3 million lives." and combined two suggestions ("mass movement" and "extreme temperatures") into a single reference. (C) P5 used the first reference "very similar" to verify a fact he was describing. (D) P6 got only two suggestions and created most references using the filtering interface.

one of the three scenarios (they were free to pick any) that were provided to them in the form of one or more charts.

We concluded the session with a reflection survey on the usability and usefulness rating different statements on a 5-point Likert scale (1-strongly disagree, 5-strongly agree). Moreover, we discussed the overall user experience, potential improvements, and limitations in semi-structured interviews.

6.5.3 Results

All participants successfully completed all three tasks within the limited session time. The first two tasks were marked as complete when the participants had created all required references. They created references in Task 1 (3 references) and Task 2 (9 references) with minimal intervention from the moderator. For these two tasks, Kori successfully suggested 6 references (2 for Task 1 and 4 for Task 2) as expected. In Task 3, every participant created a short story with one or more charts; Figure 6.6 shows four example stories. In total, Kori suggested 64 references for Task 3. (The small boxplots show the distribution of suggestions for all 11 participants). These 64 suggestions also include 25 instances where participants explicitly triggered suggestions. Among these, 48 references (including 25 explicitly triggered suggestions, all correct point references) were correct and 16 incorrect. The incorrectly suggested references were rarely ignored (3) and mostly discarded (13). Besides, the participants manually created 40 references.

For point references, they often relied on automatic suggestions. We observed comparatively fewer instances of explicit triggering by typing @ than that of selecting a text phrase. While participants, P3 (Figure 6.6 A) and P6 (Figure 6.6 D), largely relied on the manual construction of references, P8 (Figure 6.6 B) made use of more automatic suggestions. We observe the frequent use of direct manipulation mode for geographical maps, bar charts, and line plots. Comparatively, the participants employed filtering mode comparatively more frequently for all types of charts, especially for scatter plots and bubble charts.

Participants rated their satisfaction of the overall experience (median, mode = 4, IQR = 0, 5) and usefulness of the tool (median, mode = 5, IQR = 0, 5). While participants found all features of the tool intuitive and self-explanatory, they rated the reference construction interface (median, mode = 4, IQR = 1, 5) higher than the automatic suggestions (median, mode = 4, IQR = 0, 5).

All participants mentioned that the tool is intuitive, easy to use, and has high learnability. Two participants (P3, P4) even said that they would have easily discovered all the features without a demonstration. Participants specifically liked the reference construction interface. P4 praised the direct manipulation of regions on the map to construct a reference and described it as easy as “click regions of interest and done. Impressive!”. P5 stated, “It is surprising that manual linking is so flexible and works with so many chart types.” P6 appreciated two different modes of reference construction. P11 favored the filtering mode over direct manipulation, saying it is more flexible and precise. The participant further elaborated on its usefulness by adding: “Finding one data point in millions of data points is just impossible in the brushing interface.” (This comment was in response to a scatter plot with a lot of overlapping points.) P9 complained about not being able to brush the color legend of a chart (Kori only supports the direct manipulation of marks inside the chart area).

All participants agreed that the automatic suggestion is valuable and complements the reference construction yet criticized that it is not smart enough. P9 stated that the automatic suggestions are helpful and work reliably for simple cases. P1 commented, *“suggestions are cool but not too smart.”* Nonetheless, none of the participants considered suggestions as distracting. On the contrary, P2 and P7 suggested making their presence more noticeable. However, wrongly identified references were a bother for one participant (P11). Three participants (P2, P7, P10) complained about long delays that often took place in automatic suggestions. P8 remarked that the suggestions are often too short, and their grouping needs to be better supported using the interface—the participant wanted to group two references that were correctly suggested in a sentence (Kori does not support grouping of suggested references).

Surprisingly, several participants (P5, P6, P9, P11) came up with a different use case that we did not have in mind. They used interactive references as a means to explore the data. P5 described his experience as *“I wasn’t sure about a statement”*—Task 3: the participant had an assumption that California, Florida, and Texas have similar trends for Covid-19 but was not sure (Figure 6.6 C)—*“and I could verify it using interactive linking. Readers can do too.”* P6 reported that interactive linking provides insights and helps in understanding the data. She described her opinion as *“I would say it’s not only about writing but also [for] doing analysis while writing.”* Two participants (P1, P5) suggested having an embedded chart creation interface. Without such an interface, it is limited to what could be explored, P1 pointed out and said, *“Charts were limited and non-modifiable. Kinda limits the scope of investigative reporting.”* Several participants highlighted the lack of some convenience features, such as editing a reference that has been suggested or created, deleting a reference without deleting its text, and a user interface to group the suggested references.

When asked about the usefulness of Kori, most participants (P1–P6, P11) saw its value in creating information for reporting, presentation, and communication scenarios. P11 highlighted the benefit of interactive references as *“Readers may not interpret my intention correctly. This tool makes it much clearer by creating explicit references to make sure readers look at what I intended.”* Similarly, P8 mentioned that visual highlighting through interactive links may be better than adding direct annotations on top of a chart, as it could result in visual clutter.

6.6 Limitations, Challenges, and Future Opportunities

This section introduced, Kori, a mixed-initiative system to support authors in creating interactive data documents containing references between the text and charts. The analysis of visualization–text references in existing articles (Section 3.5) informed the design of Kori. Findings on point and interval references, as well as the grouping of the references, guided the development of the automatic suggestion pipeline. Besides, a flexible manual interface provides a complementary way to construct references. While the two-fold evaluation reveals that Kori is a valuable, useful, and easy-to-use authoring tool, it has several limitations that may lead to interesting challenges and possible future directions.

Reliability of automatic suggestions – Both algorithmic and user evaluation of Kori reveal that the automatic linking feature was valuable, yet limited and prone to many false negatives. A key technical challenge is to make the automatic linking as reliable and accurate as possible. However, due

to the small size of our training dataset, Kori relied on a heuristics-based approach to identify text–visualization links. Although some stages of the 4-stage pipelined approach (Section 6.3.2) leverage recent advances in natural language processing (e.g., a pre-trained state-of-the-art neural dependency parser³⁷) and has better accuracy, the approach does not learn a single model that integrates all these pipelines into one system. Therefore, it would be interesting to explore an approach that learns a fully end-to-end model that identifies all references between a body of text and its associated chart directly from the raw input. However, such a system would require collecting a fairly large training set of annotated examples, but may allow for the use of more sophisticated contextualized embedding models (e.g., BERT or Transformers as opposed to static FastText) that can be subsequently fine-tuned on the collected training set. Another problem with the automatic suggestion approach is that, currently, it does not resolve the user’s intention on which chart to link to. As a result, we observed that it often creates an interactive link to a wrong chart rather than the one expected by the user, especially when many charts share the same underlying dataset. Resolving such ambiguity in user intent can be challenging without the user’s explicit input. A potential resolution method would consider the user’s current cursor or the distance from the text to the chart.

Linking to many charts – Another related issue is that Kori only supports one-to-one mapping, resulting in a unidirectional link from one text phrase to a single chart. However, in some cases, it might be desirable to create a reference from one text phrase to many charts that may show different aspects of the data for the same text phrase. Such linking might facilitate, for instance, comparisons across multiple charts. However, it is unclear whether supporting many-to-many mapping models is ideal. Not only the authoring interface would become complicated, but also the benefit for reading may be limited or even adversarial. A user study would be useful to shed light on its advantage.

Better reading support – The user study focused on the authoring interface as it is the core contribution compared to existing research focusing on the reading experience. However, once links are established, they can facilitate the reading process. In this work, our primary mode of operation to address the split-attention effect in a visualization–text reference is through interactive highlighting. According to the cognitive load theory and the cognitive theory of multimedia learning, our highlighting approach follows the signaling principle or temporal contiguity principle.¹¹² However, we also foresee many opportunities to provide an improved reading experience beyond simply revealing references upon hovering over a text phrase. For instance, the user might want to see all relevant text phrases linked to a specific chart for a better understanding of the underlying data context. To meet this need, the tool can annotate related visual marks in the chart with the relevant text phrases. This is similar to the *gather* operation in exploring embedded word-scale visualizations by Goffin et al.⁵¹, while it is the opposite approach to *Elastic Documents*¹¹ collecting relevant visualizations for a selected phrase.

Suggestions for data-driven text – In contrast to the fully automated solutions described in Chapter 4, Kori does not yet offer any help regarding the automatic generation of data-driven text. It only suggests links in the author written text. Integrating further automation in this regard is another promising direction. It is possible to automatically generate the text that describes the data (as discussed in Chapter 4 and Chapter 5). Beyond these approaches, there has also been a flurry of recent work on data-driven table-to-text generation with deep learning techniques.¹⁸³ Adapting such meth-

ods to suggest short data-driven texts in the context of chart-to-text generation may hint at important analysis aspects to the author of the document and is, therefore, an interesting future extension of Kori.

Data analysis and chart construction support – As suggested by some participants during the user evaluation, Kori lacks an interface for analyzing the data and construction of visualizations. Consequently, this limits the scope of exploratory data analysis inside the tool—that is required for writing about data—as users have to rely on external tools to analyze and import their created visualizations to Kori. With the current technology stack of Kori, it would be straightforward to connect or even include Voyager¹⁸⁴ in Kori’s user interface. Voyager* allows for exploratory data analysis using an easy-to-use graphical user interface and can export charts as Vega-Lite specifications. The inclusion of such an interface inside Kori would be a step toward making it an independent analysis and writing tool that requires no programming at all.

*<http://vega.github.io/voyager>

7

Conclusion

The dissertation has investigated an underexplored—*communication*—aspect of visualization systems by proposing the idea of *interactive data documents* that bridges the gap between exploratory and explanatory representations of data. The design of these interactive data documents is informed by the empirical findings of data-driven stories, in particular with respect to the interplay of text and visualizations. The understanding of this interplay along with the concepts of explorable explanations¹⁷⁷ and exploration¹⁸⁹ inspire the design of an interactive linking model between text (or audio) and visualization for a bimodal data representation. Targeting a broader audience, this thesis then introduces a generalizable concept for automatically producing interactive data documents that offers *exploration* as it explains the data and background in a textual way as well as it supports the exploration of the data through interactive visualizations. The applicability and usefulness are shown by instantiating the concept to a number of different application domains and datasets (bibliographic, bivariate geographic, code quality, and knowledge graph data) through a series of web-based prototypical solutions (VIS Author Profiles, Interactive Map Reports, Code Quality Documents, and VisKconnect). Going beyond 2D representations of data, the thesis has also discussed the proposed concept for virtual reality visualizations. Finally, considering the challenges and limitations of existing authoring tools for producing such interactive content, the thesis explored an easy-to-use authoring solution that leverages a mixed-initiative interface to author an interactive data document.

This chapter concludes the thesis with a discussion of potential avenues of future research, either stemming from the limitations of the presented generic approach and specific prototype solutions or related to possible natural extensions.

7.1 Limitations and Future Directions

Reflecting on the proposed generic approach and thinking beyond the individual systems, the following section highlights three different aspects—that may be considered as limitations to some extent—with concrete suggestions for further improvement and future work.

7.1.1 Detailed User Evaluation

The developed individual prototype systems—that are instantiations of the proposed generic concept of interactive data documents—are evaluated, and their usefulness is already demonstrated in a variety of different but specific application scenarios. However, it cannot yet be claimed that this combination of longer textual explanations and visualization would always perform better than a purely visualization-based representation as the two approaches—the proposed interactive data documents and pure visualization system—were never contrasted in a user study. For such comparisons, it is needed to first develop the pure visualization-based representation to show the equivalent information. Then, a user study can be performed to compare the two representations. A quantitative study could answer which of the representations is better with respect to accuracy and answer times. Likewise, a qualitative approach could reflect on how self-explaining the representations are and how users work with them. The textual explanations might influence the way users interpret the visualizations. Existing research has already discovered that informative captions and titles can reduce the mental effort to process a data visualization.^{86,180} However, the impact of longer textual explanations on the accompanying visualizations remains yet to be explored and could be an interesting future direction.

In interactive data documents, the split-attention effect between text and visualizations—resulting from bimodal data representation—is primarily handled through interactive visualization–text linking, positional linking, consistent color linking, and word-sized graphics integrated into the text. According to the cognitive load theory and the cognitive theory of multimedia learning, these linking methods follow the signaling principle or temporal contiguity principle.¹¹² Both theories suggest integrating information at the temporal dimension (i.e., through interactive referencing) or at the spatial dimension, reducing the spatial proximity (through word-sized graphics or including text inside visualizations) between the text and visualizations. To reaffirm and quantify how well these linking methods work in reducing the split-attention effect in a data document, quantitative studies using eye tracking would be beneficial. Such studies would enable a precise understanding of the reduced split-attention phenomenon by studying eye fixations on certain regions of the data document and how often gaze motions occur from one region to another.

7.1.2 Advance Natural Language Text Generation

In this thesis, the proposed approach has mostly relied on relatively simple template-based natural language text generation due to its deterministic nature and sufficient flexibility. While such an approach is easier to design and control, it suffers from domain dependence, would require large maintenance efforts (for instance, in case of data extensions), and is not generalizable to different application domains or datasets. With the recent advances in natural language generation (e.g., sophisticated mod-

els like Generative Pre-trained Transformer 3), many researchers are exploring table-to-text¹⁸⁷ and data-to-text¹³⁴ generation approaches to produce text from structured data representations. Unlike template-based approaches, these approaches use neural networks and are end-to-end trained without explicitly modeling what to say and in what order. Another future direction would be to explore such advanced approaches as a replacement to the template-based text generation.

Although these approaches require substantial training data, once trained, they may generate more natural looking text and be more generalizable. However, on the downside, these approaches are usually harder to control and predict what they would generate. Therefore, another challenge would be to integrate these explanations with the visual views through the proposed linking model. Probably, in this case, another step would be required to automatically analyze the response of these advanced generation approaches to extract potential links between the generated text and visualization (like the one described in Section 6.3.2).

7.1.3 Closing the Loop from Explanation to Exploration in Authoring Tools

During the user evaluation of the authoring tool Kori, we observed an interesting insight that users—participants of the user study—often use the authoring interface to explore the data. For instance, they used suggested references as opportunities to examine detailed aspects of the data and actively inspected the data by trying out different selections in the manual construction interface. Kori, in contrast, was designed under the assumption that users would already have explored the data in any analysis tool of their choice and had prepared visualizations they would need before beginning to draft an interactive document. Consequently, Kori did not include a visualization creation interface.

However, observing the usage of Kori by participants while creating interactive content, it becomes clear that exploration and explanation go hand in hand during the composition of an interactive data document. Explanation of an insight might lead to the need of exploring another related aspect of data—for instance, to quickly test a hypothesis that comes to the author’s mind. This may require the construction of another entirely new visualization or modifying the existing visualization. This observation demonstrates a potential need for augmenting authoring tools like Kori to support data exploration within the explanation interface, supporting the full life cycle of data analysis and communication.

What is particularly interesting in this direction is how we can support creating visualizations to explore additional aspects of the data, starting from text phrases or already created interactive references. In this regard, it might be possible to include further automation for suggesting other relevant—or even alternative—visualizations based on the already authored text, similar to interactive widgets in Voder¹⁶¹. That is, going back to exploration from drafting explanations and then coming back is particularly underexplored in the literature.

7.2 Outlook

The principal idea of this dissertation is to create an expressive, self-explanatory, and exploratory data representation that goes beyond traditional visualization systems and data-driven storytelling. The

resulting representation targets an adequate balance between these two extremes—exploration and explanation—to communicate data analysis results to a broader audience in a way that is neither as restrictive as solely following the author’s line of arguments without any flexibility of exploration nor too unguided, complex, and overwhelming to explore. In a nutshell, the resulting data representation, interactive data documents, can still be classified as a visual analytics solution that puts a lot more emphasis on presentation, storytelling, and dissemination.

I believe that this explorative data representation can prove to be a powerful and effective model to enhance the outreach of visualizations, increase transparency and trust in data communication, and foster deep understanding.

References

- [1] Authorea. <https://www.authorea.com/>. Accessed: 2020-08-10.
- [2] Elasticsearch. www.elastic.co/elasticsearch. Online; Accessed 20 June 2021.
- [3] Mirror of Apache Xerces2 Java. https://github.com/apache/xerces2-j/tree/xerces_j_1. Accessed: 2019-07-22.
- [4] Vega editor. <https://vega.github.io/editor>. Accessed: 2021-03-30.
- [5] M. Abdelaal, F. Heimerl, and S. Koch. ColTop: Visual topic-based analysis of scientific community structure. In *Proceedings of the 2017 International Symposium on Big Data Visual Analytics, BDVA*, pp. 1–8. IEEE, 2017. doi: 10.1109/BDVA.2017.8114622
- [6] M. J. Adler and C. Van Doren. *How to Read a Book*. Simon and Schuster, 1972.
- [7] F. Amini, N. Henry Riche, B. Lee, A. Monroy-Hernández, and P. Irani. Authoring data-driven videos with dataclips. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):501–510, 2017. doi: 10.1109/TVCG.2016.2598647
- [8] P. Ayres and J. Sweller. *The Split-Attention Principle in Multimedia Learning*, p. 135–146. Cambridge Handbooks in Psychology. Cambridge University Press, 2005. doi: 10.1017/CBO9780511816819.009
- [9] P. Ayres and J. Sweller. The split-attention principle in multimedia learning. In R. E. Mayer, ed., *The Cambridge Handbook of Multimedia Learning*, pp. 135–146. Cambridge University Press, 2005.
- [10] B. Bach, Z. Wang, M. Farinella, D. Murray-Rust, and N. Henry Riche. Design patterns for data comics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–12. ACM, 2018. doi: 10.1145/3173574.3173612
- [11] S. K. Badam, Z. Liu, and N. Elmqvist. Elastic Documents: Coupling text and tables through contextual visualizations for enhanced document reading. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):661–671, 2018. doi: 10.1109/TVCG.2018.2865119
- [12] J. Bansiya and C. G. Davis. A hierarchical model for object-oriented design quality assessment. *IEEE Transactions on Software Engineering (TSE)*, 28(1):4–17, 2002. doi: 10.1109/32.979986
- [13] O. Barral, S. Lallé, and C. Conati. Understanding the effectiveness of adaptive guidance for narrative visualization: a gaze-based analysis. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pp. 1–9, 2020. doi: 10.1145/3377325.3377517
- [14] F. Beck, T. Blascheck, T. Ertl, and D. Weiskopf. Word-sized eye-tracking visualizations. In *Workshop on Eye Tracking and Visualization*, pp. 113–128. Springer, 2015.

- [15] F. Beck, S. Koch, and D. Weiskopf. Visual analysis and dissemination of scientific literature collections with SurVis. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):180–189, 2016. doi: 10.1109/TVCG.2015.2467757
- [16] F. Beck, O. Moseler, S. Diehl, and G. D. Rey. In situ understanding of performance bottlenecks through visually augmented code. In *Proceedings of the International Conference on Program Comprehension (ICPC)*, pp. 63–72. IEEE, 2013. doi: 10.1109/ICPC.2013.6613834
- [17] F. Beck, H. A. Siddiqui, A. Bergel, and D. Weiskopf. Method Execution Reports: Generating text and visualization to describe program behavior. In *Proceedings of the 5th IEEE Working Conference on Software Visualization*, pp. 1–10. IEEE, 2017. doi: 10.1109/VISSOFT.2017.11
- [18] F. Beck and D. Weiskopf. Word-sized graphics for scientific texts. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 23(6):1576–1587, 2017. doi: 10.1109/TVCG.2017.2674958
- [19] M. Bostock, V. Ogievetsky, and J. Heer. D3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, Dec 2011. doi: 10.1109/TVCG.2011.185
- [20] D. Braun, E. Reiter, and A. Siddharthan. SaferDrive: An NLG-based behaviour change support system for drivers. *Natural Language Engineering*, pp. 1–38, 2018. doi: 10.1017/S1351324918000050
- [21] M. Brehmer, B. Lee, N. Henry Riche, D. Tittsworth, K. Lytvynets, D. Edge, and C. White. Timeline Storyteller: The design & deployment of an interactive authoring tool for expressive timeline narratives. In *Proceedings of the Computation + Journalism Symposium.*, 2019.
- [22] C. Brewer and A. J. Campbell. Beyond graduated circles: varied point symbols for representing quantitative data on maps. *Cartographic Perspectives*, (29):6–25, 1998. doi: 10.14714/CP29.672
- [23] C. A. Brewer, A. M. MacEachren, L. W. Pickle, and D. Herrmann. Mapping mortality: Evaluating color schemes for choropleth maps. *Annals of the Association of American Geographers*, 87(3):411–438, 1997. doi: 10.1111/1467-8306.00061
- [24] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.
- [25] H. Burden and R. Heldal. Natural language generation from class diagrams. In *Proceedings of the International Workshop on Model-Driven Engineering, Verification and Validation (MoDeVVA)*, pp. 8:1–8:8. ACM, 2011. doi: 10.1145/2095654.2095665
- [26] J. C. Castro-Alonso, P. Ayres, and J. Sweller. Instructional visualizations, cognitive load theory, and visuospatial processing. In *Visuospatial processing for education in health and natural sciences*, pp. 111–143. Springer, 2019.
- [27] T. Chandler, M. Cordeil, T. Czauderna, T. Dwyer, J. Glowacki, C. Goncu, M. Klapperstueck, K. Klein, K. Marriott, F. Schreiber, et al. Immersive analytics. In *2015 Big Data Visual Analytics*, pp. 1–8. IEEE, 2015. doi: 10.1109/BDVA.2015.7314296
- [28] R. Chang, C. Ziemkiewicz, T. M. Green, and W. Ribarsky. Defining insight for visual analytics. *IEEE Computer Graphics and Applications*, 29(2):14–17, 2009. doi: 10.1109/MCG.2009.22

- [29] S. R. Chidamber and C. F. Kemerer. A metrics suite for object oriented design. *IEEE Transactions on Software Engineering (TSE)*, 20(6):476–493, 1994. doi: 10.1109/32.295895
- [30] A. Choudhry, M. Sharma, P. Chundury, T. Kapler, D. W. Gray, N. Ramakrishnan, and N. Elmqvist. Once upon a time in visualization: Understanding the use of textual narratives for causality. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1332–1342, 2021. doi: 10.1109/TVCG.2020.3030358
- [31] M. Conlen and J. Heer. Idyll: A markup language for authoring and publishing interactive articles on the web. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18, pp. 977–989. ACM, 2018. doi: 10.1145/3242587.3242600
- [32] K. A. Cook and J. J. Thomas. Illuminating the path: The research and development agenda for visual analytics. Technical report, Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2005.
- [33] L. F. Cortés-Coy, M. Linares-Vásquez, J. Aponte, and D. Poshyvanyk. On automatically generating commit messages via summarization of source code changes. In *Proceedings of the IEEE International Working Conference on Source Code Analysis and Manipulation (SCAM)*, pp. 275–284. IEEE, 2014. doi: 10.1109/SCAM.2014.14
- [34] R. Dale, S. Geldof, and J.-P. Prost. Using natural language generation in automatic route description. *Journal of Research and practice in Information Technology*, 37(1):89, 2005.
- [35] R. B. Dean and W. Dixon. Simplified statistics for small numbers of observations. *Analytical Chemistry*, 23(4):636–638, 1951. doi: 10.1021/ac60052a025
- [36] S. Demir, S. Carberry, and K. F. McCoy. Summarizing information graphics textually. *Computational Linguistics*, 38(3):527–574, 2012. doi: 10.1162/COLI_a_00091
- [37] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pp. 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1423
- [38] M. Dörk, N. Henry Riche, G. Ramos, and S. Dumais. PivotPaths: Strolling through faceted information spaces. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2709–2718, 2012. doi: 10.1109/TVCG.2012.252
- [39] S. Dutta and T. J. Overbye. Information processing and visualization of power system wide area time varying data. In *Proceedings of the 2013 IEEE Symposium on Computational Intelligence Applications in Smart Grid*, pp. 6–12. IEEE, 2013.
- [40] M. E. Elmer. *Symbol Considerations for Bivariate Thematic Mapping*. PhD thesis, University of Wisconsin–Madison, 2012.
- [41] N. Elmqvist and P. Tsigas. CiteWiz: a tool for the visualization of scientific citation networks. *Information Visualization*, 6(3):215–232, 2007. doi: 10.1057/palgrave.ivs.9500156
- [42] 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*, 23(9):2179–2198, 2017. doi: 10.1109/TVCG.2016.2610422

- [43] T. G. Filó, M. Bigonha, and K. Ferreira. A catalogue of thresholds for object-oriented software metrics. In *Proceedings of the International Conference on Advances and Trends in Software Engineering (SOFTENG)*, pp. 48–55, 2015.
- [44] H. T. Fisher. Mapping information: The graphic display of quantitative information. *ABT BOOKS, 55 WHEELER ST., CAMBRIDGE, MA 02138, USA, 1983, 384*, 1983.
- [45] J. J. Flannery. The relative effectiveness of some common graduated point symbols in the presentation of quantitative data. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 8(2):96–109, 1971. doi: 10.3138/J647-1776-745H-3667
- [46] T.-L. Fung, J.-K. Chou, and K.-L. Ma. A design study of personal bibliographic data visualization. In *Proceedings of the 2016 IEEE Pacific Visualization Symposium*, PacificVis, pp. 244–248. IEEE, 2016. doi: 10.1109/PACIFICVIS.2016.7465279
- [47] B. Ganter and R. Wille. *Formal concept analysis: mathematical foundations*. Springer Science & Business Media, 1st ed., 2012. doi: 10.1007/978-3-642-59830-2
- [48] A. Gatt and E. Kraemer. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170, 2018. doi: 10.1613/jair.5477
- [49] A. Gatt and E. Reiter. SimpleNLG: A realisation engine for practical applications. In *Proceedings of the 12th European Workshop on Natural Language Generation*, pp. 90–93. Association for Computational Linguistics, 2009.
- [50] D. Gkatzia, O. Lemon, and V. Rieser. Data-to-text generation improves decision-making under uncertainty. *IEEE Computational Intelligence Magazine (CIM)*, 12(3):10–17, 2017. doi: 10.1109/MCI.2017.2708998
- [51] P. Goffin, P. Isenberg, T. Blascheck, and W. Willett. Interaction techniques for visual exploration using embedded word-scale visualizations. In *CHI 2020 - Conference on Human Factors in Computing Systems*. ACM, 2020. doi: 10.1145/3313831.3376842
- [52] P. Goffin, W. Willett, J. D. Fekete, and P. Isenberg. Exploring the placement and design of word-scale visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2291–2300, Dec 2014. doi: 10.1109/TVCG.2014.2346435
- [53] P. Goffin, W. Willett, J.-D. Fekete, and P. Isenberg. Design considerations for enhancing word-scale visualizations with interaction. In *Posters of the Conference on Information Visualization*. IEEE, 2015.
- [54] S. Gottschalk and E. Demidova. EventKG—The hub of event knowledge on the web—and biographical timeline generation. *Semantic Web*, 10(6):1039–1070, 2019.
- [55] S. Gratzl, A. Lex, N. Gehlenborg, N. Cosgrove, and M. Streit. From visual exploration to storytelling and back again. In *Computer Graphics Forum*, vol. 35, pp. 491–500. Wiley Online Library, 2016.
- [56] D. Han, G. Parsad, H. Kim, J. Shim, O.-S. Kwon, K. A. Son, J. Lee, I. Cho, and S. Ko. HisVA: a visual analytics system for learning history. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021. doi: 10.1109/TVCG.2021.3086414
- [57] W. Härdle and L. Simar. *Applied multivariate statistical analysis*, vol. 22007. Springer, 2007.

- [58] M. Harward, W. Irwin, and N. Churcher. In situ software visualisation. In *Proceedings of the Australian Software Engineering Conference (ASWEC)*, pp. 171–180. IEEE, 2010. doi: 10.1109/ASWEC.2010.18
- [59] Q. He, M. Zhu, B. Lu, H. Liu, and Q. Shen. MENA: Visual analysis of multivariate egocentric network evolution. In *Proceedings of the 2016 International Conference on Virtual Reality and Visualization, ICVRV*, pp. 488–496. IEEE, 2016. doi: 10.1109/ICVRV.2016.88
- [60] J. Heer, M. Agrawala, and W. Willett. Generalized selection via interactive query relaxation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 959–968, 2008.
- [61] N. Henry, J.-D. Fekete, and M. J. McGuffin. NodeTrix: a hybrid visualization of social networks. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1302–1309, 2007. doi: 10.1109/tvcg.2007.70582
- [62] M. Hlawatsch, F. Sadlo, H. Jang, and D. Weiskopf. Pathline Glyphs. *Computer Graphics Forum*, 33(2):497–506, 2014.
- [63] D. C. Hoaglin, F. Mosteller, and J. W. Tukey. *Understanding Robust and Exploratory Data Analysis*, vol. 1. Wiley Classic Library, 2000.
- [64] J. Hoffswell, A. Satyanarayan, and J. Heer. Augmenting code with in situ visualizations to aid program understanding. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI)*, pp. 532:1–532:12. ACM, 2018. doi: 10.1145/3173574.3174106
- [65] A. Hogan, E. Blomqvist, M. Cochez, C. D’amato, G. D. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, and et al. Knowledge graphs. *ACM Computing Surveys*, 54(4):1–37, Jul 2021. doi: 10.1145/3447772
- [66] D. Howard and A. M. MacEachren. Interface design for geographic visualization: Tools for representing reliability. *Cartography and Geographic Information Systems*, 23(2):59–77, 1996. doi: 10.1559/152304096782562109
- [67] T.-H. Huang and M. L. Huang. Analysis and visualization of co-authorship networks for understanding academic collaboration and knowledge domain of individual researchers. In *Proceedings of the 2006 International Conference on Computer Graphics, Imaging and Visualisation*, pp. 18–23. IEEE, 2006. doi: 10.1109/CGIV.2006.20
- [68] J. Hullman, N. Diakopoulos, and E. Adar. Contextifier: automatic generation of annotated stock visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI*, pp. 2707–2716. ACM, 2013. doi: 10.1145/2470654.2481374
- [69] J. Hullman, S. Drucker, N. Henry Riche, B. Lee, D. Fisher, and E. Adar. A deeper understanding of sequence in narrative visualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2406–2415, 2013. doi: 10.1109/TVCG.2013.119
- [70] J. Hullman, R. Kosara, and H. Lam. Finding a clear path: Structuring strategies for visualization sequences. In *Computer Graphics Forum*, vol. 36, pp. 365–375, 2017. doi: doi.org/10.1111/cgf.13194
- [71] J. Hunter, A. Gatt, F. Portet, E. Reiter, and S. Sripada. Using natural language generation technology to improve information flows in intensive care units. In *ECAI*, pp. 678–682, 2008. doi: 10.3233/978-1-58603-891-5-678

- [72] C. Hurter, N. H. Riche, S. M. Drucker, M. Cordeil, R. Alligier, and R. Vuillemot. Fiberclay: Sculpting three dimensional trajectories to reveal structural insights. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):704–714, 2019.
- [73] P. Isenberg, B. Lee, H. Qu, and M. Cordeil. Immersive visual data stories. In *Immersive Analytics*, pp. 165–184. Springer, 2018.
- [74] A. Jain and J. M. Keller. Textual summarization of events leading to health alerts. In *Engineering in Medicine and Biology Society, 37th Annual International Conference of the IEEE, EMBC '15*, pp. 7634–7637, 2015. doi: 10.1109/EMBC.2015.7320160
- [75] J. Jo, F. Vernier, P. Dragicevic, and J.-D. Fekete. A declarative rendering model for multiclass density maps. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):470–480, 2019. doi: 10.1109/TVCG.2018.2865141
- [76] P. Jordan, N. L. Green, C. Thomas, and S. Holm. TBI-Doc: Generating patient & clinician reports from brain imaging data. In *Proceedings of the 8th International Natural Language Generation Conference*, pp. 143–146. Association for Computational Linguistics, 2014.
- [77] J.-C. Kalo, L. Fichtel, P. Ehler, and W.-T. Balke. KnowlyBERT-Hybrid query answering over language models and knowledge graphs. In *International Semantic Web Conference*, pp. 294–310. Springer, 2020.
- [78] A. Kapoor. Visdown. <https://visdown.com/>, 2016. Accessed: 2019-01-13.
- [79] R. Khulusi, J. Kusnick, J. Focht, and S. Jänicke. An interactive chart of biography. In *2019 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 257–266. IEEE, 2019.
- [80] D. H. Kim, E. Hoque, J. Kim, and M. Agrawala. Facilitating document reading by linking text and tables. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology (UIST)*, pp. 423–434. ACM, 2018. doi: 10.1145/3242587.3242617
- [81] N. W. Kim, N. Henry Riche, B. Bach, G. Xu, M. Brehmer, K. Hinckley, M. Pahud, H. Xia, M. J. McGuffin, and H. Pfister. Datatoon: Drawing dynamic network comics with pen+ touch interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 105. ACM, 2019.
- [82] N. W. Kim, H. Im, N. Henry Riche, A. Wang, K. Gajos, and H. Pfister. Dataselfie: Empowering people to design personalized visuals to represent their data. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, p. 79. ACM, 2019.
- [83] N. W. Kim, E. Schweickart, Z. Liu, M. Dontcheva, W. Li, J. Popovic, and H. Pfister. Data-driven guides: Supporting expressive design for information graphics. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):491–500, 2016.
- [84] S. Kim, R. Maciejewski, A. Malik, Y. Jang, D. S. Ebert, and T. Isenberg. Bristle Maps: A multivariate abstraction technique for geovisualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(9):1438–1454, 2013. doi: 10.1109/TVCG.2013.66
- [85] R. Kochhar. Middle class fortunes in western europe. <https://www.pewresearch.org/global/2017/04/24/middle-class-fortunes-in-western-europe/>. Accessed: 2020-08-10.
- [86] H.-K. Kong, Z. Liu, and K. Karahalios. Frames and slants in titles of visualizations on controversial topics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 438. ACM, 2018. doi: 10.1145/3173574.3174012

- [87] N. Kong, M. A. Hearst, and M. Agrawala. Extracting references between text and charts via crowdsourcing. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 31–40. ACM, 2014.
- [88] R. Kosara and J. Mackinlay. Storytelling: The next step for visualization. *Computer*, 46(5):44–50, 2013. doi: 10.1109/MC.2013.36
- [89] T. Kurosawa and Y. Takama. Co-authorship networks visualization system for supporting survey of researchers’ future activities. *Journal of Emerging Technologies in Web Intelligence*, 4(1):3–14, 2012. doi: 10.4304/jetwi.4.1.3-14
- [90] B. C. Kwon, F. Stoffel, D. Jäckle, B. Lee, and D. Keim. VisJockey: Enriching Data Stories through Orchestrated Interactive Visualization. In *Computation+Journalism Symposium 2014*, 2014.
- [91] S. Lallé, D. Toker, and C. Conati. Gaze-driven adaptive interventions for magazine-style narrative visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 2019.
- [92] S. Latif, S. Agarwal, S. Gottschalk, C. Chrosch, F. Feit, J. Jahn, T. Braun, Y. C. Tchenko, E. Demidova, and F. Beck. Visually connecting historical figures through event knowledge graphs. In *2021 IEEE Visualization Conference (VIS)*, pp. 156–160, 2021. doi: 10.1109/VIS49827.2021.9623313
- [93] S. Latif and F. Beck. Visually augmenting documents with data. *Computing in Science Engineering*, 20(6):96–103, 2018. doi: 10.1109/MCSE.2018.2875316
- [94] S. Latif and F. Beck. Interactive Map Reports summarizing bivariate geographic data. *Visual Informatics*, 3(1):27–37, 2019. doi: 10.1016/j.visinf.2019.03.004
- [95] S. Latif and F. Beck. VIS Author Profiles: Interactive descriptions of publication records combining text and visualization. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):152–161, Jan 2019. doi: 10.1109/TVCG.2018.2865022
- [96] S. Latif, S. Chen, and F. Beck. A deeper understanding of visualization-text interplay in geographic data-driven stories. *Computer Graphics Forum*, 40(3):311–322, 2021. doi: 10.1111/cgf.14309
- [97] S. Latif, D. Liu, and F. Beck. Exploring interactive linking between text and visualization. In *EuroVis 2018 - Short Papers*, pp. 91–94. The Eurographics Association, 2018. doi: 10.2312/eurovisshort.20181084
- [98] S. Latif, K. Su, and F. Beck. Authoring combined textual and visual descriptions of graph data. In *EuroVis 2019 - Short Papers*. The Eurographics Association, may 2019. doi: 10.2312/evs.20191180
- [99] S. Latif, H. Tarner, and F. Beck. Talking realities: Audio guides in virtual reality visualizations. *IEEE Computer Graphics and Applications*, pp. 1–1, 2021. doi: 10.1109/MCG.2021.3058129
- [100] S. Latif, Z. Zhou, Y. Kim, F. Beck, and N. W. Kim. Kori: Interactive synthesis of text and charts in data documents. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021. doi: 10.1109/TVCG.2021.3114802
- [101] B. Lavoie, O. Rambow, and E. Reiter. The ModelExplainer. In *Demonstration Notes of the International Natural Language Generation Workshop (INLG)*, pp. 9–12, 1996.
- [102] B. Lavoie, O. Rambow, and E. Reiter. Customizable descriptions of object-oriented models. In *Proceedings of the Conference on Applied Natural Language Processing (ANLC)*, pp. 253–256. Association for Computational Linguistics, 1997. doi: 10.3115/974557.974594

- [103] P. Leskinen, E. Hyvönen, and J. Tuominen. Analyzing and visualizing prosopographical linked data based on biographies. In *Proceedings of the Second Conference on Biographical Data in a Digital World 2017*, vol. 2119 of *CEUR Workshop Proceedings*, pp. 39–44, 2017.
- [104] A. Lex, N. Gehlenborg, H. Strobel, R. Vuillemot, and H. Pfister. Upset: Visualization of intersecting sets. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1983–1992, 2014. doi: 10.1109/TVCG.2014.2346248
- [105] Q. Liu, Y. Hu, L. Shi, X. Mu, Y. Zhang, and J. Tang. EgoNetCloud: Event-based egocentric dynamic network visualization. In *Proceedings of the 2015 IEEE Conference on Visual Analytics Science and Technology*, VAST, pp. 65–72. IEEE, 2015. doi: 10.1109/VAST.2015.7347632
- [106] Z. Liu, J. Thompson, A. Wilson, M. Dontcheva, J. Delorey, S. Grigg, B. Kerr, and J. Stasko. Data illustrator: Augmenting vector design tools with lazy data binding for expressive visualization authoring. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 123. ACM, 2018.
- [107] Z. Liu, J. Thompson, A. Wilson, M. Dontcheva, J. Delorey, S. Grigg, B. Kerr, and J. Stasko. Data Illustrator: Augmenting vector design tools with lazy data binding for expressive visualization authoring. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 123:1–123:13. ACM, 2018. doi: 10.1145/3173574.3173697
- [108] A. M. MacEachren. *How maps work: representation, visualization, and design*. Guilford Press, 2004.
- [109] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM Transactions On Graphics*, 5(2):110–141, 1986.
- [110] J. L. S. Malqui, N. M. L. Romero, R. Garcia, H. Alemdar, and J. L. Comba. How do soccer teams coordinate consecutive passes? a visual analytics system for analysing the complexity of passing sequences using soccer flow motifs. *Computers & Graphics*, 84:122–133, 2019. doi: 10.1016/j.cag.2019.08.010
- [111] J. Marian. Defect prediction: Xerxes, July 2010. doi: 10.5281/zenodo.268474
- [112] R. E. Mayer and L. Fiorella. 12 principles for reducing extraneous processing in multimedia learning: Coherence, signaling, redundancy, spatial contiguity, and temporal contiguity principles. In *The Cambridge handbook of multimedia learning*, vol. 279. Cambridge University Press, 2014.
- [113] P. W. McBurney and C. McMillan. Automatic documentation generation via source code summarization of method context. In *Proceedings of the International Conference on Program Comprehension (ICPC)*, pp. 279–290. ACM, 2014. doi: 10.1145/2597008.2597149
- [114] T. J. McCabe. A complexity measure. *IEEE Transactions on Software Engineering (TSE)*, SE-2(4):308–320, 1976. doi: 10.1109/TSE.1976.233837
- [115] S. McKenna, N. Henry Riche, B. Lee, J. Boy, and M. Meyer. Visual narrative flow: Exploring factors shaping data visualization story reading experiences. In *Computer Graphics Forum*, vol. 36, pp. 377–387. Wiley Online Library, 2017.
- [116] G. McNeill and S. A. Hale. Viz-Blocks: Building Visualizations and Documents in the Browser. In J. Johansson, F. Sadlo, and G. E. Marai, eds., *EuroVis 2019 - Short Papers*. The Eurographics Association, 2019. doi: 10.2312/evs.20191177
- [117] R. Metoyer, Q. Zhi, B. Janczuk, and W. Scheirer. Coupling story to visualization: Using textual analysis as a bridge between data and interpretation. In *23rd International Conference on Intelligent User Interfaces*, p. 503–507. ACM, 2018. doi: 10.1145/3172944.3173007

- [118] F. Meziane, N. Athanasakis, and S. Ananiadou. Generating natural language specifications from UML class diagrams. *Requirements Engineering (RE)*, 13(1):1–18, 2008. doi: 10.1007/s00766-007-0054-0
- [119] T. Mikolov, E. Grave, P. Bojanowski, C. Puhersch, and A. Joulin. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.
- [120] 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*, vol. 26. Curran Associates, Inc., 2013.
- [121] I. Milligan. Illusionary order: Online databases, optical character recognition, and Canadian history, 1997-2010. *Canadian Historical Review*, 94(4):540–569, 2013. doi: 10.3138/chr.694
- [122] V. O. Mittal, S. F. Roth, J. D. Moore, J. Mattis, and G. Carenini. Generating explanatory captions for information graphics. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI*, pp. 1276–1283. Morgan Kaufmann Publishers Inc., 1995.
- [123] M. Monmonier. Strategies for the visualization of geographic time-series data. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 27(1):30–45, 1990. doi: 10.3138/U558-H737-6577-8U31
- [124] L. Moreno, J. Aponte, G. Sridhara, A. Marcus, L. Pollock, and K. Vijay-Shanker. Automatic generation of natural language summaries for Java classes. In *Proceedings of the IEEE International Conference on Program Comprehension (ICPC)*, pp. 23–32. IEEE, 2013. doi: 10.1109/ICPC.2013.6613830
- [125] H. Mumtaz, S. Latif, F. Beck, and D. Weiskopf. Explorantive code quality documents. *IEEE Transactions on Visualization and Computer Graphics*, 26(1), 2020. doi: 10.1109/TVCG.2019.2934669
- [126] M. Nafari and C. Weaver. Query2question: translating visualization interaction into natural language. *IEEE transactions on visualization and computer graphics*, 21(6):756–769, 2015. doi: 10.1109/TVCG.2015.2396062
- [127] E. Nelson. The impact of bivariate symbol design on task performance in a map setting. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 37(4):61–78, 2000. doi: 10.3138/V743-K505-5510-66Q5
- [128] P. H. Nguyen, K. Xu, R. Walker, and B. W. Wong. TimeSets: Timeline visualization with set relations. *Information Visualization*, 15(3):253–269, 2016.
- [129] C. North. Toward measuring visualization insight. *IEEE Computer Graphics and Applications*, 26(3):6–9, 2006. doi: 10.1109/MCG.2006.70
- [130] P. Nowakowski, E. Ciepiela, D. Hareźlak, J. Kocot, M. Kasztelnik, T. Bartyński, J. Meizner, G. Dyk, and M. Malawski. The collage authoring environment. *Procedia Computer Science*, 4:608–617, 2011.
- [131] A. Ottley, A. Kaszowska, R. J. Crouser, and E. M. Peck. The Curious Case of Combining Text and Visualization. In *EuroVis 2019 - Short Papers*. The Eurographics Association, 2019. doi: 10.2312/evs.20191181
- [132] A. Ouni, M. Kessentini, H. Sahraoui, and M. Boukadoum. Maintainability defects detection and correction: A multi-objective approach. *Automated Software Engineering (ASE)*, 20(1):47–79, 2013. doi: 10.1007/s10515-011-0098-8

- [133] J. Priestley. *A Chart of Biography*. 1765.
- [134] R. Puduppully, L. Dong, and M. Lapata. Data-to-text generation with content selection and planning. In *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, pp. 6908–6915, 2019. doi: 10.1609/aaai.v33i01.33016908
- [135] A. Ramos-Soto, A. J. Bugarin, S. Barro, and J. Taboada. Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data. *IEEE Transactions on Fuzzy Systems*, 23(1):44–57, 2015. doi: 10.1109/TFUZZ.2014.2328011
- [136] A. Ramos-Soto, B. Vazquez-Barreiros, A. Bugarín, A. Gewerc, and S. Barro. Evaluation of a data-to-text system for verbalizing a learning analytics dashboard. *International Journal of Intelligent Systems*, 32(2):177–193, 2017. doi: 10.1002/int.21835
- [137] E. Reiter. Nlg vs. templates. In *Proceedings of the 5th European Workshop on Natural Language Generation (EWNLG'95)*, pp. 95–106. Leiden, 1995.
- [138] E. Reiter, R. Dale, and Z. Feng. *Building natural language generation systems*. MIT Press, 2000.
- [139] F. Reitz. A framework for an ego-centered and time-aware visualization of relations in arbitrary data repositories. *arXiv preprint arXiv:1009.5183*, 2010.
- [140] D. Ren, M. Brehmer, B. Lee, T. Höllerer, and E. K. Choe. ChartAccent: Annotation for data-driven storytelling. In *2017 IEEE Pacific Visualization Symposium*, pp. 230–239, 2017. doi: 10.1109/PACIFICVIS.2017.8031599
- [141] D. Ren, T. Höllerer, and X. Yuan. ivisdesigner: Expressive interactive design of information visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2092–2101, 2014.
- [142] D. Ren, B. Lee, and M. Brehmer. Charticator: Interactive construction of bespoke chart layouts. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):789–799, 2018.
- [143] N. H. Riche, C. Hurter, N. Diakopoulos, and S. Carpendale. *Data-driven storytelling*. CRC Press, 2018.
- [144] P. J. Rousseeuw, I. Ruts, and J. W. Tukey. The bagplot: A bivariate boxplot. *The American Statistician*, 53(4):382–387, 1999. doi: 10.1080/00031305.1999.10474494
- [145] P. Saraiya, C. North, and K. Duca. An insight-based methodology for evaluating bioinformatics visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):443–456, 2005. doi: 10.1109/TVCG.2005.53
- [146] A. Satyanarayan and J. Heer. Authoring narrative visualizations with Ellipsis. In *Computer Graphics Forum*, vol. 33, pp. 361–370. Wiley Online Library, 2014. doi: 10.1111/cgf.12392
- [147] A. Satyanarayan and J. Heer. Lyra: An interactive visualization design environment. In *Computer Graphics Forum*, vol. 33, pp. 351–360. Wiley Online Library, 2014.
- [148] A. Satyanarayan, D. Moritz, K. Wongsuphasawat, and J. Heer. Vega-lite: A grammar of interactive graphics. *IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis)*, 2017.
- [149] A. Satyanarayan, R. Russell, J. Hoffswell, and J. Heer. Reactive vega: A streaming dataflow architecture for declarative interactive visualization. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):659–668, 2016.

- [150] C. Schulz, N. Rodrigues, K. Damarla, A. Henicke, and D. Weiskopf. Visual exploration of mainframe workloads. In *Proceedings of the SIGGRAPH Asia Symposium on Visualization (SA17VIS)*, pp. 4:1–4:7. ACM, 2017. doi: 10.1145/3139295.3139312
- [151] E. Segel and J. Heer. Narrative visualization: Telling stories with data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1139–1148, 2010. doi: 10.1109/TVCG.2010.179
- [152] V. Setlur, S. E. Battersby, M. Tory, R. Gossweiler, and A. X. Chang. Eviza: A natural language interface for visual analysis. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pp. 365–377, 2016.
- [153] D. Shi, X. Xu, F. Sun, Y. Shi, and N. Cao. Calliope: Automatic visual data story generation from a spreadsheet. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):453–463, 2021. doi: 10.1109/TVCG.2020.3030403
- [154] L. Shi, C. Wang, Z. Wen, H. Qu, C. Lin, and Q. Liao. 1.5D egocentric dynamic network visualization. *IEEE Transactions on Visualization and Computer Graphics*, 21:624–637, 2015. doi: 10.1109/TVCG.2014.2383380
- [155] T. Shibata, J. Kim, D. M. Hoffman, and M. S. Banks. The zone of comfort: Predicting visual discomfort with stereo displays. *Journal of Vision*, 11(8):11–11, 2011. doi: 10.1167/11.8.11
- [156] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *The Craft of Information Visualization*, pp. 364–371. Elsevier, 2003. doi: 10.1016/B978-155860915-0/50046-9
- [157] R. Sicat, J. Li, J. Choi, M. Cordeil, W. Jeong, B. Bach, and H. Pfister. DXR: A toolkit for building immersive data visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):715–725, 2019. doi: 10.1109/TVCG.2018.2865152
- [158] T. Souza Costa, S. Gottschalk, and E. Demidova. Event-QA: A dataset for event-centric question answering over knowledge graphs. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 3157–3164, 2020.
- [159] G. Sridhara, E. Hill, D. Muppaneni, L. Pollock, and K. Vijay-Shanker. Towards automatically generating summary comments for Java methods. In *Proceedings of the IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 43–52. ACM, 2010. doi: 10.1145/1858996.1859006
- [160] A. Srinivasan, S. M. Drucker, A. Endert, and J. Stasko. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 25(1):672–681, 2019. doi: 10.1109/TVCG.2018.2865145
- [161] A. Srinivasan, S. M. Drucker, A. Endert, and J. Stasko. Augmenting visualizations with interactive data facts to facilitate interpretation and communication. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):672–681, 2019. doi: 10.1109/TVCG.2018.2865145
- [162] S. G. Sripada and F. Gao. Summarizing dive computer data: A case study in integrating textual and graphical presentations of numerical data. In *Workshop on Multimodal Output Generation*, MOG ’07, p. 149, 2007.
- [163] M. Steinberger, M. Waldner, M. Streit, A. Lex, and D. Schmalstieg. Context-preserving visual links. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2249–2258, 2011.

- [164] C. Stolte, D. Tang, and P. Hanrahan. Polaris: A system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):52–65, 2002.
- [165] M. Sulír, M. Bačíková, S. Chodarev, and J. Porubán. Visual augmentation of source code editors: A systematic mapping study. *Journal of Visual Languages & Computing (JVLC)*, 2018. doi: 10.1016/j.jvlc.2018.10.001
- [166] J. Sweller. Implications of cognitive load theory for multimedia learning. In R. E. Mayer, ed., *The Cambridge Handbook of Multimedia Learning*, pp. 19–30. Cambridge University Press, 2005.
- [167] J. Sweller, P. Ayres, and S. Kalyuga. The split-attention effect. In *Cognitive Load Theory*, pp. 111–128. Springer, 2011.
- [168] J. Sweller, J. van Merriënboer, and F. Paas. Cognitive architecture and instructional design. *Educational Psychology Review*, 10(3):251–296, 1998.
- [169] B. Swift, A. Sorensen, H. Gardner, and J. Hosking. Visual code annotations for cyberphysical programming. In *Proceedings of the International Workshop on Live Programming (LIVE)*, pp. 27–30. IEEE, 2013. doi: 10.1109/LIVE.2013.6617345
- [170] K. Thomas and S. Sripada. Atlas.txt: Linking geo-referenced data to text for NLG. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, ENLG ’07, pp. 163–166, 2007.
- [171] D. Toker, C. Conati, and G. Carenini. User-adaptive support for processing magazine style narrative visualizations: Identifying user characteristics that matter. In *23rd International Conference on Intelligent User Interfaces*, pp. 199–204. ACM, 2018. doi: 10.1145/3172944.3173009
- [172] M. Tory and V. Setlur. Do what i mean, not what i say! design considerations for supporting intent and context in analytical conversation. In *2019 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 93–103, 2019. doi: 10.1109/VAST47406.2019.8986918
- [173] E. R. Tufte. *Beautiful Evidence*. Graphics Press, 1st ed., 2006.
- [174] J. W. Tukey. *Exploratory Data Analysis*, vol. 2. Reading, Mass., 1977.
- [175] R. Turner, S. Sripada, E. Reiter, and I. P. Davy. Using spatial reference frames to generate grounded textual summaries of georeferenced data. In *Proceedings of the Fifth International Natural Language Generation Conference*, INLG ’08, pp. 16–24. Association for Computational Linguistics, Stroudsburg, PA, USA, 2008. doi: 10.3115/1708322.1708328
- [176] K. van Deemter, E. Krahmer, and M. Theune. Plan-based vs. template-based nlg: a false opposition? *Becker and Busemann (1999)*, pp. 1–5, 1999.
- [177] B. Victor. Explorable explanations. <http://worrydream.com/ExplorableExplanations>, 2011. Accessed: 2019-03-28.
- [178] W. Wahlster, E. André, W. Finkler, H.-J. Profitlich, and T. Rist. Plan-based integration of natural language and graphics generation. *Artificial Intelligence*, 63(1-2):387–427, 1993. doi: 10.1016/0004-3702(93)90022-4
- [179] Y. Wang, Z. Sun, H. Zhang, W. Cui, K. Xu, X. Ma, and D. Zhang. Datashot: Automatic generation of fact sheets from tabular data. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):895–905, 2019. doi: 10.1109/TVCG.2019.2934398

- [180] D. L. Wanzer, T. Azzam, N. D. Jones, and D. Skousen. The role of titles in enhancing data visualization. *Evaluation and Program Planning*, 84:101896, 2021. doi: 10.1016/j.evalprogplan.2020.101896
- [181] D. Weiskopf, M. Borchers, T. Ertl, M. Falk, O. Fechtig, R. Frank, F. Grave, A. King, U. Kraus, T. Müller, H. Nollert, I. R. Mendez, H. Ruder, T. Schafhitzel, S. Schär, C. Zahn, and M. Zatloukal. Explanatory and illustrative visualization of special and general relativity. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 12(4):522–534, 2006. doi: 10.1109/TVCG.2006.69
- [182] R. Wille. Restructuring lattice theory: an approach based on hierarchies of concepts. In *Ordered Sets*, pp. 445–470. Springer, 1982. doi: 10.1007/978-94-009-7798-3_15
- [183] S. Wiseman, S. Shieber, and A. Rush. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2253–2263, 2017. doi: 10.18653/v1/D17-1239
- [184] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE transactions on visualization and computer graphics*, 22(1):649–658, 2015.
- [185] Y. Wu, N. Pitipornvivat, J. Zhao, S. Yang, G. Huang, and H. Qu. egoSlider: Visual analysis of egocentric network evolution. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):260–269, 2016. doi: 10.1109/TVCG.2015.2468151
- [186] H. Xia, N. Henry Riche, F. Chevalier, B. De Araujo, and D. Wigdor. Dataink: Direct and creative data-oriented drawing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 223. ACM, 2018.
- [187] Y. Yang, J. Cao, Y. Wen, and P. Zhang. Table to text generation with accurate content copying. *Scientific reports*, 11(1):1–12, 2021. doi: 10.1038/s41598-021-00813-6
- [188] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI, pp. 401–408. ACM, 2003. doi: 10.1145/642611.642681
- [189] A. Ynnerman, J. Löwgren, and L. Tibell. Explorantion: A new science communication paradigm. *IEEE Computer Graphics and Applications (CG&A)*, 38(3):13–20, 2018. doi: 10.1109/MCG.2018.032421649
- [190] B. Yu and C. T. Silva. Flowsense: A natural language interface for visual data exploration within a dataflow system. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1–11, 2019.
- [191] B. Yu and C. T. Silva. FlowSense: A natural language interface for visual data exploration within a dataflow system. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1–11, 2020. doi: 10.1109/TVCG.2019.2934668
- [192] M. Zhang, T. Hall, and N. Baddoo. Code bad smells: A review of current knowledge. *Journal of Software Maintenance and Evolution: Research and Practice (JSME)*, 23(3):179–202, 2011. doi: 10.1002/smr.521
- [193] J. Zhao, C. Collins, F. Chevalier, and R. Balakrishnan. Interactive exploration of implicit and explicit relations in faceted datasets. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2080–2089, 2013. doi: 10.1109/TVCG.2013.167

- [194] Q. Zhi, A. Ottley, and R. Metoyer. Linking and layout: Exploring the integration of text and visualization in storytelling. In *Computer Graphics Forum*, vol. 38, pp. 675–685, 2019. doi: 10.1111/cgf.13719

Eidesstattliche Erklärung zu § 14 Abs. 1 Nr. 6

Ich gebe folgende eidesstattliche Erklärung ab:

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig ohne unzulässige Hilfe Dritter verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und alle wörtlich oder inhaltlich übernommenen Stellen unter der Angabe der Quelle als solche gekennzeichnet habe.

Die Grundsätze für die Sicherung guter wissenschaftlicher Praxis an der Universität Duisburg-Essen sind beachtet worden.

Ich habe die Arbeit keiner anderen Stelle zu Prüfungszwecken vorgelegt.

Ort, Datum

Unterschrift.



This thesis was typeset using \LaTeX , originally developed by Leslie Lamport and based on Donald Knuth's \TeX . The body text is set in 11 point Egenolff-Berner Garamond, a revival of Claude Garamont's humanist typeface. The above illustration, "Science Experiment 02", was created by Ben Schlitter and released under [CC BY-NC-ND 3.0](#). A template that can be used to format a PhD thesis with this look and feel has been released under the permissive [MIT \(X11\)](#) license, and can be found online at github.com/suchow/Dissertate or from its author, Jordan Suchow, at suchow@post.harvard.edu.