Visual Analytics of Social Media for Situation Awareness

Von der Fakultät Informatik, Elektrotechnik und
Informationstechnik der Universität Stuttgart
zur Erlangung der Würde eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigte Abhandlung

Vorgelegt von

Dennis Benjamin Thom

aus Esslingen am Neckar

Hauptberichter:	Prof. Dr. Thomas Ertl
Mitberichter:	Prof. Dr. David Ebert

Tag der mündlichen Prüfung: 06.03.2015

Institut für Visualisierung und Interaktive Systeme
der Universität Stuttgart

2015

Acknowledgments

Every thesis is a journey, and I would like to thank all the people that accompanied and supported me during mine. I thank my advisor Thomas Ertl for the opportunity to work on a highly interesting topic, for counseling me through all the ups and downs of the challenge, and for creating a unique and inspiring work environment at the VIS/VISUS institute. I thank David Ebert for his ongoing interest in my work, for the multiple opportunities to visit him in West Lafayette, and for the collaboration with his great students at Purdue University.

During my thesis I had the chance of working together with many exceptional people. Most notably, I thank all members of VIS/VISUS for being such a bunch of lovely, crazy, and out-of-the-box-thinking characters. I thank my friends and co-authors Harald Bosch, Steffen Koch, Robert Krüger, Michael Wörner, and Florian Heimerl for sharing all the laughs, tears, and midnight coding hours. Their support and the successful cooperation formed the very basis of this thesis. Additionally, I thank Harald, Steffen, and Robert for proof-reading the initial draft. For our fruitful cooperations, I thank all the colleagues and collaborators that I met over the course of the research projects PESCaDO, VASA, and VACCINE.

My deepest gratitude goes to my parents Gudrun and Klaus, and to my partner Sandra for their love and support in all those years. Without them, I would probably have given up long ago. Finally, I would like to express a special thank you to my grandmother, who has always been a good friend, adviser, and my biggest fan. I hope you can see this, wherever you are right now.

Dennis Thom
University of Stuttgart
March 2015

Contents

Acknowledgments **iii**

Abstract **xiii**

German Abstract — Zusammenfassung **xv**

1 Introduction **1**
1.1 Motivation . . . 3
1.2 Contribution . . . 4
1.3 Overview . . . 6

2 Foundations and Approach **9**
2.1 Visual Analytics . . . 10
2.1.1 Abstract Data Visualization . . . 11
2.1.2 Definition and Model . . . 12
2.1.3 Common Characteristics . . . 14
2.1.4 Data Mining Components . . . 18
2.2 Harvesting Information from Social Media . . . 21
2.2.1 Social Media Microdocuments . . . 23
2.2.2 Social Media as a Data Source . . . 25
2.2.3 Twitter API . . . 29
2.3 Situation Awareness . . . 30
2.3.1 Definition and Model . . . 31
2.3.2 Leveraging Social Media . . . 33
2.3.3 Future Requirements . . . 34
2.4 The Social Media Analytics Model . . . 36
2.4.1 Addressing the Open Challenges . . . 37
2.4.2 Model . . . 39
2.4.3 State of the Art . . . 41

3 Query Optimization **45**
3.1 Microdocument Retrieval Challenges . . . 46
3.2 The *TreeQueST* Exploration Approach . . . 48
3.2.1 User Interface and Exploration Process . . . 49
3.2.2 Query Creation and Evaluation . . . 50
3.3 Background . . . 51
3.3.1 Evolution of Scatter/Gather . . . 51
3.3.2 Visual Information Spaces . . . 53

3.3.3 Hierarchical Information Retrieval . . . 53
3.4 Treemap Cluster Visualization . . . 54
3.4.1 Agglomerative Clustering . . . 54
3.4.2 Message Similarity . . . 55
3.4.3 Visual Tree Representation . . . 57
3.4.4 Tag Clouds, Spatial Layout and Exploration Lens . . . 59
3.5 Automated Query Construction . . . 61
3.5.1 Algorithm . . . 61
3.5.2 Evaluation and Manipulation . . . 62
3.6 Case Study . . . 63

4 Visual Event Discovery **69**
4.1 Spatiotemporal Anomalies . . . 71
4.2 Background . . . 74
4.2.1 Visual Analytics of Spatiotemporal Data . . . 75
4.2.2 Social Media Event Discovery . . . 76
4.2.3 Term Relevance Metrics . . . 77
4.3 *TagMap* Event Discovery . . . 78
4.3.1 Procedure Overview . . . 79
4.3.2 Stream-enabled Cluster Analysis . . . 81
4.3.3 Adaptive Visualization . . . 86
4.3.4 Case Studies . . . 88
4.4 Term Relevance Normalization . . . 93
4.4.1 Geo-aware *tf-idf* . . . 94
4.4.2 Scalable Implementation Strategies . . . 97
4.4.3 Measure Performance . . . 102
4.4.4 *TagMap* Integration . . . 105

5 Task-Adaptive Detection and Drill-Down **109**
5.1 Background . . . 111
5.2 Visual Active Learning . . . 112
5.3 Classifier Orchestration . . . 115
5.3.1 Interactive Filter Management . . . 116
5.3.2 Tasks and Capabilities . . . 118
5.4 Monitoring Workflow: Doing by Learning . . . 118
5.5 Case Study . . . 120

6 The *ScatterBlogs* Platform **125**
6.1 Data Structures and Index . . . 127
6.2 System Architecture . . . 128
6.3 User Interface . . . 131

6.3.1 Basic Exploration Tools . . . 132
6.4 Implementation of the Analytics Cycle . . . 134
6.4.1 Step 1: Query Optimization and Retrieval . . . 134
6.4.2 Step 2: Overview and Indication . . . 135
6.4.3 Step 3: Task-adaptive Filters . . . 136
6.4.4 Closing the Loop . . . 137

7 Evaluation **139**
7.1 Evaluating Visual Analytics . . . 140
7.1.1 Evaluation Design and Benchmarks . . . 140
7.1.2 Practical Considerations . . . 142
7.2 *ScatterBlogs* Domain Expert Study . . . 143
7.2.1 Experimental Setup . . . 143
7.2.2 Findings and Comments . . . 149
7.2.3 Survey . . . 154
7.3 *TreeQueST* User Study . . . 156
7.3.1 Experimental Setup . . . 157
7.3.2 Findings and Comments . . . 158
7.3.3 Survey . . . 159

8 Conclusion and Outlook **163**
8.1 Summary of Contributions . . . 163
8.2 Lessons Learned . . . 166
8.2.1 Two Sides of the Medal . . . 166
8.2.2 The Latent Challenge . . . 167
8.3 Generalization . . . 168
8.4 Future Work . . . 170

Bibliography **173**

List of Figures

Chapter 1

1.1 Twitter message of US airways 2009 incident 3

Chapter 2

2.1 The information visualization pipeline 11
2.2 Visual analytics model 14
2.3 Overview of social media services 22
2.4 Twitter usage during the 2011 Virginia Earthquake 28
2.5 Situation awareness model 31
2.6 The social media analytics model 40
2.7 Number of Twitter-related papers by year 42

Chapter 3

3.1 Overview of the *TreeQueST* UI 48
3.2 Exploration loop of *TreeQueST* 50
3.3 Scatter/Gather as presented by Pirolli et al. 52
3.4 Dendrogram based on 530 tweets 58
3.5 Finding the distance transition with the L-Method 59
3.6 *TreeQueST* spatialization of messages 60
3.7 Initial overview in the Oculus VR case study 64
3.8 *TreeQueST* usage example: Sentiment analysis 65
3.9 *TreeQueST* usage example: Topic hierarchy 66

Chapter 4

4.1 Retrieval of messages during the VAST Challenge 2011 72
4.2 Traffic incident cluster in the VAST Challenge 2011 73
4.3 Visual cluster discovery in the VAST Challenge 2011 74
4.4 Activities to generate the *TagMap* 80
4.5 Simulation of stream clustering 85
4.6 *TagMap*: Label aggregation and zoom levels 87
4.7 *TagMap* analysis of the Virginia Earthquake 2011 90
4.8 *TagMap* analysis of the London Riots 2011 91
4.9 *TagMap* analysis of Hurricane Irene 92
4.10 Density adaptive quadtree-grid 98

4.11 The *idd* splatting process 99
4.12 Adaptive grid splatting 101
4.13 *idd* evaluation: Comic-Con 2012 105
4.14 *TagMap* enhanced by *idd* evaluation 106
4.15 Different events shown with *idd* highlighting 107

Chapter 5

5.1 Visual active learning UI 114
5.2 Filter orchestration tool 116
5.3 Integration of classifier training and monitoring 119
5.4 Classifier case study: First flood tweet 121
5.5 Classifier case study: Three precision levels 122
5.6 Classifier case study: Combination of classifiers 123
5.7 Classifier case study: Final overview 124

Chapter 6

6.1 *ScatterBlogs* system architecture 129
6.2 *ScatterBlogs* user interface 131
6.3 *ContentLens* technique 133
6.4 Integration of *TreeQueST* 135

Chapter 7

7.1 *TagMap* applied to the 2013 German Floods 145
7.2 *ContentLens* analysis of public riots 149
7.3 Usefulness ratings for *ScatterBlogs* 150
7.4 Results of the usability questionnaire 155
7.5 The plaintext search tool 157
7.6 Snapshot from the *TreeQueST* study 158
7.7 Likert scores for *TreeQueST* search 160
7.8 Likert scores for *TreeQueST* features 161

List of Abbreviations and Acronyms

AI	Artificial Intelligence
AL	Active Learning
API	Application Programmer Interface
DHS	Department of Homeland Security
HCI	Human Computer Interaction
IEEE	Institute of Electrical and Electronics Engineers
InfoVis	Information Visualization
IR	Information Retrieval
KDE	Kernel Density Estimation
LDA	Latent Dirichlet allocation
MCV	Multiple Coordinated Views
MDS	Multidimensional Scaling
NLP	Natural Language Processing
PCA	Principal Component Analysis
SA	Situation Awareness
SVM	Support Vector Machine
UI	User Interface
VA	Visual Analytics
VAST	IEEE Conference on Visual Analytics Science and Technology

Abstract

With the emergence of social media services and other user-centered web platforms the nature of the modern internet changed substantially. While it has since been a vast source of information and news on all kinds of topics, it recently grew into a continuous stream of knowledge, observations, thoughts, and situation reports. They are provided in real-time by millions of people from all over the world. This change also offers completely new possibilities for domains that rely on good situation awareness, such as disaster management, emergency response, disease control, and several forms of command and control environments. Analysts can find eyewitness videos of ongoing critical events in Youtube, they can observe the movement and communication behavior of Facebook users during evacuation measures, and they are enabled to trace the outspread of an epidemic disease just by highlighting symptom related keyword usage in Twitter.

However, the data sizes that need to be processed in order to identify relevant entries, produce comprehensible overviews, and detect anomalous patterns pose one of the most challenging analytics problems of our time. Not only the *volume* of data generated on a daily basis is larger than any other single database from the pre-internet era. The data is furthermore streamed in real-time at substantial *velocity*; it comes in a great *variety*, including text snippets, images, videos and network information; and it contains inaccuracies, misleading information, rumors, and fake meta-data, leading to uncertain *veracity*. In contrast to most other computer science challenges, social media analytics thus fully covers all characteristics that have been commonly referred to as the "four V's" of big data.

By tightly integrating approaches from the areas of data mining, information retrieval, natural language processing, human computer interaction, and data visualization the emerging field of visual analytics has been devised to tackle these challenges. As a descendant of the more general field of information visualization, visual analytics strives to merge the strengths of highly interactive visual interfaces with the computational power of automatic statistical algorithms. The goal of this combination is to advance problem solving in areas where a human analyst alone would be overwhelmed by the data volumes, while, at the same time, sheer processing power alone would not enable analysts to identify underlying patterns and relate information to semantic knowledge.

This thesis identifies four visual analytics requirements that have to be addressed to allow comprehensive situation awareness based on social media: Access to

data, visualization of context, coping with semantic complexity, and scalable processing. Based on core ideas of visual analytics, this work contributes three distinct techniques that allow to tackle access, context, and complexity, as well as a prototypical implementation that integrates all of them and allows scalable processing of the data. Means of iterative query optimization and hierarchical exploration of data samples are presented that allow to cope with the problem of rate limited web data collection. The challenge of relating information to space, time, and context is solved by a novel technique that automatically detects and visually highlights possibly relevant events. Here, a sophisticated language model based on large volumes of data is employed to separate meaningful and related information from signal noise. Finally, the possibility to drill-down into complex topics and to enable ongoing situation monitoring is achieved by means of interactive classifier training and orchestration.

The thesis furthermore presents an overarching analytics model, which integrates all solutions and relates their distinct capabilities. The techniques, their prototypical implementation, as well as the overarching analytics model are thoroughly evaluated, and they are compared with other approaches in context of the relatively young scientific discourse.

Along these lines, it is demonstrated how the aspects of user-driven detection and data-driven discovery distinctly align with supervised and unsupervised methods in machine learning. From the lessons learned, it is conclusively shown that visual configuration and steering of supervised classification on one hand, and the enhancement of visual interfaces through unsupervised clustering on the other hand, are two complementary concepts embedded at the very heart of visual analytics. The presented overarching analytics model might help to further enhance previous definition approaches and ostensive conceptions existing in the field.

German Abstract
—Zusammenfassung—

Mit dem Aufkommen der sozialen Medien und anderer nutzer-zentrierter Web-Plattformen hat sich die Natur des modernen Internets entscheidend verändert. Obschon es seit jeher eine gewaltige Quelle von Informationen und Neuigkeiten zu verschiedensten Themen war, ist es in jüngster Zeit zu einem unaufhörlichen Strom von Wissen, Beobachtungen, Gedanken, und persönlichen Statusberichten angewachsen. Diese Informationen werden in Echtzeit von Millionen von Nutzern aus der ganzen Welt bereitgestellt. Gleichzeitig bedeutet die Veränderung auch völlig neue Möglichkeiten für Anwendungsdomänen, in denen das sogenannte Situationsbewusstsein eine entscheidende Rolle spielt, sowie etwa dem Katastropheschutz, der Notfallrettung, der Seuchenkontrolle und vielen anderen Umgebungen mit Leit- und Kommandoständen. Datenanalysten können nun zeitnah Videos von Augenzeugen in Youtube finden, sie können das Bewegungs- und Kommunikationsverhalten von Facebooknutzern während Evakuierungsmaßnahmen beobachten oder sie können die Verbreitung einer Infektionskrankheit nachzeichnen, indem Erwähnungen von Symptomen in Twitter aufgezeigt werden.

Die Datenmengen die verarbeitet werden müssen, um relevante Einträge zu finden, umfassende Übersichten zu erzeugen und abnormale Muster zu erkennen, bedeuten jedoch einige der größten informatischen Herausforderungen unserer Zeit. Nicht nur ist der Umfang (volume) der täglich erzeugten Inhalte größer als jede einzelne Datenbank, welche vor dem Internet-Zeitalter entstanden ist. Die Daten werden darüber hinaus mit gewaltigen Durchsätzen (velocity) in Echtzeit übertragen; sie weisen eine erhebliche inhaltliche und strukturelle Vielfalt (variety) auf; und sie sind oft mit Ungenauigkeiten, irreführenden Hinweisen, Gerüchten und gefälschten Informationen versehen, was zu Problemen mit unklarer Vertrauenswürdigkeit (veracity) führt. Im Gegensatz zu den meisten anderen informatischen Herausforderungen treffen daher auf Daten aus sozialen Medien alle charakteristischen Eigenschaften zu, welche gemeinhin als die "vier V's" von Big Data bezeichnet werden.

Das aufkeimende Forschungsfeld der Visual Analytics wurde geschaffen, um genau diese Art von Problemen zu lösen. Dazu werden Ansätze aus den Bereichen Data Mining, Information Retrieval, Computerlinguistik, Mensch-Maschine-Interaktion und Datenvisualisierung miteinander verbunden. Als ein Teilgebiet des allgemeineren Feldes der Informationsvisualisierung versucht Visual Analytics die Stärken von hochinteraktiven visuellen Schnittstellen mit den

Möglichkeiten automatischer statistischer Verfahren zu vereinen. Das Ziel dieser Verbindung besteht darin, Problemlösungen in Bereichen zu entwickeln, in denen ein menschlicher Analyst von der Datenfülle überwältigt wäre, während reine Rechenkraft nicht ausreichen würde, um subtile Muster zu identifizieren und Informationen mit Kontextwissen in Bezug zu setzen.

Diese Arbeit identifiziert vier Anforderungen, welche berücksichtigt werden müssen, um eine umfassende Situationseinschätzung basierend auf Daten aus sozialen Medien zu ermöglichen. Diese umfassen die Erfassung der Daten, die visuelle Kontextualisierung, die Bewältigung semantischer Komplexität der Inhalte und ihre skalierbare Verarbeitung. Basierend auf zentralen Ansätzen der Visual Analytics stellt die Arbeit drei Methoden bereit, welche es erlauben die Probleme der Erfassung, Kontextualisierung und Komplexität zu bewältigen. Darüber hinaus wird eine prototypische Implementierung vorgestellt, welche die Lösungen integriert und die skalierbare Verarbeitung der Daten sicherstellt. Zur Anforderung der Datenerfassung wird ein Verfahren erläutert, welches die Erstellung und iterative Verbesserung von Suchanfragen basierend auf hierarchischer Exploration ermöglicht. Auf diese Weise kann dem Problem von Anfrage- und Volumen-limitierten Web-Schnittstellen begegnet werden. Die Anforderung, Informationen in einen zeitlichen, räumlichen und inhaltlichen Kontext zu setzen, wird von einer neuartigen Technik erfüllt, welche automatisch Ereignisse erkennt und übersichtlich visualisiert. Zu diesem Zweck kommt weiterhin ein hochentwickeltes statistisches Sprachmodell zum Einsatz, welches es erlaubt, aussagekräftige und zusammengehörige Information von Hintergrundrauschen zu trennen. Die Fähigkeit zu tiefergehender Untersuchung komplexer Inhalts- und Verweisstrukturen wird schließlich durch Verfahren interaktiver maschineller Lernverfahren und der visuellen Orchestrierung der daraus entstehenden Modelle ermöglicht.

Um die Verfahren miteinander zu verbinden, wird ein übergreifendes analytisches Modell vorgestellt, welches ihre komplementären Eigenschaften zueinander in Bezug setzt. Die vorgestellten Methoden, ihre prototypische Implementierung sowie das übergreifende analytische Modell wurden im Rahmen der Arbeit umfassend evaluiert und werden mit anderen Ansätzen im Kontext des noch jungen wissenschaftlichen Diskurses verglichen.

Im Rahmen der Arbeit wird weiterhin erörtert, dass die Aspekte benutzergesteuerter Erkennung zum einen und datengetriebener Entdeckung zum anderen in der Informationsvisualisierung eine naheliegende Verwandschaft zu überwachten und unüberwachten Verfahren im Bereich maschinellen Lernens aufweisen. Basierend auf den Erfahrungen dieser Arbeit wird veranschaulicht, dass die visuelle Konfiguration und Steuerung überwachter Klassifikationsverfahren und die Erweiterung visueller Schnittstellen durch unüberwachte Clustering-

verfahren zwei komplementäre Konzepte sind, welche bereits in der Natur des Visual Analytics Ansatzes zu finden sind. Das Schema des übergreifenden, analytischen Modells könnte daher helfen, bestehende Auffassungen und ostensive Definitionsansätze des Forschungsfeldes zu erweitern.

CHAPTER 1

Introduction

"It's after 2001. Where is HAL?" This question was asked 2007 by cognitive science pioneer Marvin Minsky in a talk about the state of artificial intelligence (AI). He was referring to the Stanley Kubrick movie *2001: A Space Odyssey*, where mankind has created the sentient and highly intelligent computer HAL 9000, which can solve complex problems, operate a spacecraft, and communicate with the crew on a human interaction level. With his statement, Minsky wanted to highlight that - in contrast to the hopes of enthusiastic researchers during the golden age of AI - no such thinking machine existed at that time.

Researchers realized early on that the computing power of machines allowed them to outperform human competitors in many tasks. Impressive examples of such capabilities have been shown. To this day, computers have beaten world champions in chess, proven mathematical theories, and won against human competitors in the game show Jeopardy. However, with the maturing of AI, scientists also realized that certain human capabilities are not so easily replicated by machines. They particularly include the ability to semantically relate information, discover hidden patterns based on experience, come up with creative solutions, and employ intuition to make decisions based on incomplete information. In 2014, machines with such capabilities are still subject in science fiction, and it might take decades if not centuries until significant progress in this direction is made.

Instead of replacing human capabilities by fully automated algorithms, the field of *visual analytics* (VA) has emerged as the vision of bringing the best of

both worlds together in order to solve big challenges in *our time*. For one thing, artificial intelligence researchers have employed the computing power of machines to devise fully automated problem solutions. This particularly comprises approaches from the domains of data mining, natural language processing, and pattern recognition, which can all be seen as building blocks in creating intelligent systems. For another thing, data visualization experts have since leveraged the capabilities of human analysts by representing large and complex information in a comprehensible form. Here, methods of data exploration, filtering, zooming, and retrieving details of data in highly interactive information displays have comprised central parts of the methodology.

The field of visual analytics has been conceived as an interfacing science that leverages, and even merges, techniques from both areas to support problem solving with large, dynamic, inhomogeneous, and uncertain data. It adapts technologies to either enhance traditional data visualizations with machine learning tools, and/or to better understand and steer automated data analysis by incorporating visual interfaces. In the first case, computational models are used to filter, summarize, and organize data to facilitate scalable visual representations. They can also help to detect or predict patterns, anomalies, and events in order to highlight them to the user. In the second case, intermediary results and parameters of analytical models are visualized and directly controlled by the user, or the training of such models can interactively be supervised and adapted.

There are still many important problem domains and open visual analytics research questions that have not been addressed. And the demand for computational powerful, yet semantically adaptive solutions is growing every year.

Specifically with the increasing popularity of *social media* services, the internet is flooded with humungous amounts of information on a daily basis. The challenge of exploiting this information for various purposes, such as crisis intelligence, marketing research, public safety, and social sciences, poses completely new demands for data analytics. Intelligent and semantically aware solutions are needed because the challenge not only lies in the volume and velocity of data, but also in the difficulties of understanding and relating content as well as in the specifics of handling inhomogeneous information. In terms of visual analytics, open research questions particularly exist in the areas of scalable real-time analysis, automated anomaly highlighting, and integration of supervised and unsupervised methods. Driven by the challenge of leveraging social media to enable situation awareness, the goal of this thesis is to present a model, generic techniques, and a software platform to advance the field.

1.1 Motivation

On January 15, 2009, a single Twitter message changed substantially how many intelligence and media analysts saw the world. Janis Krums, a New York visitor, was on a tourist ferry on the Hudson river, when suddenly US Airways Flight 1549 conducted an emergency landing on the water. Krums' reaction was to get out his phone, photograph the half-drowned airplane, and send the picture to his 170 Twitter followers. Krums' ferry helped in the rescue operation and all passengers survived. His message, however, was one of the first pieces of electronically transmitted semantic information about the event. Not even air traffic control of LaGuardia Airport had completely realized at this point what had happened. By means of numerous re-tweets, the message circled the globe literally at the speed of light. And before authorities were able to fully grasp the situation, hundreds of web users had already seen the photograph.

In the years to come, Janis Krums was only the first representative of a new kind of casual, mobile, and ubiquitous on-site event reporters that emerged together with the growth of social media. Since that time, people have been using the services to provide eyewitness reports of ongoing bushfires, earthquakes, thunderstorms, floodings, public riots, contagious diseases, criminal acts, and various other critical events. Attached with pictures and videos, their messages have provided accounts of the affected areas of events and their severity, personally observed damage and injuries, problems with support and transport infrastructure, as well as information on the progress of preparatory and evacuation activity. What many analysts therefore realized at this point, was that social media is not just a fun means to communicate with friends and

► **Figure 1.1 —** The message of Janis Krums, written on January 15, 2009, was one of the first disaster-related observations reported in Twitter. The image was linked via TwitPic, an auxiliary service for attaching Twitter messages with media. Similar on-site eyewitness reports can significantly help to support situation assessment.

to share funny pictures, but that it can also serve as a sophisticated information source to facilitate remote, real-time, and ubiquitous situation awareness.

However, there are also significant challenges involved in harvesting information from these sources. The major difficulties in utilizing the data to generate insights exist between the poles of extracting the right data, finding the needle in the haystack, and creating a comprehensive overview out of millions, sometimes cryptic and highly context-dependent information items. In 2009, Krums' message was one of approximately 30 million Twitter messages that were posted on this day. Until 2014, Twitter experienced significant growth and his message would now have been one of 500 million daily messages. That his particular message quickly reached so many people was also a matter of fortunate circumstances, as his large follower base frequently read his messages and quickly re-posted it to influential participants in the network. However, important messages often just drown in the flood of data, or, even more frequently, they are just pieces in a larger puzzle. In this case, they also need contextualization before their information content can be helpful.

Social media analytics has thus been identified as one of the first true *big data* problems of our time [IBM et al., 2011]. The reason for this view, are not only the challenges with message quantity (volume) and real-time streaming nature (velocity) of the data, but also of incorporating different forms, sources, and structures of posts (variety), as well as of understanding, relating, and verifying their content (veracity). These four essential aspects, which are also called the "four V's" of big data, have been commonly used by many authors to define the challenge [see Ming et al., 2013; Wang et al., 2014]. By tightly integrating machine learning, natural language processing, and data visualization techniques, visual analytics has been devised to tackle particularly this kind of problems. As a descendant of the more general field of information visualization, it strives to merge the strengths of highly interactive visual interfaces with the computational power of automated statistical models. The goal of this combination is to advance problem solving in areas where a human analyst alone would be overwhelmed by the data volumes, while, at the same time, sheer processing power alone would not enable the discovery of underlying patterns and to relate contents to semantic knowledge.

1.2 Contribution

The primary contribution of this thesis is a novel visual analytics model devised to enable sophisticated information harvesting from streaming text data. The ultimate goal is to support the analyst in all stages of the process from formulating an initial information need to generating a holistic situation overview.

The thesis thus identifies major challenges that can be addressed by visual analytics means and presents a methodology to tackle them. To this end, a thorough literature review was conducted, which, among others, investigated accounts from crisis response experts, public safety professionals, and visual analytics as well as social science researchers. Based on their reports, required capabilities to facilitate sophisticated situation awareness from social media have been identified. The highlighted difficulties of analyzing the data can be categorized in four dimensions, which essentially reflect the four initial research questions of this thesis:

- How to get the right data from rate-limited web APIs and explore remote datasets with limited accessibility? (**Data Access**)
- How to visualize data in a consistent situation overview, aggregate and contextualize messages, and highlight anomalies? (**Data Context**)
- How to drill down on events and topics, understand semantics, and filter relevant items in ongoing monitoring? (**Data Complexity**)
- How to create highly interactive systems that can cope with millions of records per day? (**Data Management**)

Based on these questions, the thesis develops a model that comprises three distinct techniques addressing *access*, *context*, and *complexity*. They have been implemented in a prototypical visual analytics platform, called *ScatterBlogs*, which integrates all of them to allow scalable real-time *management* of the data. The model arranges the three components in a larger analytics cycle that consists of pulling the data from the service APIs, filtering them by means of adaptive classification, and highlighting anomalous events and outliers in a comprehensive overview.

Along these lines, it is also shown how the aspects of indication and drill-down tightly relate to each other and can be aligned with an iterative analytics cycle. Applying automated aggregation helps to better understand filtered data, which then informs the creation and orchestration of new filters to allow drill-down. The result of this stage either triggers the next cycle or enables ongoing monitoring of the situation with an optimized configuration. The lessons learned with the model will thus also help to acquire a deeper understanding of the specific role that unsupervised and supervised models can play in visual analytics of streaming data.

1.3 Overview

The thesis is structured as follows. Foundations of social media, situation awareness, and visual analytics are presented in Chapter 2. They, respectively, constitute the data basis, challenge, and methodology of this thesis. The details of the central approach, the overarching social media analytics model, will also be discussed at the end of that chapter. Following these preliminaries, the thesis presents the four distinct components of the model, which correspond to the central research questions.

The problem of data access is solved by means of iterative query optimization in Chapter 3. An analyst can interactively explore samples of filtered data streams pulled from available web APIs and employ a novel algorithm to automatically generate queries based on user-selected topics. By this means, the analyst can iteratively investigate relevant subsets of the dataset that is "hidden" behind the APIs and come up with a monitoring query that fits his or her continuous information need.

The challenge of generating aggregated overviews, providing anomaly indication, and highlighting outliers is addressed by a novel event discovery scheme in Chapter 4. It combines automated anomaly identification with an adaptive spatiotemporal visualization. To this end, an unsupervised machine learning model based on K-Means was specifically designed to facilitate scalable processing of real-time data and to optimize recall in time-critical environments. While the algorithm can produce an overfitting of the data, this limitation is turned into a benefit by the visualization that uses it to convey all possibly relevant information by means of semantic zooming. To enhance the basic discovery scheme, a language-based model is shown that allows to assess the statistical abnormality of previously discovered content.

Chapter 5 introduces a method to interactively filter the data, drill down into relevant topics, and allow ongoing situation monitoring. Supervised classifiers have previously been identified as powerful tools to separate relevant information from signal noise. However, creating and applying them has been a difficult and time-consuming task. In this thesis, a technique is explained that combines visual active learning of classifiers with means of highly interactive classifier orchestration. Analysts are thus enabled to better understand the filters and to perform task-adaptive drill-down and monitoring.

The data management aspect is addressed by the visual analytics system *ScatterBlogs*, which will be introduced in Chapter 6. It implements and integrates all of the presented techniques and served as a reference platform to evaluate them. Its sophisticated architecture handles large volumes of streaming data and allows plug-in integration of versatile tools. It furthermore provides mech-

anisms for searching, filtering, and exploring archived social media records and allows a seamless transition from post-analysis to real-time monitoring.

All of the developed techniques, the model, and the reference implementation have been thoroughly evaluated. In addition to case studies and algorithm benchmarks that are presented underway, Chapter 7 discusses results of two conclusive evaluations that have been conducted to examine the individual solutions depicted in this thesis. Based on a broad-scale field study with almost 30 domain experts as well as a real-time analytics user study of the query optimization component, the chapter conclusively demonstrates the applicability and usefulness of the methodology. Following these observations, the thesis concludes with a summary discussion, final remarks, and future perspectives in Chapter 8.

Parts and ideas of this thesis were already published in various journals and proceedings. Where appropriate, ideas, content, and material from the following publications was used:

- H. Bosch, D. Thom, M. Wörner, S. Koch, E. Puttmann, D. Jackle, and T. Ertl. ScatterBlogs: Geo-spatial document analysis. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 309–310. IEEE Computer Society, 2011
- D. Thom, H. Bosch, S. Koch, M. Wörner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 41–48. IEEE Computer Society, 2012
- D. Thom, H. Bosch, and T. Ertl. Inverse document density: A smooth measure for location-dependent term irregularities. In *International Conference on Computational Linguistics COLING*, pages 2603–2618. Indian Institute of Technology Bombay, 2012
- H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Krüger, M. Wörner, and T. Ertl. ScatterBlogs2: Real-time monitoring of microblog messages through user-guided filtering. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2022–2031, 2013
- D. Thom, H. Bosch, R. Krüger, and T. Ertl. Using large scale aggregated knowledge for social media location discovery. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1464–1473. IEEE Computer Society, 2014

- D. Thom, M. Wörner, and S. Koch. Scatterscopes: Understanding events in real-time through spatiotemporal indication and hierarchical drilldown. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, VAST Challenge, pages 1–2. IEEE VAST USB Proceedings, 2014
- D. Thom and T. Ertl. TreeQueST: A treemap-based query sandbox for microdocument retrieval. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1714–1723. IEEE Computer Society, 2015
- D. Thom, R. Krüger, T. Ertl, U. Bechstedt, A. Platz, J. Zisgen, and B. Volland. Can Twitter really save your life? A broad-scale expert study of visual social media analytics for situation awareness. In *IEEE Pacific Visualization Symposium (PacificVis)*. IEEE Computer Society, 2015, to appear

Various research conducted in the course of this thesis is not directly related to the presented model and techniques. Interested readers can refer to the following co-authored publications for further visual analytics research in the areas of movement behavior analysis, predictive analytics, web content orchestration, and similarity metrics: [Klenk et al., 2009], [Heim et al., 2011], [Bosch et al., 2011a], [Krüger et al., 2012b], [Krüger et al., 2012a], [Chae et al., 2012], [Bosch et al., 2012], [Jäckle et al., 2013], [Mittelstädt et al., 2013], [Krüger et al., 2013a], [Chae et al., 2013], [Krüger et al., 2013b], [Andrienko et al., 2013], [Krüger et al., 2014a], [Zisgen et al., 2014], [Krüger et al., 2014b], [Chae et al., 2014], [Lu et al., 2014], [Mittelstädt et al., 2015].

CHAPTER 2

Foundations and Approach

It was a combination of various scientific advancements, popular research directions, and novel challenges that eventually culminated in what is known today as visual analytics (VA). Most important influences can be found in the conception of *visual data mining* [Shneiderman, 2001; Kreuseler and Schumann, 2002; Keim, 2002], which comprised novel ideas to combine information visualization and automated data analysis; the notion of *information foraging loops* [Pirolli and Card, 2005], which modeled the process of sensemaking in analytical reasoning; and the book "Illuminating the Path" [Thomas and Cook, 2005], which highlighted the practical relevance of these approaches and proposed a research agenda of enabling visual access to big data challenges.

This chapter provides an overview of the field of visual analytics, presents associated formal models, and relates them to the more general research area of information visualization. It furthermore highlights common features in the domain, such as fusion of automated and interactive components, humans in the loop, integration with analytical reasoning, and usage of data exploration tools. Moreover, we will have a look at statistics-driven data mining components, which play an important role in this thesis.

The second part of the chapter reviews various forms of existing social media services and illustrates how a unified perspective on this kind of data source can be achieved. We will furthermore see how various big-data properties make analysis and information harvesting in this domain a challenging task. At the same time, it is also highlighted how the widespread adoption of the platforms

and the global penetration make them such a valuable information source and a perfect playground for visual analytics.

Based on recent studies on the topic, the relevance of social media to situation awareness, particularly in the context of disaster management, crisis response, and critical infrastructure management, is discussed in the third part. Along these lines, it is highlighted how visual analytics of social media can blend in with existing models of situation awareness and sensemaking.

After the foundations have been laid, the chapter concludes with an introduction of the social media analytics model. It outlines proof-of-concept components, an overarching analytics cycle, and a prototypical implementation of both that illustrate how visual analytics can comprehensively leverage information to enable task-oriented situation awareness.

Parts of this chapter have previously been published in:

- D. Thom and T. Ertl. TreeQueST: A treemap-based query sandbox for microdocument retrieval. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1714–1723. IEEE Computer Society, 2015
- D. Thom, H. Bosch, S. Koch, M. Wörner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 41–48. IEEE Computer Society, 2012
- H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Krüger, M. Wörner, and T. Ertl. ScatterBlogs2: Real-time monitoring of microblog messages through user-guided filtering. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2022–2031, 2013

2.1 Visual Analytics

A common conception of visual analytics brings the fields of visualization and human computer interaction (HCI) as well as knowledge discovery and data mining (KDD) together to tackle the challenge of analyzing large, heterogeneous, high-dimensional, unreliable, and real-time streaming data. Auxiliary research can furthermore be found in data management and data fusion, which are necessary to store and process big and heterogeneous data as well as perception and cognition science, which consider multimodal user interfaces that scale with the volume of visual input. Visual analytics is considered particularly

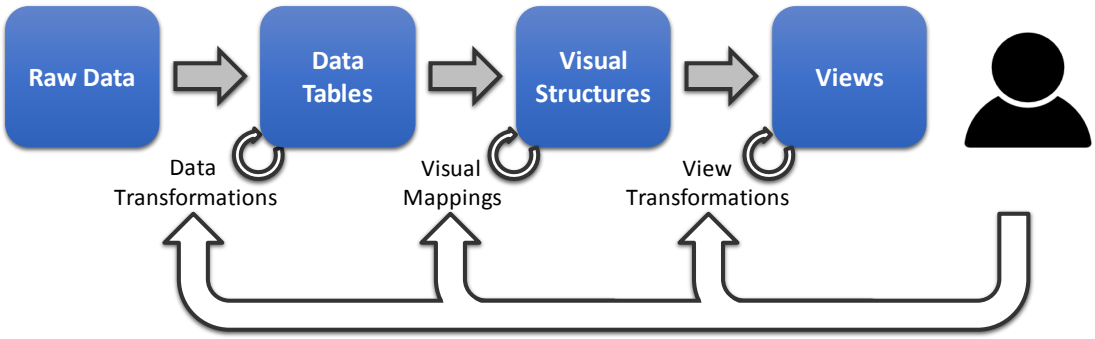


Figure 2.1 — The information visualization pipeline. Model and figure were adapted from [Card et al., 1999, p. 17].

relevant if the goals of analysis are vague and available knowledge about the data is sparse.

2.1.1 Abstract Data Visualization

The roots of visual analytics can be found in the scientific community and research area of information visualization (InfoVis) - the science of creating interactive visual representations of abstract data. In this area, the process of creating such interfaces and facilitating insight generation is firmly rooted in the idea of data visualization pipelines. An often cited model, defined by Card et al. [1999], describes the transition from raw data over structured data tables and visual structures to rendered graphics (Figure 2.1).

In this model, raw data can refer to all kinds of file or database formats, such as CSV-files or SQL-databases, from all kinds of sources, such as text corpora, sensor readings, or spreadsheets. By transforming them into data tables, the data is represented as relational property tuples based on defined data types. In the next step, these attributes are mapped to graphical properties, such as relative position of a line, length of a rectangle, or color of a disc, that together form combined structures, such as a bar chart or a network diagram. In the last step the computed structures have to be rendered, i.e., based on the visual structures and view transformations, a color has to be assigned to each pixel on the screen. In the complete process, user interaction can influence the transition between stages, such as selecting data transformations and filters, configuring visual mappings, e.g., colors and shapes, as well as enabling various view transformations, e.g., zooming and panning.

While such pipelines describe the algorithmic side, i.e., how to build programs that turn data into images, InfoVis has also established significant knowledge

in designing interactive user interfaces. In this regard, a fundamental architecture principle of information visualization was prominently introduced by Ben Shneiderman [1996]. The *visual information seeking mantra* tells us to always allow "[o]verview first, zoom and filter, then details-on-demand". An initial visual representation should thus always show the complete collection of available data before interactive means are used to explore interesting parts and investigate details of possibly relevant items. Illustrative examples that follow the pipeline and mantra include classic tree and graph visualizations, such as Vizter [Heer and Boyd, 2005], hierarchical visualizations, such as zoomable Treemaps [Johnson and Shneiderman, 1991; Blanch and Lecolinet, 2007], or interactive diagram-based techniques, such as Gapminder [Rosling, 2007].

However, the more or less direct mapping from data items to visual items of the pipeline quickly reaches its limits once we move from several thousand data records to several million data records. In this case, one quickly experiences challenges of [also cf. Thomas and Cook, 2005, pp. 24–28]:

- **Data Scalability** - Limited number of records that can be handled by database management, in-memory processing, and index structures.
- **Visual Scalability** - Limited maximum number of visual items that can be shown by display hardware or perceived and comprehended by a human.
- **Interaction Scalability** - Limited data processing and rendering performance to enable fluent and dynamic user interactions with the visualization system.

By incorporating automated analysis into the visualization process, we can build models that allow us to aggregate, organize, and adaptively filter data, to automatically highlight anomalies and outliers, and to incorporate iterative reasoning in the process. We can therefore reduce the number of visual items by providing summarized overviews, better utilize storage and processing architecture through intelligent preprocessing strategies, and enhance interactions by preventing the display of signal noise and superfluous information. Based on these premises, visual analytics strives to advance information visualization by tackling the scalability challenges, facilitating comprehensive exploration, and allowing an analytical approach to understanding vast data volumes.

2.1.2 Definition and Model

One of the early definitions of visual analytics was provided by Thomas and Cook [2005] in their groundbreaking work "Illuminating the Path: The Research

and Development Agenda for Visual Analytics". Under the impression of the September 11, 2001 attacks on the US homeland and the limited capabilities of security analysts to predict and prevent such threats, they highlighted the necessity to create a new research area that would enable "analytical reasoning facilitated by interactive visual interfaces". Besides the security-centered aspects, other application domains were quickly identified by researchers. They include climate monitoring, marketing, economics, transportation, aviation, digital humanities, urban mobility, patent research, and computer networks, as well as supportive analysis in astronomy, chemistry, biology, physics, and medical science.

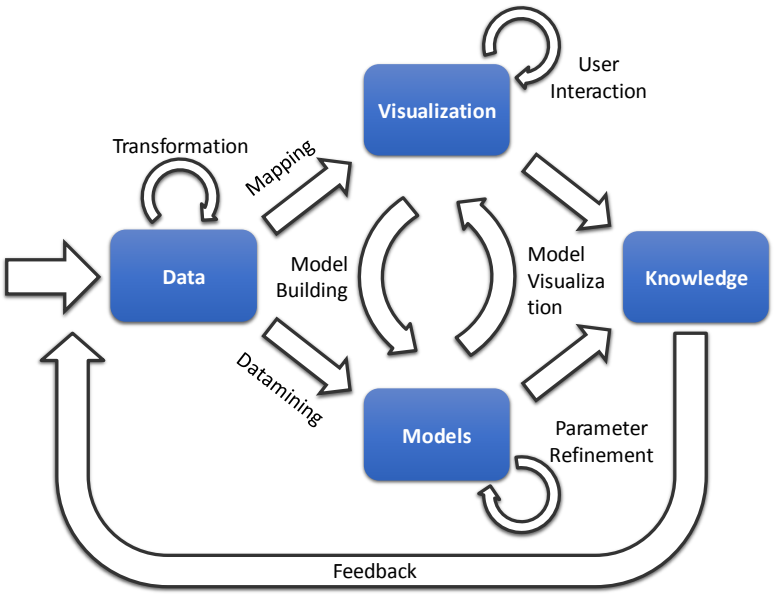
Based on the initial conceptions, Daniel Keim et al. [2008, 2010] comprehensively defined the *why*, *what*, and *how* of visual analytics from a slightly different perspective. They see the goal of visual analytics not just in tackling challenges that cannot be solved by traditional InfoVis, but rather in turning the data overload into an actual opportunity to create new capabilities. For example, in the context of this thesis, one could not just try to enable emergency analysts to filter social media for emergency calls and directed requests from the public, but also to employ it as a real-world sensor by discovering useful information and suspicious patterns in undirected chatter.

Following the overarching goal of turning data volumes into possibilities, the *what* of visual analytics is answered by Keim et al. with the following definition, which extends the earlier conception of Thomas and Cook:

> Visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision-making on the basis of very large and complex datasets. 2008, p. 157

In this new definition, *automated analysis* now plays a central role and particularly refers to statistics-driven data mining tools, such as associative rule building, regression, classification, and clustering. The combination of means is thus proposed to tackle challenges that can neither be covered by plain algorithmic approaches nor by sole cognitive efforts. Instead, automated analysis is used to facilitate big data processing, and human cognition is incorporated to employ domain knowledge, discover unexpected patterns, and enable analytical reasoning.

Finally, the *how* of visual analytics is formalized by Keim et al. with the knowledge generation model (Figure 2.2). It describes the process of visual analytics systems as a transformation from data to insights [Keim et al., 2008]. In the graph, the arrows describe functions of data processing, hypothesis building,



◀ **Figure 2.2** — Visual analytics knowledge generation model according to Keim et al. [2008, 2010].

visualization, and user interaction. Raw data input is first cleaned, pre-filtered, and mapped to usable data structures. In the upper branch of the model, data structures are mapped to visual structures, and visualizations are rendered. The user can interact with them by zooming, panning, selecting, and other operators. The sub-graph spanned by the nodes *Data* and *Visualization* thus resembles the information visualization pipeline as introduced in Section 2.1.1. It is further enhanced by means of automated and user-driven hypothesis generation in the lower branch. Statistical algorithms are used for automatically forming hypotheses from data and creating models. Based on these models, additional visualizations can be produced that illustrate underlying structures to the user. By exploring the visualizations from input and model data, the user can form own hypothesis and further refine the model by adapting its parameters. The final outcome of the process are insights and knowledge, which are either directly gathered from visualizations or constitute an output of the formal models. The complete process is embedded in an iterative loop: The generated knowledge at the end of the process can again be used to select new input or refine the preprocessing at the beginning.

2.1.3 Common Characteristics

The definitions and models by Thomas, Cook, and Keim et al. significantly helped to lead the way. However, the field is still young, and it might not yet be decided what will eventually become the most convincing and most useful conception employed in the scientific community. Several important works published at major visual analytics conferences and journals would

not easily fit under the existing models or would at least require significant extensions. Tableau Desktop[1], for example, is often cited as one of the few commercially available visual analytics systems - yet it does not exhibit any form of automated model generation. More important to its recognition as visual analytics system are its tight integration of data preprocessing, visual exploration, and information retrieval, as well as its sophisticated support in real-world analytical reasoning. To broaden and to better understand the scope of visual analytics, new definitions and models are thus still being developed and advanced [e.g. Sacha et al., 2014].

An ostensive conception of the field, which is implicitly assumed by many researchers, can be thought of as something Ludwig Wittgenstein [2001] would have called a *family resemblance*: There is no single defining feature that is shared by all approaches, nor is there an exclusive set of necessary and sufficient conditions that tells us when the term can be applied. Instead, the approaches are connected by overlapping similarities. In effect, these similarities establish a tightly linked network of instances allowing us to draw a mental boundary. The idea of visual analytics is thus often conveyed to students, researchers, and other audiences by pointing out examples that exhibit one or more of the characteristic features of the field. The following paragraphs highlight the features that are specifically relevant in the context of this thesis.

Fusion of data mining and interactive visualization

Before visual analytics became an important concept, the area of visual data mining already introduced the idea to combine techniques from data mining and interactive data visualization [Keim, 2002]. These approaches were primarily aimed at better understanding automated analysis by enabling result visualization and interactive parameter selection, or at enhancing visualizations by means of automated data organization, information aggregation, outlier detection, and pattern recognition. For example, in the SGI MineSet system [Brunk et al., 1997] one can visualize a 3D overview of automatically generated decision trees and interactively adjust classification parameters. An example of employing automated data organization was shown by Trutschl et al. [2003], who use self-organizing maps (SOM) to show more data items in scatterplots based on artificial jitter.

However, in visual analytics, the basic approach of *combining* existing means is elevated to the point were techniques from both fields are specifically designed to *merge* with each other and to blend in with the process of analytical reasoning. More than just showing the execution and results of automated

[1] `http://www.tableausoftware.com/products/desktop`

analysis, visual analytics allows to interactively *steer* the process and explore results based on interactive adjustments. And more than just employing analytical algorithms as auxiliary tools, visual analytics designs *build* on their characteristics to enable new perspectives on the data. Creating such methods thus raises significant challenges in both areas and requires more than the mere combination of out-of-the-box solutions. Examples of such approaches will be shown in this thesis, including specifically adapted hierarchical and partitional cluster analysis (Chapters 3, 4) as well as visualizations that turn limitations of automated classification into means of facilitating task-oriented problem solving (Chapter 5).

Human in the loop

In traditional information visualization, the user operates the system as a tool to solve a specific goal, while the visualizations primarily serve as a human-computer interface to abstract data. In visual analytics, by contrast, it is often not only the system that is used by the user, but, at least in a figurative sense, the user that is also used by the system. While the computer offers powerful means to process and statistically evaluate large and complex data, there can be steps in the analytical process where semantics, context knowledge, experience, or plain intuition are needed to produce the right kind of results. Especially in domains with human-generated textual data, human-generated media, or real-world sensor data, experience and skills of domain experts can not be algorithmically formalized. In addition to that, it is often required that a human takes responsibility for decision-making and actions. Several visual analytics approaches thus rely on an iterative process, in which intermediate computing results are visualized to the user, who in turn can inform the system about the best next steps (Kerren and Schreiber [2012] provide recent discussions on this aspect). Visualizations are thus sometimes designed to query the user for new information and commands or to incorporate the human perspective on a problem that cannot be solved well by the machine. An example will be shown in Chapter 5 of this thesis. In this case, the analyst's opinion on the semantics of tweets is queried by the system to inform the automated creation of classifiers.

Emphasis on sensemaking and analytical reasoning

To some degree, the visual information seeking mantra already reflects aspects of reasoning, such as exploring a problem domain and breaking larger problems into smaller pieces. However, the primary function of traditional InfoVis is to make abstract data accessible to humans. Their design is usually driven

by the visualization pipeline and thus implements a rather straightforward transformation from data to images to achieve this goal. If applied to analytical tasks, the users have to decide for themselves when and how they can employ the visualization. Visual analytics, on the other hand, often starts from a pipeline of analytical reasoning - sometimes characterized by a specific domain [Endert et al., 2014] - and examines how visualizations and automated algorithms can be interwoven with that process.

Several approaches adhere to the analytics model presented by Pirolli and Card [2005]. Based on empirical studies with various intelligence analysis experts, they organize the process of analytical reasoning in two activity loops called *information foraging* and *sensemaking*. In the foraging phase, situation-related documents and media are first searched, filtered, and collected into a storage (shoebox) from external sources, e.g., websites or news media. Secondly, by reading collections of documents, relevant information snippets are extracted and organized in evidence files. Finally, the information from the files is arranged in a schema, such as a mental model or a computer-based visualization. In the sensemaking phase, hypothesis and theories are first built based on the schema, and a representation, report, or a story is then produced to inform decision makers and other audiences. The phases are conceived as loops, as analysts would often move back from later steps to earlier steps. They could, for example, re-evaluate stories based on feedback from decision makers, search for further theory support in the schema, re-examine evidence and draw new relationships, or search for additional snippets and documents based on their earlier hypotheses.

Visual analytics systems such as Aruvi [Shrinivasan and van Wijk, 2008], TRIST/nSpace [Jonker et al., 2005], and PatViz [Koch et al., 2011] have been presented that tightly integrate with various stages of this model. Approaches employ information retrieval to collect documents, clustering and classification to filter and organize entities, highly interactive visualizations to represent schematized knowledge, tools to formulate and verify hypotheses, and components that record analytical provenance and provide automated reporting capabilities to convey results to others.

Data-generic and integrated combination of exploration tools

Enabling data exploration as part of visual analytics is often facilitated by integration of multiple visualization paradigms. For example, *overview+detail* is the idea of letting users investigate relevant or interesting parts of a dataset, but at the same time providing them with an overview of the complete set in a separate window [Card et al., 1999]. A popular example of this idea are

miniature representations of zoomable geographic maps that always show the complete map in a small view. In a similar vein, the idea of *focus+context* also allows users to investigate details of the data (focus) but surrounding or related background information (context) is shown in the very same place. Most frequently employed techniques from this category are fisheye views [Furnas, 1986; Sarkar and Brown, 1992], which use spatial distortion of the area surrounding a selected focus region, and exploration lenses [Bier et al., 1994; Panagiotidis et al., 2011; Hurter et al., 2011], which provide additional information by modifying, filtering, or annotating the presentation within the focus region. Furthermore, *multiple coordinated views* (MCV) and *brushing and linking* [see Carr, 1999] are used as standard techniques to show different perspectives on the data in multiple UI windows and to highlight data subsests that were selected in one window by corresponding visual clues in the others.

All of these techniques were developed and have already been used in the early stages of information visualization. Visual analytics approaches, however, often combine a multitude of them in tightly integrated and data-generic systems that also incorporate high-throughput storage management and sophisticated information retrieval tools. Examples of such systems include Palantir [Wright et al., 2009], Jigsaw [Stasko et al., 2008], and the already mentioned Tableau Desktop. More than just accumulating capabilities, these approaches provide new dimensions of interacting with the data and enable a mutually informed application of techniques. In addition to that, these self-contained solutions enable evaluations with actual domain experts based on complex, multistage, and real-world analytics tasks.

2.1.4 Data Mining Components

Statics-driven machine learning plays an important role in visual analytics in general and in this thesis in particular. Typically, one can differentiate between *supervised* and *unsupervised* methods, which are both used to organize data, recognize patterns and anomalies, as well as to predict future trends and outcomes. The categorical difference is grounded in the different ways how the respective models are trained. To create supervised models, users have to provide a representative dataset that contains samples of correct mappings from possible input parameters to possible output parameters. Based on this training set, the algorithm learns to produce the right outputs for yet unknown inputs. By contrast, unsupervised models are grounded on a given formal definition of *distance* or *neighborhood* among data objects and use it to automatically detect groups, patterns, and outliers in a given dataset.

Supervised Methods

Supervised models are most often associated with *classification* and *regression*. To perform regression, the model is trained with value-tuples that map input vectors to correct output vectors. Based on this data, the algorithm builds a function that approximates the hidden behavior, which can then in turn be used to map unknown values to probable outcomes. For example, based on movie features, such as budget, actor star value, and genre, a model could predict future box office revenues [Joshi et al., 2010].

In case of (binary) classification, the algorithm is presented with examples of data objects that belong to one of two classes. Usually these data objects are represented in a vector space where each dimension corresponds to a feature, e.g., for textual data, a document could be represented by a vector of the frequencies of term occurrences. The desired training result is a classifier, e.g., a high-dimensional hypersurface or a decision tree, that best separates the two classes in the feature space. For instance, the classes could be spam and non-spam e-mails. The classifier, which results from training with labeled examples of both, could then be a hyperplane that separates their term frequency vectors based on distinctive language use. Most popular algorithms for classification include *neural networks*, *random forests*, *naive Bayes* and *support vector machines* (SVM). A comprehensive overview of techniques can be found in [Kotsiantis, 2007].

Training of supervised methods can either be done by pre-labeling a corpus of data or by employing *active learning* (AL). In the latter case, the algorithm starts with unlabeled data and iteratively queries the user to evaluate data objects that would particularly help to advance the training process. Depending on the used algorithm, this can be done by selecting objects for which the classification is most uncertain, which produce the largest overall error, or which would have most significant impact on the model. Detailed explanations to active learning and a survey of recent and ongoing research has been provided by Burr Settles [2009].

Unsupervised Methods

In most contexts, unsupervised models are almost used synonymous with *cluster analysis*. They can be further categorized in partitional algorithms, which divide the input dataset into a finite number of disjoint subsets, and hierarchical algorithms, which organize the data in a nested structure. Both of these types work on unlabeled data but assume the existence of some predefined distance or similarity function between data objects. If the data objects are represented in a real-valued vector space, a simple choice can be *Euclidean Distance*. *Cosine*

Similarity is often used in context of textual data. Typically, the goal of clustering is to either partition the data into classes of similar objects, to detect coherent structures or patterns hidden in the data, to separate relevant data structures from noise, or to find outliers not belonging to any group. A comprehensive survey of data clustering was published by Jain et al. [1999].

A frequently employed partitional clustering method is *K-Means* [Lloyd, 1982], as it is fast, easy to implement, and often sufficient to produce the desired results. The algorithm starts with a random partition of the vector space into k disjoint regions. For each region, the mean vector of data objects, called the centroid, is computed. In the next step, the distance between the centroids and each data object is evaluated based on the provided metric, and each data object is assigned to the closest or most similar centroid. Based on this assignment, new initial regions are defined, and the process is repeated by updating the centroids to the new means. The algorithm can be terminated if the centroid locations are no longer changed, or if some quality function, e.g., the average of squared distances, moves below a given threshold. The algorithm is relatively fast, and it always converges to a local optimum because each iteration reduces the average sum of distances. However, depending on the random initial partition, this is not necessarily a global optimum. A further limitation is that the user obviously has to pre-define the number of expected clusters. More recent variations such as X-Means [Pelleg and Moore, 2000] therefore try to iteratively assess the optimum number of k from the structure of the underlying dataset. This concept also served as basis for a newly developed algorithm that will be presented in Chapter 4.

Performance Metrics

If clustering or classification are used for information retrieval, they can be evaluated by performance metrics. For example, in cases like the spam and non-spam classification, the model would usually be applied to retrieve relevant e-mails from the set of all e-mails. The user would thus be particularly interested in one of the classes, while the other class would just contain all the unwanted results. Similarly, clustering can be used to find relevant patterns and anomalies in a dataset and separate it from all other data (i.e. noise). The desired retrieval results would then be the data objects contained in these clusters.

In such cases, the quality of models is often measured in terms of *precision* and *recall*. Precision is the number of correctly detected relevant results divided by the number of all elements that the model considered relevant. The better the precision, the less irrelevant elements the analyst has to check when investigating the result set. On the other hand, recall is the number of correctly detected

relevant results divided by the number of relevant elements in the complete dataset. The better the recall, the higher the chance that analysts will find all results relevant to their reasoning.

However, solutions can easily optimize just one those aspects. For example, precision can be optimized by detecting just one relevant element and assigning all other elements to the non-relevant class. Conversely, recall can be optimized by simply returning all elements as relevant. Therefore, solutions are only reasonable if they optimize both, precision and recall or achieve at least an acceptable trade-off. This quality can be measured by the F_1-Score, which combines both aspects:

$$F_1 = \frac{2 * precision * recall}{precision + recall}$$

As precision and recall are both valued between 0 and 1, the best F_1-Score is also 1, and the worst is 0. If only one of both is near 1, and the other is near 0, the F_1-Score will also be near 0, which properly reflects the bad trade-off.

2.2 Harvesting Information from Social Media

The term *Web 2.0* was popularized in 2004 by Tim O'Reilly to describe novel internet platforms that break with the static page viewing nature of the early web. That architecture would enhance the dynamics of interaction, lower the barriers for sharing user-generated content, and enable online social networking. In other words: *Web 2.0* highlights the transformation from a web of *consumption* to a web of *participation*.

While the Web 2.0 idea was more focused on the technological and business-related aspects of the platforms, the notion of *social media* recently became more popular in the public discourse [cf. Schuerig, 2014]. The term is also more frequently used to address the impacts that the services have on society: In the pre-internet era, reaching a broad audience through media was a more exclusive ability of journalists, book authors, musicians, filmmakers, politicians or scientists. Through the rise of the early internet, this exclusive circle was then extended to skilled enthusiasts that tackled the technical challenges of creating a website and making it accessible to others. Finally, with social media platforms, the technical effort has been reduced to basically writing some text and clicking a button in an existing web client or an easy-to-use mobile application. Nowadays, social media is thus frequently used by large parts of society to share regular updates on their thoughts, opinions, observations, feelings, or physical conditions, and to connect with their friends, relatives or the general public. To illustrate the variety of social media, Solis and JESS3

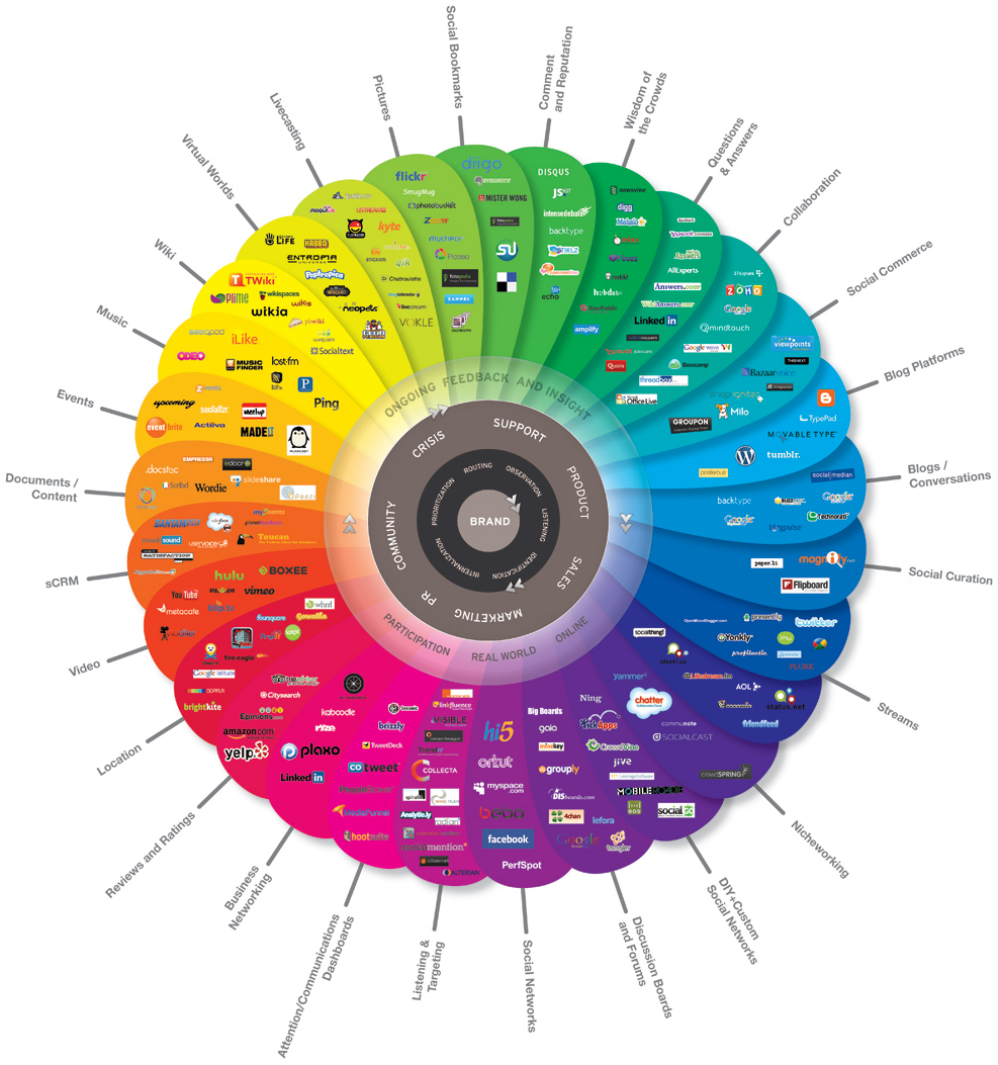


Figure 2.3 — The diagram shows an overview of popular social media services organized in 28 distinct categories. In terms of situation awareness, the relevant services can broadly be categorized into microblogging, content-sharing, and social networks. (Image created by Solis and JESS3 [2014] / CC BY 2.5).

[2014] recently published an overview of platforms comprising more than 220 services organized in a landscape of 28 distinct categories (Figure 2.3).

However, from a data analyst's viewpoint, who is mostly concerned with situation awareness, the most relevant services can essentially be summarized in three primary categories: First, there are microblogging and microinteraction sites, which allow users to upload and share short snippets of textual, referencing, or evaluative information, such as personal status updates, URLs to other

websites, product ratings, or the user's current geolocation. Popular examples for this type include Twitter[2], Tumblr[3], Bitly[4], Yelp[5], and Foursquare[6].

The second category comprises content-sharing platforms, which allow users to provide user-generated videos, images, or sound-files, and also to comment on the content of other users. Youtube[7], Instagram[8], Flickr[9], and Soundcloud[10] are frequently mentioned platforms in this domain. And finally, there are social networks, such as Facebook[11] and Google+[12], that are used to map and manage social links and to allow more directed and private communication with other users or specific groups of users.

The first two of the aforementioned categories are particularly relevant in situation awareness. Even more so if users frequently post from mobile devices, such as smartphones and tablets, and provide location data based on the device's GPS-capabilities. Although there is also important information shared on social networks, they are much less accessible to most analysts due to the more private nature of the services: Since the idea of microblogs and content-sharing platforms is to publish information to a broad audience, the provided content is often publicly available by default. By contrast, in social networking sites, users mostly have to explicitly opt-in to share their communication with people outside of their private circles. It is therefore often difficult or even impossible to acquire and utilize larger volumes of this kind of data on a legal basis.

2.2.1 Social Media Microdocuments

Most social media platforms have a defining primary feature or support a certain activity for which they are best known and which allows their categorization, as it was done in the previous section. However, it is important to note that the boundaries are neither strict in terms of content types or structure, nor in terms of the specific information sharing behavior of the communities. For

[2] http://www.twitter.com
[3] http://www.tumblr.com
[4] http://www.bitly.com
[5] http://www.yelp.com
[6] http://www.foursquare.com
[7] http://www.youtube.com
[8] http://www.instagram.com
[9] http://www.flickr.com
[10] http://www.soundcloud.com
[11] http://www.facebook.com
[12] http://plus.google.com

example, while the focus of Twitter is microblogging, people can also attach images, videos, and URLs to their posts. In Youtube and Flickr they can provide textual annotations with their media, and in almost all services they can register connections to other users to form some kind of social network. Based on this observation, social media aggregators, such as Datasift[13] and Gnip[14], have been established that enable data analysts to abstract from the specifics of each platform. They allow generic requests, e.g., based on textual keywords, geographic bounding boxes, and temporal ranges, in order to deliver corresponding data from multiple platforms in a unified format.

Following these existing standardization efforts, each singular social media entry, such as a Twitter message, a Youtube video, or a Facebook post, can essentially be described by a range of formal attributes. This kind of unified information container has also been referred to as *microdocument* [Wu et al., 2011]. Three primary attributes of these documents are specifically relevant in situation awareness, as they allow to assign provided information to a distinct point in space and time. They also serve as pivotal data features in the approaches of this thesis:

- **Location** - Spatial information can be given in forms of geographic coordinates based on GPS measurements (e.g. latitude, longitude), in forms of a geographic label manually assigned by the user (also called a Geo-Tag), or via geocoding that was derived from the user's connection data (i.e. IP-address resolution). Depending on the type of location information, the precision thus varies between the maximum GPS precision, which is between 5 to 10 meters [Wing et al., 2005] and the size of nameable geographic entities, such as districts, cities, or states. In most cases, this information is only provided if the user explicitly enables the feature.

- **Time** - In contrast to location, usually every post is at least assigned a timestamp that includes the date and time of its creation. Although it is often stored exact to the second, the actual precision varies in the range of minutes depending on the service as well as the processing time of the client and server.

- **Content** - In case of microblog services, the content of an entry would be the actual message text, while in Youtube or Flickr, it would be the provided video or image together with its description. Usually, the length or size of possible content is limited, e.g., a Twitter message can only

[13] `http://datasift.com/`

[14] `http://gnip.com/`

contain 140 characters, and a Youtube video can, by default, not exceed a length of 15 minutes.

Furthermore, there are several secondary attributes that can be used to establish context with existing topics, the users' history, social connections, associated media, as well as to assess the relevance of messages and the users' influence in the network:

- **Username** - A screenname or ID that uniquely identifies the author of the post.
- **Tags/Hashtags** - Special identifiers are used by the community to denote popular topics.
- **Usermentions** - References made to other users of the platform.
- **Citation** - Information how often a post has been referenced by other users. Examples include "re-tweets" in Twitter, "likes" in Facebook, and ratings in Youtube.
- **Replies** - Comments or replies that users have written to respond to the post.
- **Hyperlinks** - URLs provided in the entry to attach media or highlight related websites.

By separating primary from secondary attributes, it is also more easy to implement privacy preserving analytics. It will almost always be relevant to see which information has been posted when and where. However, there are several analysis tasks that do not need to consider personally identifiable information, such as connecting multiple entries to one user.

2.2.2 Social Media as a Data Source

Besides formal attributes of individual entries, there exist several properties that characterize social media as a data source for information retrieval. On one hand, we have the big data aspects, including volume, velocity, variety, and veracity of the information streams. On the other hand, from the popularity and widespread adoption of the services also result a tremendous timeliness of reports and global distribution of users. They essentially allow us to employ the community as ubiquitous "social sensors" [Sakaki et al., 2010]. Together these properties make analysis of the data such a relevant, yet challenging endeavor.

Volume and Velocity

The totality of online social media content is on its way to becoming the largest collection of text, images, and video ever created by mankind. According to the services, users upload - on a daily basis - more than 500 million documents to Twitter [2014], more than 60 million photos to Instagram [2014], and about 144.000 hours of video to Youtube [2014].

Most services offer real-time API access, which allows to collect data as a continuous stream of records. To process and index them at such velocities, traditional storage solutions like relational SQL-based databases quickly reach their limits. Most services thus rely on more recent solutions that have been adapted to the specifics of content hosting, such as NoSQL or Graph Databases [see Moniruzzaman and Hossain, 2013]. They fulfill high requirements on frequently inserting and retrieving singular records or pages, i.e., horizontal scaling, at the cost of limited capabilities in terms of data organization, consistency, and relational query complexity.

While the platforms only have to store the data and usually provide limited functionality to search and explore archived entries, the situation is more challenging for data analytics software. For example, while all Twitter messages since the beginning of the service can be retrieved if the URL is known, the platform's search engine only provides access to data with a maximum age of seven days. Visual analytics systems, however, should provide indexed access to the complete corpus of data in order to compare current situations with observations from past events. Furthermore, data analysts have more complex requirements for possible request parameters, aggregation, preprocessing, and real-time visualization than regular users. It is therefore necessary to adapt and enhance existing storage solutions to these specific needs in order to enable comprehensive information mining from the data.

Variety and Veracity

Despite its size, social media is also one of the most unorganized and heterogeneous collection of data ever created. The variety of social media results from the large amount of platforms and the different ways they are used. Moreover, most social media entries are unstructured data, i.e., text and media, which, compared to sensor readings or business data, can address any type of subject matter. Although it has been discussed in Section 2.2.1 how the structure of entries can be unified to some degree, the types and focus of content will still be quite heterogeneous depending on the communication paradigm of the platform. This thesis primarily addresses content that is of particular relevance in gaining situation awareness, i.e., short status updates or observations possibly

assigned with images and videos, and probably provided from mobile GPS-enabled devices. However, future efforts should also address more advanced possibilities of platform orchestration, consolidation, and standardization.

The veracity challenges associated with social media stem from the uncertainties in creditability of the authors, the meaning, contextualization, and provenance of their content, as well as data processing errors, including inaccuracies or missing availability of time and location. There have been cases where users unwittingly or maliciously spread misinformation and rumors that misguided investigations of authorities [Starbird et al., 2014; Oyeyemi et al., 2014]. Moreover, the semantics of individual posts is often difficult to understand without additional background information.

Timeliness and Penetration

While the big data aspects constitute the challenging nature of processing and analyzing social media, there are also several beneficial attributes that allow to turn the data overload into opportunity (cf. Section 2.1.2). According to a recent report from Twitter [2014], more than 75% of its active users frequently access the service from mobile devices. Many of them are thus almost always enabled to provide timely reports, observations, and imagery about ongoing situations. Although difficult to assess, the global penetration of social media usage has been a subject of various reports. A recent survey by We Are Social [2014] allows the conclusion that between 5% to 7% of the population of underdeveloped countries, between 25% to 35% of the population of developing countries, and between 40% to 55% of the population of developed countries participate in some form of social media. The density of social media users, particularly in larger cities, thus establishes a ubiquitous network of possible event reporters.

The timeliness and spatial penetration of the data were impressively validated during an experiment that was conducted as preparatory work for this thesis. Here, Twitter messages related to the 2011 Virginia Earthquake[15] had been collected - i.e. containing the keywords `quake` or `earthquake`. They were plotted on a map at successive time-steps together with a visualization of seismic waves that was based on Rapid Earthquake Viewer [2011] data. The results in Figure 2.4 show how numerous users in all affected regions immediately react to the event. Their geo-enabled messages often appear in less than 20 to 30 seconds after the jolts could have been felt at their respective locations. Although there are surely more profound ways to detect an earthquake, the reaction demonstrated the

[15] `http://en.wikipedia.org/wiki/2011_Virginia_earthquake`

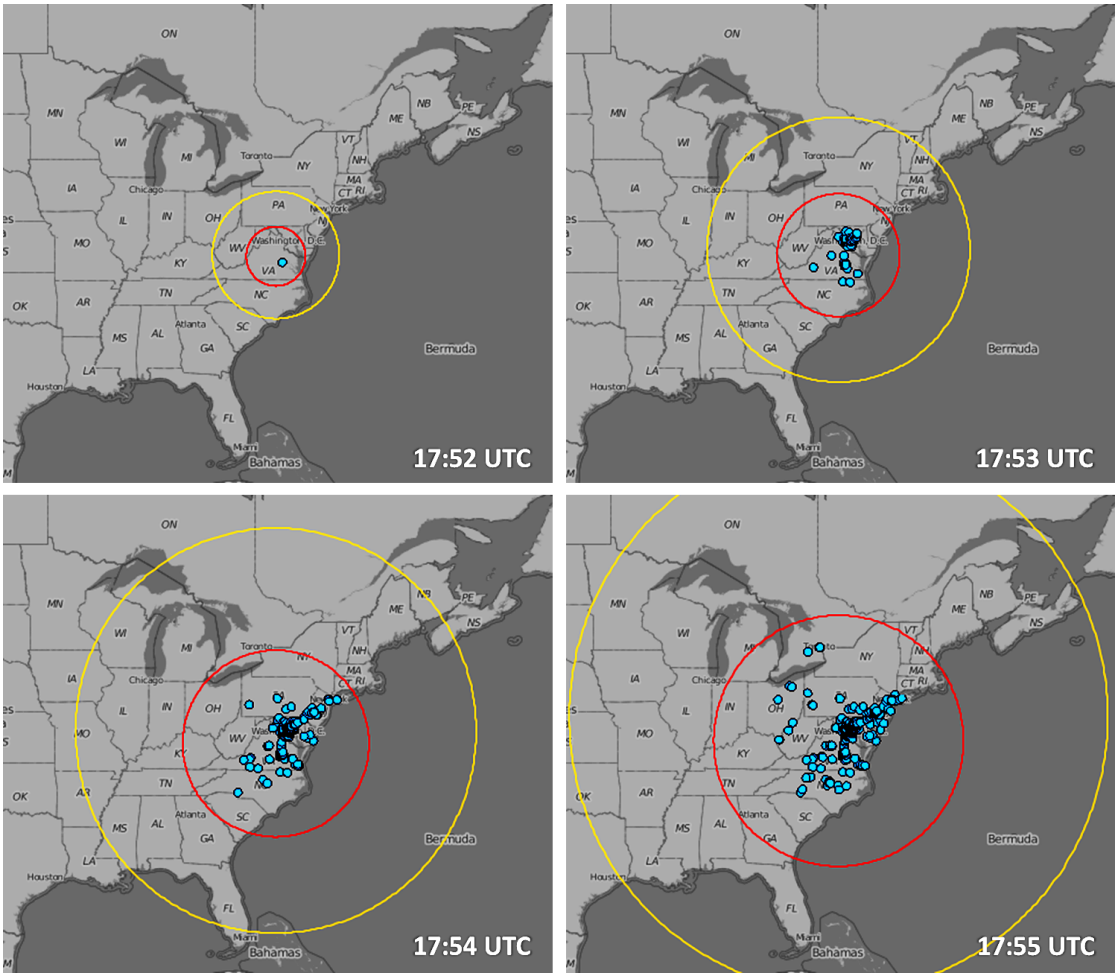


Figure 2.4 — The screenshots illustrate Twitter messages during the 2011 Virginia Earthquake. The earthquake occurred on August 23, at 17:51:04 UTC. The measurable p-wave moved at about 7.5 km/s (outer, yellow circle), and the s-wave (the jolts that most people feel) moved at about 3.5 km/s (inner, red circle). Locations of all messages containing the keyword `earthquake` are marked in blue. The first such message was received at 17:51:46 from Richmond, which is about 60 kilometers from the epicenter (upper left figure). It just reads: *Earthquake in Richmond*. Under consideration of transmission and processing time, this message must have been sent about 20 to 30 seconds after the jolts were first felt in the city.

distribution and behavior of the crowd, and that similar information density and timeliness could also be expected for other events.

2.2.3 Twitter API

Throughout this work, Twitter served as a primary data source for the development and evaluation of techniques and methodology. The service was chosen as representative example due to its size, the public availability of real-time streaming and large-scale data volumes, its ubiquitous usage in mobile environments, the observed timeliness in critical situations, and the variety of content and formats, including attached photos, videos, and URLs. The specifics of retrieving data from the Twitter application programmer interface (API) are therefore addressed in this section. Nonetheless, the service is considered a substitute for all similar services described before, particularly microblogging, microinteraction, and content-sharing sites.

As of this writing, Twitter has more than 240 million active users and an archive comprising more than 300 billion messages (also called tweets) [Twitter, 2014]. The service thus resembles a perfect source to gather public opinions, thoughts, and especially observations. To collect larger amounts of data from the service, developers can access two distinct APIs, called the *Search API*[16] and the *Streaming API*[17], which are, similar to most other services, both subject to significant access-rate-limitations.

The *Search API* allows requests based on keyword queries as well as metadata filters, and it provides results up to a maximum age of seven days. As mentioned before, once reaching that age, tweets are removed from the search index. During important events, tweets can be generated at a rate of more than 140.000 messages per second Krikorian [2013]. However, due to the rate-limitation, the *Search API* only allows the collection of approximately 17.000 tweets within each 15 minute time-slot.

The *Streaming API* accepts similar request parameters and continuously streams data for the requests at a higher rate. Here the limits depend on the access-role of the API user. Without special arrangements, the throughput will usually be bounded at less than 10% of the unfiltered data stream. Also, the Streaming API cannot be used to retrieve past data, and each Twitter account may only create one standing connection based on fixed request parameters. Therefore, analysts will often work with a subset of the actual document corpus and have

[16] `https://dev.twitter.com/rest/public/search`
[17] `https://dev.twitter.com/streaming/overview`

to create smart queries in order to drill down on this corpus without actually having it locally available.

As a prerequisite to various techniques presented in this thesis, the Streaming API has been used to collect larger volumes of corpus data. Based on a continuous harvesting over the course of three years (approx. between 2011-2014), a comprehensive archive, comprising more than 6 TB of data, has been established. The database is separated into one dataset that contains almost all geolocated Twitter messages, and one that contains a random sample of 5 to 10 percent of the complete Twitter corpus. How well such streaming API samples represent the actual distribution of content and topics has recently been discussed by Morstatter et al. [2013].

2.3 Situation Awareness

Decision-making is a function that maps data about the current state of environment to actions that manipulate it. The process of extracting information from the data, identifying relevant entities, building a mental model, and deriving hypothesis about future states is called *situation assessment*. The state of knowledge that is based on this model, and that allows to close the transition between data and actions, is called *situation awareness* (SA).

Situation awareness is a key element in the primary application areas of this thesis, which particularly include public safety, critical infrastructure protection, and disaster management, including emergency preparedness, response, and recovery. However, beyond these domains, situation awareness is also an important concept in almost all activities that involve time-critical, dynamic decision-making. The developed techniques and analytics schemes can thus easily be adapted to a range of other domains. Amongst others, they include law enforcement, journalism, traffic control, risk assessment, military command & control, cyber-security, and event planning.

The following section first provides a formal definition of situation awareness. It will then discuss the relevance of social media for achieving situation awareness in disaster management and public safety, highlight how social media is already used in these contexts, and discuss the open challenges that could be tackled by employing visual analytics. It concludes with a brief overview of event-centered SA terminology used in this thesis.

2.3.1 Definition and Model

While there have been abundant attempts to formalize the concept, one of most commonly cited conceptions was proposed by Mica Endsley [1988]. She defines situation awareness as "[...] the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future." In her later work she also introduces first steps towards a theory of situation awareness based on this definition [Endsley, 1995]. She presents a model that relates system and human factors of decision-making, and that integrates SA as a three level process between the given state of environment and the performance of actions. A simplified version of this model is illustrated in Figure 2.5.

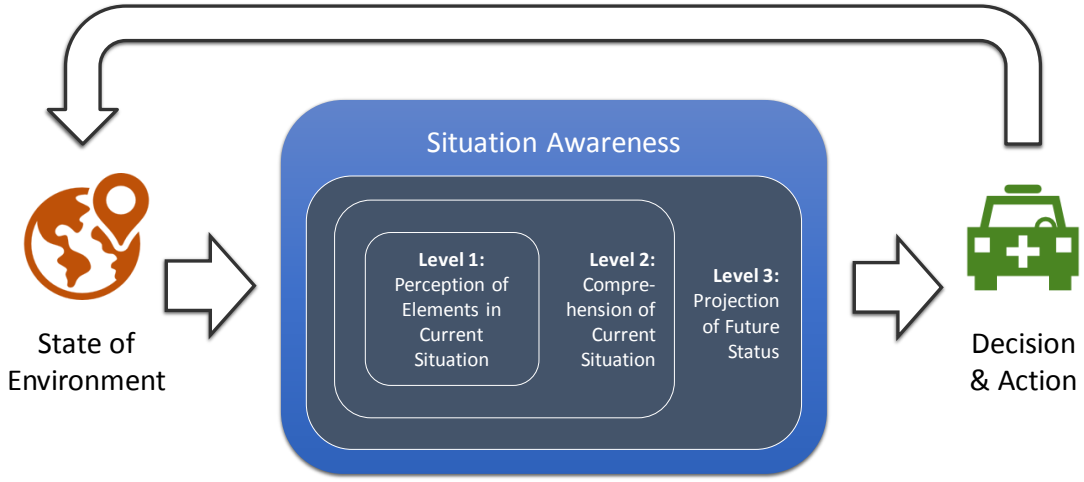


Figure 2.5 — The situation awareness model, adapted from [Endsley, 1995, p. 35]. Between observing the environment and making a decision stand (**Level 1:**) perception of relevant entities, (**Level 2:**) comprehension how they relate to each other, to situation context, and background knowledge, as well as identification of patterns, and (**Level 3:**) Projecting how they will behave in the future and build hypothesis about hidden variables. (Icons by Freepik / CC BY 3.0.)

In the first level of the SA model, decision makers have to identify relevant entities of the environment and their attributes, such as location, direction, velocity, or intention. For example, in case of an earthquake, emergency responders have to identify most severely damaged cities, districts, and buildings, the number of citizens still in danger, the location and destination of response forces in action, and the availability of shelters, evacuation routes, and hospitals.

In the second level of SA, they have to create a holistic picture of the situation by relating these entities with each other, assessing their significance, and recognizing patterns. In the earthquake scenario, this could amount to understanding that multiple response forces have reacted to emergency calls from less affected regions, while, at the same time, the severe damages in other districts might prevent people from requesting help via telephone lines.

Finally, in the third level of SA, decision makers have to project the current situation to possible future states of the environment and form hypothesis about hidden variables. Emergency analysts might realize that several larger hospitals serving as central drop points for the ambulances are also located in the severely damaged areas. As this might result in a significant bottleneck to the rescue operations, they could then try to make inquiries with alternative locations and communicate their availability and capacities to the responders.

The level of situation awareness that can be achieved depends on the availability of data, the necessities of the situation, and the skills and expertise of the analyst. Naturally, decisions are also often made after only achieving level 1 of SA, e.g., perceiving relevant entities without understanding the situation, or level 2, e.g., understanding the current situation but failing to anticipate how it will behave in the future or how various decisions might impact it.

In that vein, the relationship between situation awareness and sensemaking (as introduced in Section 2.1.3) has been elaborated by Klein et al. [2006]. While they see situation awareness to be "[...] about the knowledge state that's achieved—either knowledge of current data elements, or inferences drawn from these data, or predictions that can be made using these inferences"', sensemaking would be "[...] about the process of achieving these kinds of outcomes, the strategies, and the barriers encountered".

From this perspective, sensemaking can be interpreted as an integral part of situation assessment, which especially comes into play if more in-depth intelligence gathering task have to be solved. In some domains, such as firefighting, logical inference and elaborated hypotheses are less frequently a basis for actions. Here the perception and comprehension levels of situation awareness, which can be related to information foraging, are most important and often already sufficient to make the right decisions. In other areas, such as law enforcement, it is more common to relate different observations and form theories before actions can be taken. The relevance of sensemaking to achieve level 2 and 3 situation awareness thus depends on the type of task.

2.3.2 Leveraging Social Media

In the early years of research on situation awareness, the concept has often played a more vivid role in aviation and military contexts, such as air traffic control, piloting, and air-force tactics [Andre et al., 1991; Rodgers et al., 2000; Jones and Endsley, 1996], and it was less prominent in many other areas that exhibit time-critical dynamic decision-making.

A reason can be found in the high availability of sensor information in the former domains and the lack of such means in the latter. Pilots and air traffic controllers have access to a range of powerful instruments, such as traffic radar, ground radar, and radio navigation, that inform them of the position and velocity of all relevant entities in their environment. By contrast, decision makers in other areas mostly have to rely on public phone calls, weather reports, news media, and feedback from field responders, which are often limited, insecure, and sparse information channels. If the decision makers are blind to what happens in remote areas, they cannot even enter the perception level of SA. In these situations, they have to skip that part of the model and deploy their limited resources often at random.

With today's penetration, density, timeliness, and report-style content of social media, data analysts essentially have access to a novel sensor network of their own. It can support them in better grasping the extent, the set of relevant entities, as well as the possible impacts of an ongoing situation. Various studies have thus demonstrated that information taken from social media content can have significant value in crisis analytics. Most notably, Sarah Vieweg et al. have conducted investigations to demonstrate the usefulness and applicability of the data in all stages of situation awareness.

They show, for example, that during the Oklahoma Grassfires[18] and Red River Floods[19] of April 2009 several eyewitnesses tweeted situation reports, such as the number of damaged houses or the severeness of the flames [Vieweg et al., 2010]. In total, they identify 14 distinct categories of situational updates that had been given in Twitter, which include warnings, information about preparatory activity, fire line positions and flood levels, weather information, road conditions, information about evacuation measures, volunteer offers, and general damage/injury reports. They furthermore showed that between 78% and 86% of people that contributed information also provided at least one geolocation with their posts. Since these early accounts, researchers have made quite similar observations for other critical events, such as typhoons, earthquakes, hurricanes, revolutionary and peoples movements, public riots,

[18] `http://www.srh.noaa.gov/oun/?n=events-20090409`

[19] `http://en.wikipedia.org/wiki/2009_Red_River_flood`

influenza pandemics, public events or criminal acts [Vieweg et al., 2008; Hughes and Palen, 2009; Sakaki et al., 2010; Mendoza et al., 2010; Chew and Eysenbach, 2010; Heverin and Zach, 2010; Qu et al., 2011; Starbird and Palen, 2012].

Alerted by such findings, the US Department of Homeland Security (DHS) has recently conducted an investigation how the data could be leveraged in public safety and disaster management [DHS, 2014a]. In this report, they identify various situation assessment activities that could be informed by social media to better support decision-making:

- **Monitoring** - This incorporates active and passive information search based on keywords, geographic location, or content types. Findings that contribute to SA include reports of individuals, unusual spatiotemporal patterns (e.g. lack of noise), trending topics, and overall sentiments.
- **Crowdsourcing** - Here citizens actively participate in information gathering and directly communicate observations, reports, and opinions to the crisis managers. They can do so by directly addressing them in social media, employ specific identifiers (e.g. hashtags), or using specific platforms, such as Ushahidi [Okolloh, 2009].
- **Intelligence** - This comprises the more in-depth analytical efforts in social media that go beyond pure filer-based monitoring. Tasks include verification of information that was previously gathered inside or outside of social media, understanding cascading effects, and investigating context information about an ongoing situation. While social media monitoring is important to establish level 1 situation awareness, social media intelligence thus also helps to additionally inform levels 2 and 3.

With information gathered in these activities, the analyst can decide actions that should be performed or inform supervisors. The report names, among others, damage assessment, hazard prevention, situation prediction, field verification, resource deployment, task prioritization, and logistics planning.

2.3.3 Future Requirements

In various past events, social media has occasionally been used by crisis responders and other stakeholders based on either the platforms' native search interfaces, social media dashboards, or crowdsourcing systems. A survey conducted by MacEachren et al. [2011] among 46 emergency management experts showed that LinkedIn, Twitter, and Facebook are regularly used for professional purposes by 53.6% to 67.9% of the participants. The study also revealed that

at least 39.1% already used the native Twitter search to gather information from the public in crisis management, and that they consider this a very useful feature (4.2 on average on a 5 point Likert [1932] scale). The study furthermore asked the experts about capabilities they would consider most important in future interactive tools. They voted maps, photo/video collections, temporal overviews, tag clouds, and cluster analysis the Top 5 of 13 options.

A report on social media use by various agencies during Hurricane Sandy[20] was also provided by the DHS [2014b]. For example, in New York, social media was monitored by NYC digital, a subsidiary of the mayor's office, and daily reports were generated based on this activity. With regard to existing tool usage in that event, the report lists *Google Maps* and *HootSuite*[21]. The latter is a commercially available social media dashboard that allows to post and search information across multiple services. Also, according to the report, the Red Cross collected more than 2 million entries based on keyword searches for shelter requests or emotional support. From this message corpus they selected about 10k of posts for further evaluation based on their content. In the selection effort, each message was manually investigated by volunteers. Discovered entries included information about disaster areas, comments on Red Cross efforts, and direct inquiries to the organization.

Following their observations, MacEachren et al. and both of the DHS reports enumerate several open challenges and gaps in technology that should be addressed to enhance information extraction and to fully enable utilization of the data in future crises. The requirements related to tool development can be categorized with regard to the issues of data access, context, complexity, and management as introduced in Chapter 1:

Challenges of Data Access:

- *Difficulties associated with throttling of social media streams, including limitations on the retrieval of data through APIs* [DHS, 2014a, p. 32]
- *Twitter bandwidth and update limits* [DHS, 2014b, p. 30]
- *[...] volume and access to information* [DHS, 2014a, p. 24]

Challenges of Data Context:

- *The ability to know an urgent event occurred, identify trending topics within a geographic location* [DHS, 2014a, p. 24]

[20] `http://en.wikipedia.org/wiki/Hurricane_Sandy`
[21] `https://hootsuite.com/`

- *Display emerging Twitter hashtags by topic area, bounded by geographic location* [DHS, 2014a, p. 24]
- *Method of aggregation* [DHS, 2014a, p. 24]
- *The ability to determine the severity of a specific situation through imagery and text* [DHS, 2014a, p. 24]
- *[...] limited tools to extract meaning from the generally ill-structured and cryptic formats* [MacEachren et al., 2011, p. 1]

Challenges of Data Complexity:

- *Discoverability of information, resources, and efforts* [DHS, 2014b, p. 29]
- *The ability to drill down into specific trending subject areas* [DHS, 2014a, p. 24]
- *The ability to filter urgent requests for assistance from a large volume of social media information* [DHS, 2014a, p. 24]
- *[...] limited means to sort relevant from irrelevant information* [MacEachren et al., 2011, p. 1]

Challenges of Data Management:

- *[...] large volumes of information* [MacEachren et al., 2011, p. 1]
- *Applicable search parameters* [DHS, 2014a, p. 30]
- *Update frequency* [DHS, 2014a, p. 30]
- *Storage, including server capacity, privacy and security considerations* [DHS, 2014a, p. 30]

These requirements were most strongly highlighted by the reports. Therefore, they also served as primary inspiration to inform requirements for the design of the toolset and overarching analytics model presented in this thesis.

2.4 The Social Media Analytics Model

So far, this chapter defined the data source, which is social media, the challenge, which is achieving situation awareness based on this data, and the methodology to tackle the challenge, which is visual analytics. The primary contribution of this thesis is to bring these three subjects together by establishing a social media

analytics model. This section briefly outlines the individual methods that have been developed to tackle the requirements. It then provides a more formal introduction to the overarching approach and defines how these components relate to each other. It concludes with a review of existing visual and non-visual approaches that addressed the challenge before.

2.4.1 Addressing the Open Challenges

Visual analytics can be useful at multiple stages of situation awareness. It was already indicated that the most important leverage points in emergency and crisis response naturally are the perception and comprehension levels of SA. So far, analysts had to rely on limited and sparse information channels to assess ongoing situations. Based on visual analytics of social media, their efforts can be enhanced with a novel and comprehensive sensor instrument.

In the first level of situation awareness, analysts could use visual tools to decide how relevant social media entries can be monitored and pre-filtered from the APIs. Afterwards they can apply these operations and read organized or aggregated representations of the individual messages to identify relevant locations, individuals, institutions, infrastructures, damages, injuries, and possible threats. By providing means to create, label, and organize visual representations of these observations or detected patterns, a system can furthermore help to relate them with each other, with background knowledge, and with context information.

This can foster comprehension in the second level of situation awareness. In this stage, analysts can also use directed search to additionally inform their situational picture based on observations, comments, and insider information provided in the services. Finally, a visual analytics system can even help to formulate and validate hypotheses in the third level of situation awareness by providing means to represent and organize labeled message collections. Analysts can then understand how the situation further evolves and also come up with ways to enhance their social media monitoring configuration.

To facilitate these possibilities, the methods of this thesis have been devised to address the four requirement categories identified from the domain expert reports. For each of the requirements, a prototypical approach is presented based on visual analytics methodology:

- **Data Access** $\rightarrow$ **Query Optimization** - Primary issues of data access are existing rate, update, and throughput limitations of the service APIs. This applies to both, the public search APIs and the streaming APIs. The former usually allow a limited number of requests per time interval and

also deliver a limited number of results per request. The latter only allow limited number of connections with more relaxed, yet existing throughput limitations. In the near future, the artificial limitations might be replaced by external data providers that charge clients for data access. However, this does not solve the problem for many analysts. The costs for larger data volumes are significant and the necessary resources are often not available.[22]

Chapter 3 tackles the challenge by means of iterative query optimization and hierarchical sample-set exploration. Based on an initial seed query, analysts can pick a representative subset of the data and explore it in a highly interactive visual space. They are provided with automated means to optimize the current query, which then serves as basis to collect the next sample. The final outcome will be a better understanding of the data corpus that hides behind web APIs (e.g. what is available, how should queries be built), and final request parameters that can be used to continuously collect larger data volumes via the streaming APIs.

- **Data Context → Visual Event Discovery** - Once pre-filtered data volumes have been collected, analysts often need an overview and entry point for further investigations. We have seen that analysts ask for means to indicate urgent events, identify emerging topics in geographic areas, to aggregate information, and to understand the severity of situations. Often analysts might initially have no clue what to search for, as the nature of an ongoing situation is unclear or it is even unknown that something relevant has happened.

  Chapter 4 presents a technique that automatically identifies anomalies in message streams and archived data based on cluster analysis. It then allows discovery of events based on a geographic overview visualization. As there can be a large number of such anomalies, and the system cannot automatically decide what is semantically relevant, the visualization turns overfitting of the heuristic clustering into a feature by conveying a large number of topics through zoom-adaptive aggregation. An add-on to this technique is furthermore presented that highlights statistically unusual anomalies by color coding.

- **Data Complexity → Task-adaptive Monitoring** - Once analysts have achieved an initial overview, they have to proceed with drill-down or transition into ongoing situation monitoring. Correspondingly, challenges

[22] See [Wagner, 2014] for a discussion on pricing at Datasift and Gnip. The services provide filtered access to various social media services.

in the data complexity category were characterized by necessary means to detect requested information, to investigate specific subject areas, and to continuously filter according to task-related information need.

Chapter 5 thus addresses means to filter relevant information items by user-defined classifiers. A tool for visual interactive classifier training helps analysts to generate powerful task-adaptive filters based on data from well-understood previous events. Using a visual pipes-and-filters-metaphor, analysts can then quickly orchestrate and combine basic filters to enable drill-down into sub-events or continuous monitoring of ongoing situations. The tool furthermore supports the labeling and set-based combination of topic-related message collections. It can thus additionally be used to represent and validate hypotheses.

- **Data Management → Integrated VA Platform** - Even if data is pre-filtered and collected from throughput-limited channels, it can still produce significant volumes of messages that have to be processed and managed. For example, by requesting all geolocated messages from Twitter, one currently has to handle about 10 million messages per day. This requires sophisticated solutions of storing and indexing the data to allow fluent requests, application of multiple coordinated visualizations, and highly interactive combination of versatile tools.

  Chapter 6 thus presents an introduction to *ScatterBlogs*, a visual analytics platform that was developed in the course of this thesis in order to prototypically implement, integrate, and evaluate all of the aforementioned techniques. Based on a custom-designed data storage, management, and processing solution, it allows to handle several millions of messages per day in real-time. It furthermore provides additional tools to visually explore the data and to create ad-hoc filters. In addition to real-time processing, several years worth of messages can be re-examined with all tools in post-analysis if needed.

2.4.2 Model

All presented techniques are designed to integrate with each other in order to address the grand challenge of achieving comprehensive situation awareness. To this end, each technique comprises some form of self-contained iterative analytics loop that solves the specific task, while all processes together are embedded in an overarching analytics cycle. The following chapters will successively highlight how the techniques with different capabilities as well as limitations relate to and complement each other. Based on these relationships,

they can be combined in a larger insight generation cycle, as shown in Figure 2.6. Data streams between components are highlighted in gray. Requests, interactions, and visual information flow are shown in white.

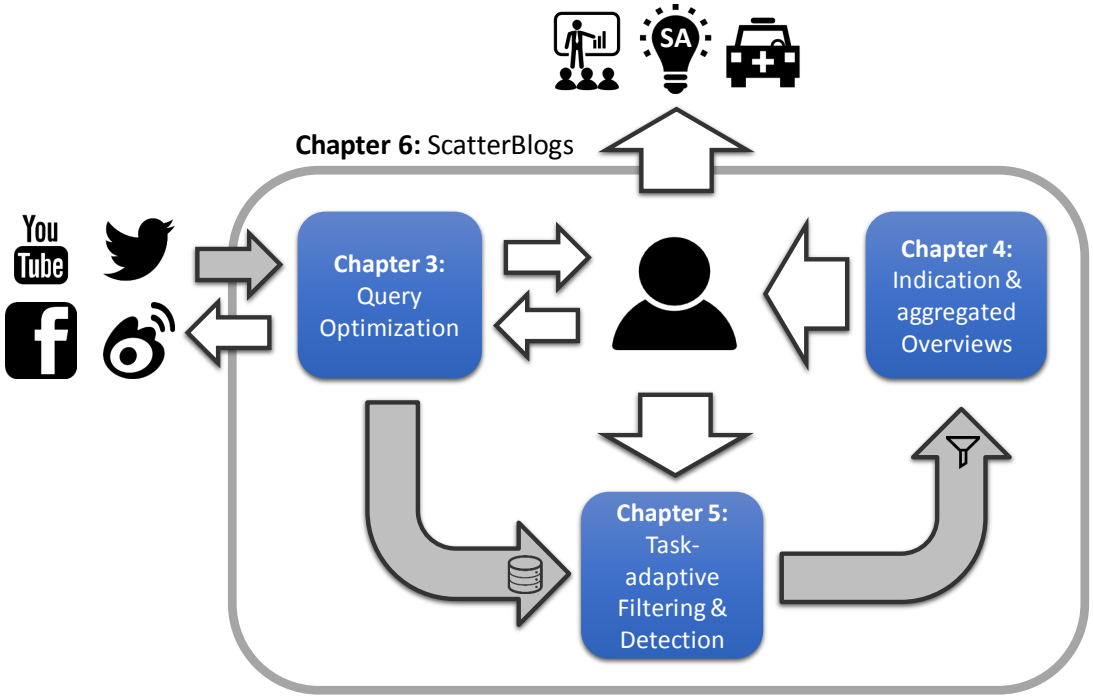


Figure 2.6 — The social media analytics model (white arrows = interaction/request flow; gray arrows = data flow). The analyst creates optimized queries, monitors and filters them based on tasks, and investigates the situation using aggregated overviews and indication. The result of the process is situation awareness, which can inform reporting and decision-making. (Icons by Freepik / CC BY 3.0.)

The model describes the following workflow: Based on user interaction with the *query optimization* component, the available data behind the web APIs is initially explored to better understand the extent of available data and to generate well-formulated queries. Initial data streams are then defined and continuously collected from the service APIs. That stream now first passes the *task-adaptive filter* components, where it is either already filtered based on a predefined monitoring configuration, or where it just passes through to allow unfiltered initial overview and exploration. The *visual event indication* component can then be activated to show possible events and unusual topics in the data. The analysts interact with it through zooming, panning, and selection, which highlights corresponding messages in all views. Based on this initial situation

assessment, they can move back to the filter component to further narrow the data stream, drill down on events, prepare the detection of possibly relevant information, or model the situation by arranging labeled message collections. The filtered data is then again presented through overviews and indication, and it can be explored further.

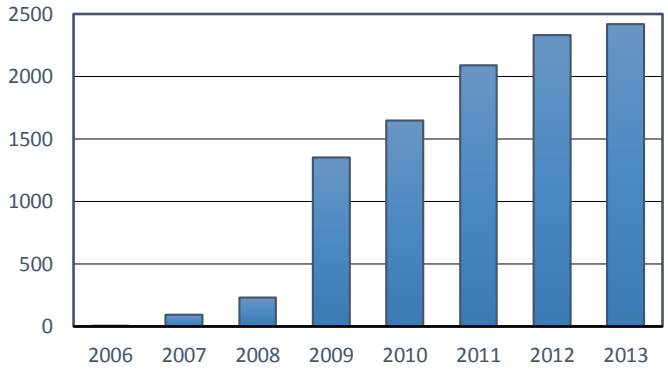
If the situation demands for a shift in focus or the collection of additional data, the analysts can turn back to the query optimization stage to harvest new samples or re-configure the data stream collection. The three stages of the larger analytics cycle are thus closely related to the shoebox, filtered evidence files, and schema of the model of Pirolli et al. (See Section 2.1.3). It is furthermore important to note that the query optimization tries to maximize recall and prevent the analysts from missing anything important. The task-adaptive filters and exploration tools subsequently optimize precision in order to find the most relevant items in the already collected data.

2.4.3 State of the Art

Research on social media analysis has seen significant growth in the last couple of years, as illustrated in Figure 2.7. Besides visualization research, the topic has received much attention in plain NLP and machine learning, where statistical methods for document retrieval, classification, and topic modeling for short and highly context-dependent snippets have been developed. For example, Weng et al. [2010] use LDA-based topic modeling to find important information related to ongoing situations, Zhao et al. [2011] establish a connection between social and traditional news media by summarizing and categorizing tweets also based on topic models, and Asur and Huberman [2010] employ linear regression to predict the success of box office movies based on Twitter. Supervised classification methods have prominently been used by Go et al. [2009] to assess the polarity of social media messages. They evaluate Naive Bayes, Maximum Entropy, and SVMs and conclude that sentiment of messages can automatically be detected with 80% accuracy.

Statistical models, however, can only be applied to data that is locally accessible to the analyst. All of these approaches thus rely on a pre-extracted corpus or use predefined queries to collect streaming data. If relevant data is not covered by the corpus or the query in the first place, it will also bypass the model.

The works in the information visualization and visual analytics domain can broadly be categorized in four application areas: Information diffusion analysis, sentiment and opinion mining, debate and news media intelligence, and situational awareness. Recent approaches primarily use topic modeling, classifi-



◄ **Figure 2.7 —** The chart shows (non-accumulative) numbers of Twitter-related Google Scholar search results since Twitter was founded in 2006. Only results containing the keyword `Twitter` in the title are counted.

cation, and entity recognition to find, filter, categorize and aggregate relevant microdocuments in interactive visualizations.

Interactive analytics systems have been shown by various researchers. Abel et al. [2012] present Twitcident, a web-based system that builds on facetted search and semantic entity extraction to retrieve relevant messages connected to emergency communication. Twitinfo [Marcus et al., 2011] automatically identifies and labels unusual bursts in Twitter streams. It shows the data based on a temporal overview and visually annotates the bursts. Leadline [Dou et al., 2012] and HierarchicalTopics [Dou et al., 2013] employ LDA analysis to separate and investigate topic streams in Twitter. Additionally, Leadline connects events in the message stream with recognized entities and HierarchicalTopics allows to explore and manually re-organize the inherent topic hierarchy. Whisper [Cao et al., 2012] visualizes geosocial information diffusion based on an interactive sunflower metaphor. Following their observations from the previously cited study, MacEachren et al. [2011] also show a prototype system, called SensePlace2, that allows for querying Twitter and depicting aggregated results on a map. The places are determined by employing named entity recognition and reverse geocoding, resolving strings to geolocations. White and Roth [2010] present TwitterHitter, suggesting geospatial analysis of Twitter messages as a suitable means for addressing different tasks in criminal investigations. More recently, Zhao et al. [2014] showed #FluxFlow, a system that helps to reveal anomalous information spreading based on multidimensional scaling, hierarchical cluster analysis, and an interface of interactively coordinated visual tools.

However, in contrast to this thesis, none of the existing works presents an integrated model that addresses all necessary stages of retrieving, processing, filtering, and contextualizing information to achieve complete situation awareness from social media. All techniques shown in this thesis have a particular focus on scalability and are, without exception, designed to enable real-time

processing of streaming data. Both aspects are rather underrepresented in previous research as it frequently focused on post-analysis of self-contained archives with manageable sizes.

The challenge of efficiently collecting new data from rate-limited web APIs is also an underestimated aspect, and it has not been addressed so far. The visual event discovery scheme presented in this thesis stands out in terms of its novel cluster analysis technique and the way of merging the distinct characteristics of visual and automated means to achieve streaming enabled aggregated overviews. Moreover, most existing approaches only concentrate on the identification and analysis of high-frequency events. While this can lead to interesting findings with respect to ongoing discussions, the information such messages convey is often *second-hand* and can be found by other means as well. The method presented in Chapter 5, which combines interactive analysis with interactive classifier training to enable task-specific monitoring, is one of the first takes on detecting individual relevant information items instead of just overarching trends.

Finally, although many of the previous approaches feature sophisticated user interfaces, none of them provided a solution that combines powerful collection, exploration, and adaptive filter under one umbrella in order to establish a tightly integrated, highly interactive, and scalable software system for exploratory analytics and comprehensive evaluation of techniques.

CHAPTER 3

Query Optimization

This chapter deals with a basic question of information retrieval that most people have already asked themselves when using web search like Google or Bing: "What is the most effective keyword query for my information need?" In the next chapters, 4 and 5, tools and techniques are described that optimize the semi-automated discovery of relevant information entities in a given corpus or stream of social media data by means of automated aggregation, statistical anomaly detection, and message classifiers. These methods, however, assume that the information entities are or will be present in some fully accessible database and just have to be identified between the irrelevant data that is also stored there. Thus, prior to their application, the data has to be collected from the search and streaming APIs of the platforms.

As described in Section 2.2.3, these APIs are usually limited with regards to the number of calls that can be made and the number of results that will be delivered for each call. It will thus often be impossible to download the complete set of messages relevant to an investigation. As a result, users frequently have to work with subsets of the actual document corpus and they have to create smart requests in order to drill down on this corpus without actually having it locally available. Although a full take of data can sometimes be bought from the services, even many companies and government institutions lack resources to make arrangements with all possibly relevant platforms or to provide the necessary hardware to process such volumes of data. As long as the APIs do not accept descriptions of classifiers or clustering algorithms as input parameters to execute them for the client, there is a need to generate highly optimized queries

and combine them with meta-data restrictions to pre-filter the data in the best possible way.

In this chapter, a method is presented that addresses the challenge based on incremental query advancement. By iteratively exploring samples from the APIs in a hierarchical cluster visualization, analysts can improve their understanding of the data and are supported in generating queries that narrow to their information need. The chapter first describes the specific challenges of microdocument retrieval, which can be distinguished from traditional web search. The visual technique devised to tackle these challenges is then illustrated based on a prototypical implementation called *TreeQueST* (A **Tree**map-based **Que**ry **S**andboxing **T**ool) (Section 3.2). Following this brief overview of the approach, the chapter provides necessary background information on the areas of information retrieval and hierarchical information spaces (Section 3.3). More technical details on the hierarchical cluster visualization, which is based on the notion of *tweet similarity*, will then be given in Section 3.4. Here it will also be described how the hierarchical structure inherently facilitates interactive exploration. In Section 3.5 the actual algorithm for automated query creation based on user-selected topics from the hierarchy is introduced. The chapter concludes with a case study that illustrates how the method can be applied to Twitter data in an actual media intelligence scenario.

Major parts of this chapter have previously been published in:

- D. Thom and T. Ertl. TreeQueST: A treemap-based query sandbox for microdocument retrieval. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1714–1723. IEEE Computer Society, 2015

3.1 Microdocument Retrieval Challenges

Websites, news articles, blogs, and other forms of traditional web documents are often self-contained and provide comprehensive and broad information on a specific subject. The primary topic of websites can usually be derived based on word frequencies, and hyperlink-networks can be used to assess their importance and centrality to a topic. Accordingly, web search engines are known to be powerful tools for the retrieval of these documents, as the relevant information for a well-formulated query is usually contained within the top-ranked results.

By contrast, the information shared in a single social media message is often limited, very specific, and highly context-dependent. And the available search

engines for messages and microdocuments, which are most often provided by the platforms themselves, frequently fail to fulfill an information need due to several reasons:

- Complete information about a topic of interest is often not contained in a small set of comprehensive documents but distributed over hundreds of messages. In this case, various users provide small context-dependent snippets that would add up to the complete picture.
- The decision to reference other messages is frequently based on the popularity of the author and not on the relevance of the message. Ranked results based on link-centrality (e.g. re-tweet count) can thus fail to retrieve relevant content from ordinary users.
- The low effort to share information leads to highly redundant message contents for popular topics. Top-ranked results can thus be dominated by repeated information that is not relevant to the analyst.
- The shortness of the texts results in a lower chance that given query words are contained in a message. The analyst thus has to provide a more comprehensive list of keywords to cover all possibly relevant entries.

The method introduced in this chapter tries to tackle these challenges by means of exploratory query optimization. It follows an analytical loop that employs hierarchical clustering and a highly interactive treemap representation of summarized clusters. In an iterative process, analysts are supported to continuously improve their understanding of the inherent topic structure of a retrieval set, find relevant discussions and events, and build a textual query that fits their specific information needs.

The approach bears strong similarities to Scatter/Gather [Cutting et al., 1992], a document browsing technique that organizes unknown corpora using cluster analysis and shows descriptive textual summaries to the user. Based on these summaries, groups of interesting topics can be selected, which then serve as new seed set for re-iterated clustering. In past research it has been shown that the method is more efficient but not more effective compared to keyword-based search engines. Although it was demonstrated that a user's overall understanding of an unknown document collection is increased, the return of investment for traditional web search has been too low to popularize the method outside of research. This chapter, however, demonstrates that the basic principle perfectly matches the characteristic challenges of microdocument retrieval and query optimization as described above.

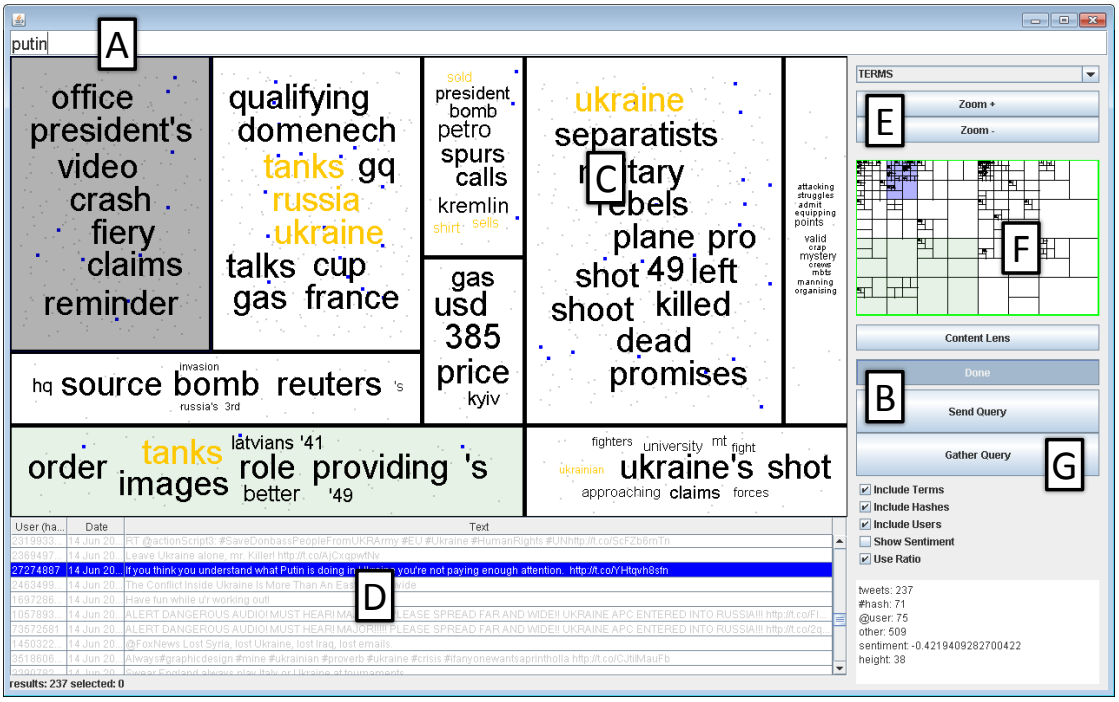


Figure 3.1 — Overview of the *TreeQueST* UI after performing a seed query and subsequently testing a different keyword on the result set. It shows A) the search bar, B) the 'Send Query'-Button, C) the generated topic hierarchy, D) individual tweets of a selected node, E) buttons to zoom-in and -out along branches, F) a minimap of the complete hierarchy and G) the button to create a new query from selected topics. Tweets and tags covered by the test query 'putin' are highlighted in blue and orange. © 2014 IEEE

3.2 The *TreeQueST* Exploration Approach

In the traditional Scatter/Gather scheme of Cutting et al. [1992], the documents of a corpus are first grouped into disjoint subsets using some cluster analysis method like K-Means or DBSCAN. The users are then presented with an aggregated representation, also called a *digest*, for each of these subsets, which would usually consist of a list of the most frequently used words of the cluster and representative titles or sentences from the documents that are most central to the cluster. The users select one or more of the digests that seem relevant to them (*gather*), and the process is repeated with the union of the selected document groups as new initial set (*scatter*). The users proceed until they reach a good understanding of the available data and/or find the documents they are interested in. The *TreeQueST* approach follows that same basic scheme but

adds another step, namely the *query*, between the gather and the scatter phase to incorporate data that is just remotely accessible. It furthermore improves the simple digests by hierarchical means of visual data exploration. In this section, the prototypical *TreeQueST* UI is first introduced to illustrate how Scatter/Gather can be adapted to the domain. It will then be further described how the query step is integrated in that process.

3.2.1 User Interface and Exploration Process

The retrieval process is started by entering initial keywords into the search box of *TreeQueST* (Figure 3.1.A). This query can contain Boolean operators like `OR`, `AND` and `NOT`, and, if Twitter is used as the data source, one can also enter `#` and `@` to specifically indicate hashtags and usermentions. By clicking on the *Send Query*-button (3.1.B), a request is send to the API that returns a fixed maximum number tweets ranked by relevance. The incoming tweets are preprocessed and hierarchically organized in a binary tree using agglomerative clustering and a newly defined tweet similarity measure (see Section 3.4.2). In contrast to the traditional Scatter/Gather approach, a visual representation of this hierarchy is then shown to the users (3.1.C) that serves as an information management space and the primary playground for all further interactions. Further details on the design of this visualization will be given in Section 3.4.3. Individual tweets within the hierarchy are shown based on a spatialization that places similar tweets more closely to each other. In addition, the contents of each area, also called a *topic*, are represented as a weighted tag cloud of terms and/or hashtags that are most specifically relevant to the tweets in this area.

Based on an initial segmentation, users can freely explore the hierarchy by changing the granularity of topics using the mouse wheel, select nodes to read individual tweets in a table (3.1.D), and zoom into the hierarchy to drill down on topics they consider interesting (3.1.E). During this process, they are supposed to get familiar with the data and gather relevant topic areas, which is done by by right-clicking on them. The gathered topic areas are highlighted in pale green. The navigation is supported by a minimap of the complete tree (Figure 3.1.F). It illustrates which node is currently selected (blue area), what viewport is currently zoomed-in (green frame), and which topics have been gathered (pale green fields). Once a relevant subset of topics is gathered, the users can proceed with the query step by clicking the button *Gather Query* (3.1.G), which automatically generates a recommended query from the selected topics.

3.2.2 Query Creation and Evaluation

As the users of *TreeQueST* are usually just working with a sample of the complete microdocument corpus, the query creation is an integral part throughout the exploration process. It essentially allows to extend the scatter step to the part of the data that is "hidden" behind the rate-limited APIs. To this end, the algorithmic query creation tries to find characteristic features of selected topics compared to the non-selected ones. The query is then generated from these features, as will be explained in Section 3.5.

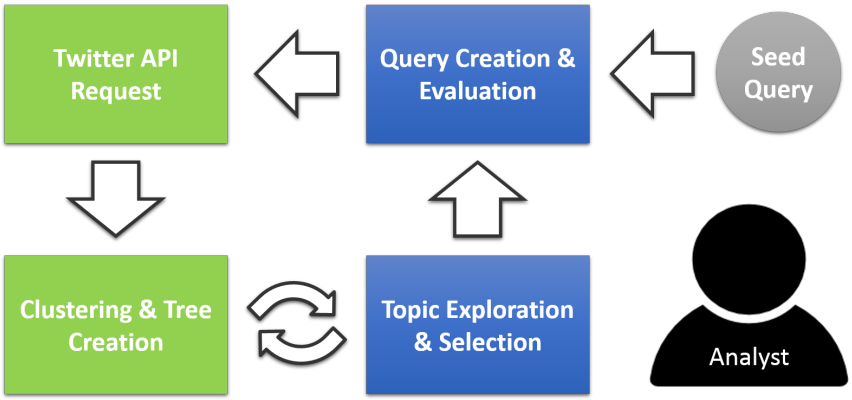


Figure 3.2 — The diagram illustrates the core exploration loop of *TreeQueST* . Elements of the interaction phase are in blue, data extraction and processing steps are in green. After creating an initial seed query, a maximum number of tweets is retrieved by a *Search API* request. The tweets are hierarchically clustered and the structure is used to create a space-filling tree. Based on this visualization, users can explore the topic hierarchy and gather topics. Using the selected topics, a new query is generated, which can then again be evaluated and modified by the user.

Once a generated or a manually created query is entered into the search bar, users can initially evaluate its effects by hitting the enter key before actually executing the query via the API. Tweets in the sample set which are covered by that query will then be highlighted in blue, and all other tweets will be displayed in gray. This can be seen for the test query `putin` that was entered in Figure 3.1. Furthermore, topic tags that are highly co-occurrent to the current query are also highlighted (orange tags in Figure 3.1). Users can use the highlights to get a feeling for the portion and content of tweets that would be retrieved from the corpus and investigate how well the individual topics in the hierarchy would be covered. They can then immediately modify the query by deleting or adding keywords and by exploring the results.

After executing the query, the results of the API request are combined with the messages from the gathered topics in a new sample set, which is then again clustered and visualized. The complete exploration loop is illustrated in Figure 3.2. The query step can also be bypassed by the users if they click on *Gather Query* with an empty search bar. Similar to traditional Scatter/Gather, only the gathered topic messages will then be re-clustered without incorporating new retrieval results.

Ultimately, the outcome of the exploration process is threefold. First of all, by working with the document hierarchy, users get a better understanding of the available subtopics, their interconnections, and individual document contents. Second, based on their research, they can evolve an optimized query that better fits their interest areas in terms of precision and recall. Third, during the process, they can retrieve a range of relevant documents and topics that establish an information space which organizes the domain of interest. The final optimized query can be used to continuously collect relevant documents from the Search and/or Streaming APIs of the service. Additionally, the users will also have acquired a thorough understanding of what to expect from the resulting data.

3.3 Background

The *TreeQueST* approach can be located at the crossroads of information retrieval, the use of visual information spaces, and recent research on microdocument analysis in natural language processing (NLP), data mining, and visual analytics. As the more general research on social media analytics has already been covered by the *state of the art* discussion (Section 2.4.3), this section focuses on the evolution of Scatter/Gather and its derivatives as well as related works on space-filling visualizations and hierarchical information retrieval.

3.3.1 Evolution of Scatter/Gather

Scatter/Gather, as a method for information retrieval, has first been popularized by Cutting et al. in their work "Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections" [Cutting et al., 1992]. Their method relies on partitional clustering and builds cluster representations based on most frequent words and representative documents.

As computing power was limited at that time, dynamic hierarchical clustering solutions, which are used in *TreeQueST*, were not an option in their research. However, later they presented a method to precompute the hierarchy and use this as a basis to achieve constant interaction-time [Cutting et al., 1993]. The

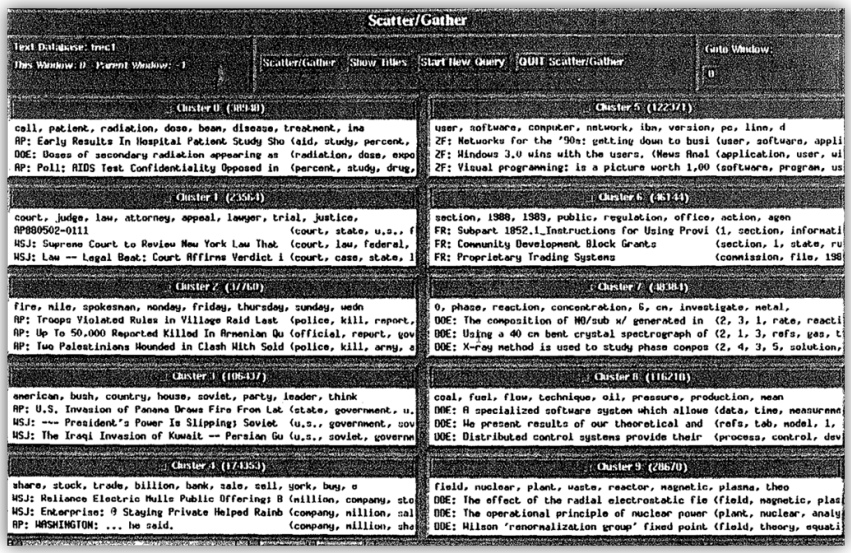


Figure 3.3 — Windowed display of cluster digests for Scatter/Gather, as presented by [Pirolli et al., 1996, p. 214]. © 1996 ACM

visualizations of these early approaches were usually plaintext lists of words and document titles or a windowed document browser. This was shown by Hearst and Pedersen [1996], where the method is applied to retrieval results, and by Pirolli et al. [1996] (Figure 3.3). Pirolli et al. also provided results of a user study, which demonstrate that Scatter/Gather helps participants to build a mental image of a given text collection, formulate richer keyword queries, and get a feeling for the portion of relevant documents in the corpus. However, their study also showed that the method is not superior in effectiveness compared to plain keyword search if the goal is just to locate specific documents. The *TreeQueST* approach revisits that discussion, and draws on the hypothesis that the latter is only true for traditional text documents, like articles, books, or websites. In the following sections, it is shown that advanced visual aggregations and techniques for query-based exploration and drill-down can be of great help, both to better understand the available information and to identify key documents and topics.

More recent results have been provided by Ke et al. [2009], who propose a novel partitional clustering algorithm as basis for Scatter/Gather, called LAIR2. They conducted a study and found that the system can be helpful in some situations but not in others. Scatter/Gather for modern web search has also been evaluated by Gong et al. [2012], who performed a user study with 24 participants and gave the users more influence on interactively changing clustering parameters. They

found that some users have difficulties applying the more complex system in a helpful way, while others endorsed the form of interactively aggregated and organized result presentations compared to traditional search engine results. However, in contrast to the *TreeQueST* approach, neither short texts, a visual representation of the topic hierarchy, nor query optimization were a focus of these works.

3.3.2 Visual Information Spaces

Space-filling tree-visualizations have been popular since Johnson and Shneiderman introduced the concept of treemaps in 1991. *TreeQueST* also employs the simple "slice and dice" layout that was proposed in that work. Since then, there have been several new ideas and advances over that original concept, particularly to optimize aspect ratio, node containment, and ordering [e.g. Bruls et al., 2000; Bederson et al., 2002]. However, it has also been argued by Blanch and Lecolinet [2007] that interactive exploration plays the most important role in making complex datasets accessible through this kind of visualization. In their work, they thoroughly examine the notion of zoomable treemaps based on three different interactions: *In Depth Navigation*, *In Breadth Navigation* and *Direct Node Selection*. They allow the user to drill down along a branch, move from a node to its neighbors on the same level, and select arbitrary nodes using gestures. *TreeQueST* provides similar means for zooming. In addition, the user can manually change the display level of nodes using the mouse wheel to select, compare and navigate between them. Furthermore, *TreeQueST* generates summarizations from the topic hierarchy and provides a minimap for orientation. Another approach was proposed by Wills [1998], who also examines the use of interactive treemaps to view hierarchical clustering results. In this work he also introduces a means to represent individual data items in a spatialization generated directly from the hierarchical layout. However, exploration by zooming is not possible in his system and the complete hierarchy is always shown, which makes it difficult to navigate in deeper levels.

3.3.3 Hierarchical Information Retrieval

There are few approaches that address the use of hierarchical clustering and space-filling visualizations to support information retrieval. In this regard, the FISPA-system from Turetken and Sharda [2004] can be considered most similar to *TreeQueST*. They use a treemap visualization based on hierarchical clustering to organize web retrieval results. However, in contrast to *TreeQueST*, they do not address microdocument retrieval and provide no means for sophisticated topic representation, change of granularity or query exploration. Traditional

web searches, as already indicated, often deliver all relevant results within the top-ranked documents. Therefore, although a slight improvement in efficiency was achieved, the participants of a user study conducted with FISPA were not more effective in finding relevant results.

HierarchicalTopics from Dou et al. [2013] is most similar to *TreeQueST* in terms of hierarchical topic exploration, and they also address microdocuments as data source. Dou et al. employ an algorithm, called Topic Rose Tree, to create a hierarchy of LDA-extracted topics. An interactive visualization represents the topics and their temporal evolution as part of this hierarchy, and the users can manipulate it based on their mental model. In contrast to their approach, means to gradually explore the topic hierarchy to overcome shortcomings of automated cluster analysis is the primary focus of *TreeQueST*. Moreover, the visual topic hierarchy primarily serves as a basis to generate optimized queries from selected topics.

3.4 Treemap Cluster Visualization

This section describes in more detail how messages are organized and represented in the hierarchical information space of *TreeQueST*. Agglomerative clustering [Florek et al., 1951] is used to find groups of related elements that constitute the topics in the visualization. As the technique requires a distance or similarity metric between data objects, the closeness of two social media messages has to be characterized based on their textual content and extracted meta-data. This section gives an example how such a metric can be defined. It is furthermore explained how the resulting cluster hierarchy is used to visually represent the dataset as hierarchical topics, and how the process is integrated with the interactive workflow.

3.4.1 Agglomerative Clustering

The goal of agglomerative clustering is to extract a nested hierarchy of similarity groups from given data. Starting with the complete set of data objects - in our case the resulting set of messages from a social media API request - the algorithm initially considers each element as an individual cluster. Based on a predefined distance function between elements, the method then repeatedly finds the two clusters with minimum distance and merges them into one larger cluster. For clusters containing more than one element, this distance is derived from the pairwise element distances using one of several existing methods. *TreeQueST* uses either the average of pairwise distances or the *Ward* method [Ward Jr, 1963] in the default configuration. The clustering terminates

when all elements have been merged into a single, large cluster. During the process, the algorithm builds a binary tree of all merges that have been performed, which is basically the final output of the method. This tree then reflects the similarity structure of the set, such that the subtrees of nodes near the root usually have a high pairwise distance that becomes smaller when moving towards the leaves.

The *dendrogram* of this process highlights the individual merge operations as well as the distances between merged clusters (see Figure 3.4). In this diagram, the data objects and cluster nodes are drawn along the horizontal axis. The vertical axis is used to display merge distances. For each performed merge, a horizontal line is drawn at the corresponding cluster distance and the merged clusters are connected to it with vertical lines. The figure also contains a split threshold and highlighted topic clusters, which will be discussed in the later sections.

It is important to note that hierarchical clustering is not an algorithm with superior speed compared to partitional clustering. The computational complexity of a naive implementation can be estimated at $\mathcal{O}(n^3)$, where n is the number of data objects. However, as *TreeQueST* usually just works with API-samples of limited size, this drawback is not a major issue in the implementation. Also, in contrast to partitional algorithms, the complete hierarchy of possible partitions is provided, and it is not necessary to define a number of expected clusters. Therefore, the structure is particularly suited for interactive exploration. *TreeQueST* employs a slightly modified Weka instance [Frank et al., 2010] to perform the clustering. It provides a relatively fast implementation based on priority queues.

3.4.2 Message Similarity

Although agglomerative clustering is a quite simple method, the actual challenge is to define and implement a metric that well estimates the similarity distance of two messages. The most popular measures for document similarity are several forms of edit distance, which work well on shorter strings with few variations, and cosine similarity, which is well suited for documents of arbitrary size. The latter considers documents similar if they frequently contain the same entities [cf. Ghosh and Strehl, 2006]. Although social media messages are usually very short documents, they are often already considered to be of similar content if they share a certain amount of nouns or other low-frequent words. This makes cosine similarity a generally reasonable choice.

The distance function of *TreeQueST* employs custom defined document feature vectors. In addition to plaintext elements, messages from Twitter and other

sorts of microdocuments provide a range of meta-features that can be used to more precisely assess their similarity. *TreeQueST* thus extracts the following features from a message:

- **Hashtags** - $\mathbf{f}_1$ - The Twitter community has established `#` as a prefix for unique names that indicate a common topic. This has been widely accepted and recognized throughout. Some time ago, Twitter began to automatically detect and link such annotations and to make them accessible through the API. As a hashtag is usually a clear statement of the author, telling us that the message is related to some well-known topic, it is a good indicator of its subject matter.

- **URLs** - $\mathbf{f}_2$ - Many social media message contain a URL as an invitation to its readers to visit the corresponding website or to make a comment about its content. Mentioned URLs are thus another powerful feature to determine the topic of a message.

- **Usernames** - $\mathbf{f}_3$ - Similar to hashtags, the `@`-sign is used as a prefix to highlight usernames in messages, either to indicate the recipient of the message, when used at the beginning, or as a reference to that user, when used in the middle. As users can often be assigned to the discussion around a specific topic, usermentions as well as the username of a message's author are used as additional features.

- **Terms** - $\mathbf{f}_4$ - After removing hashtags, usernames, URLs and punctuation, the rest of the textual content of the message is tokenized, lower-cased, and lemmatization and stemming are performed. The resulting tokens are then also used as features.

Low-frequent words, such as `ukraine` or `putin`, are often of higher specific relevance to a document's subject matter than high-frequent words, such as `the`, `and`, or `word`. It is therefore common to use the *inverse document frequency* (*idf*) to assess how "unusual" a term is.[1] The *idf* is usually applied in conjunction with the the *term frequency* (*tf*) of a word within the given document. However, because of the shortness of microdocuments, words rarely appear more than once in a given message. For the sake of simplicity it is thus assumed in the following definition that the *tf* for a given word and message is either 0 or 1. A sophisticated dictionary of *idf* values, which can be built by processing large samples of unfiltered social media messages (see Section 2.2.3), is supposed to be available as a prerequisite for *TreeQueST* . On this basis the feature vectors

[1] See Section 4.4.1 for a more detailed explanation of this measure.

$\mathbf{f}_1(t), \ldots, \mathbf{f}_4(t)$ of a given message m can be extracted such that an element will be either 0 if m lacks a certain hashtag/URL/username/term, and the *idf* value of that hashtag/URL/username/term otherwise. Based on these feature vectors, the distance between two messages m_1, m_2 is defined as follows:

$$d(m_1, m_2) = 1 - \sum_{i=1}^{4} w_i * \frac{\mathbf{f}_i(t_1) \cdot \mathbf{f}_i(t_2)}{\|\mathbf{f}_i(t_1)\| \, \|\mathbf{f}_i(t_2)\|}$$

with w_i being a weight assigned to each feature vector according to its power to indicate the subject matter. Experiments with Twitter data have shown that $w_1 = 0.3$, $w_2 = 0.3$, $w_3 = 0.2$, and $w_4 = 0.2$ are suitable values.

3.4.3 Visual Tree Representation

Despite its hierarchical nature, the output of agglomerative clustering is often used to produce a partitioning of the element set into disjoint groups as well. To this end, a fixed distance threshold is chosen, and, starting from the root, the tree is traversed depth-first. Every time a node with siblings of smaller distance than the threshold is encountered, all elements of the subtree are collected into one of the final clusters. However, this procedure can lead to clusters that heavily vary in size and numbers and sometimes poorly reflect existing similarity groups. By giving users interactive means to explore the resulting hierarchy, starting from clusters of high diversity and moving into clusters of high similarity, that drawback can be turned into a helpful tool of identifying actual topics within the structure.

In *TreeQueST* the cluster tree is thus visualized based on a binary space partitioning. Starting with the root node, the visual space is recursively shared between the siblings by splitting it alternating in a vertical and horizontal fashion. A recursion path is terminated when it either reaches a leaf-node, or when the similarity distance of a node's siblings is below an interactively adjustable threshold. We then call such nodes *display nodes*. All data below these nodes is aggregated to resemble a topic in the visual overview.

At the beginning and after each zoom-in, an initial distance threshold can be determined automatically by finding an appropriate cutoff point in the hierarchical cluster structure. To this end, the L-Method as proposed by Salvador and Chan [2004] is used in order to find a trade-off between size and homogeneity of clusters. Usually, the merge distances in the dendrogram are monotonically decreasing when moving from the root node down to the leaf nodes. However, for any given subtree of the hierarchy, there is usually a transition level where the distances start to become significantly smaller than before this point.

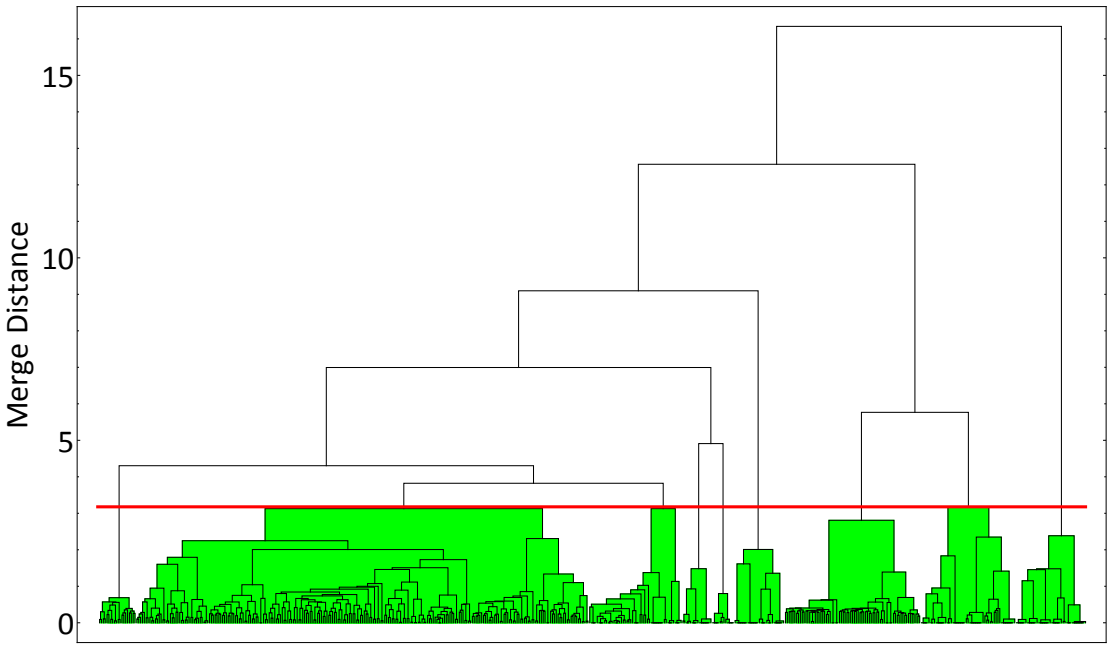
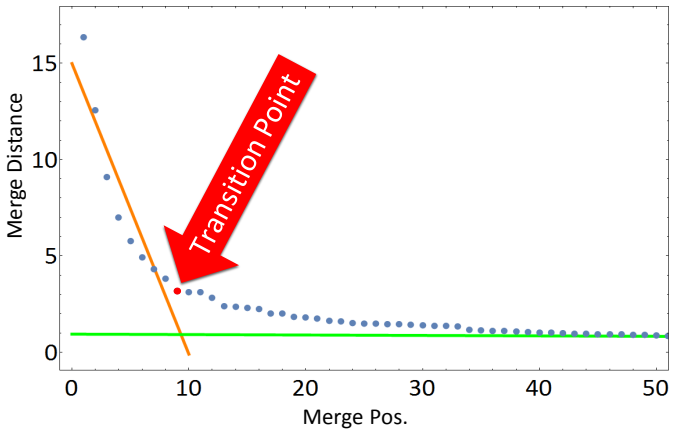


Figure 3.4 — An example dendrogram based on agglomerative clustering of 530 Twitter messages. An automatically computed distance transition point at the 9th highest merge distance is highlighted as red line. At this display level, the nine green sub-clusters would have been aggregated and visualized as topics in the information space.

This transition level thus also indicates where sub-clusters start to become significantly more homogeneous.

To detect this transition level, one first sorts all observed merge distances in decreasing order, as can be seen in the evaluation graph (Figure 3.5). The transition point can be found in the area with maximum curvature, which is also called the "knee" of the curve. As the regions to the left and right of the knee are often approximately linear, the L-Method tries to find an optimal fit for two lines that match these regions, e.g., based on least squares regression. These lines would then intersect in the transition area and thus provide an estimate of the actual knee point. To find the optimal lines, the algorithm incrementally selects a pivot point that partitions the plot points into a left and a right subset. For each partitioning, the sum of the mean squared errors of both approximations is calculated, and the pivot point with the minimum resulting errors is chosen as transition point. Figure 3.5 illustrates how this method has been performed for the clustering shown in Figure 3.4. The calculated transition point has been highlighted as red line in the dendrogram. If we visualize the



► **Figure 3.5 —** Evaluation graph of the clustering presented in Figure 3.4. Based on the fitted lines (orange and green) one can determine a distance transition point at the "knee" of the curve. In this case, the pivot point with best fit corresponds to the 9th highest merge distance.

tree hierarchy at this display level, *TreeQueST* would aggregate all messages contained in the subtrees (highlighted in green) into visual topics.

Because the tree is usually re-rendered in a few milliseconds, fluid interactions can be achieved, allowing users to intuitively change this initially determined granularity of the represented information space. Furthermore, they can select represented nodes with the mouse and zoom into that area. In this case, the selected node is used as the new root node for the visualization, and a new initial display distance for the subtree is calculated.

3.4.4 Tag Clouds, Spatial Layout and Exploration Lens

Once the above described splitting has terminated at some leaf or display node, the messages in its subtree are visualized in two layers within the space that has been assigned. For the top layer, a weighted tag cloud is generated based on either the terms, hashtags, or usermentions that are most prominently used within the messages of the subtree. By this means, users can get a quick indication of potential subtopics represented in the display node. The tags are weighted by a global *idf*-dictionary together with an ad-hoc *idf*-dictionary that is specifically generated for the cluster hierarchy. The tags which are most specifically relevant to the display node thus receive the highest weight. The tag clouds are rendered based on a Wordle-inspired [Steele and Iliinsky, 2010, Chapter 3] implementation. A global scale to determine the size of a tag based on the square root of its weight is used, as the tag relevance is then comparable between topics. In addition, the occurrence of larger tags gives users an indication whether a display node already constitutes a coherent topic, or whether they should further explore its siblings.

In the second layer, a spatialization of the messages is generated that groups more similar elements within each others neighborhood. This allows users to

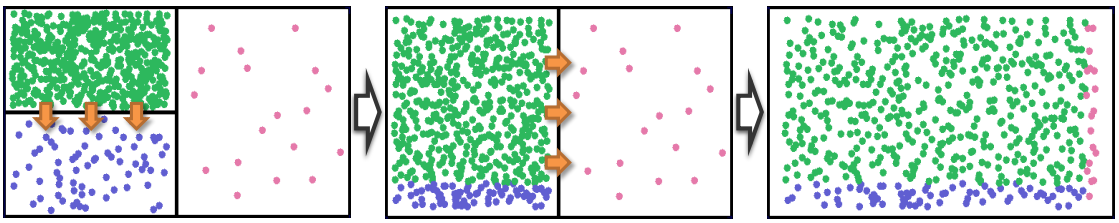


Figure 3.6 — Spatialization of messages based on the tree similarity structure. The spatial layout for any given display node is generated by subsequently merging the layouts of its siblings with an aspect ratio based on their message density. © 2014 IEEE

assess the portion and density of messages within a display node, and to find coherent similarity structures in deeper levels of the tree. The similarity-based cluster hierarchy already provides a simple means to spatialize the messages of a subtree according to their closeness. Starting from the tree's leaf nodes, the spatial layout is generated by placing the leaf's single message randomly within a $[0,1] \times [0,1]$ coordinate space. Moving upwards in the tree, the layouts of siblings of a node are then merged based on the ratio of the number of messages they contain. It can be determined based on a node's tree depth whether the merge has to be done in a horizontal or vertical fashion. The process is illustrated in Figure 3.6. By this means, the similarity information is preserved on any display level, and closely related messages will also be spatially next to each other.

The layout furthermore serves as a basis to provide an additional content exploration tool. An interactive exploration lens (see Section 2.1.3) can be moved over the messages in the second layer using the mouse cursor. The most prominent words and/or hashtags within the lens are then displayed as a focused tag cloud around it. Users can change the size of the lens and choose whether the term relevance should be ranked based on *idf*-weighted or absolute frequency. By applying this lens, users can quickly assess the range of contents within a node before they decide to drill down on the topic.

Polarity exploration is supported by an optional sentiment highlighting based on the SentiStrength Thelwall et al. [2010] library, which can be activated on demand for the spatialized messages. Users can thus find and explore areas of positive or negative affection towards or related to a topic to inform their reasoning. The individual messages are colored according to the respective sentiment of the message. Compared to color coding the complete cell, outliers are thus more visibly highlighted from the dominant sentiment in a cell. Because of

the similarity spatialization, users can also detect coherent sentiment structures in deeper levels of the hierarchy while they still navigate at upper levels.

3.5 Automated Query Construction

As part of the extended *TreeQueST* exploration process, the system generates queries that try to cover large numbers of messages contained in the gathered topics and as few messages as possible from all non-gathered topics. This section describes how this query is algorithmically created. The query is then inserted into the search bar to be executed through the service API and allow users to retrieve more messages for the topics they consider interesting. To evaluate whether the query would achieve the desired results, users are furthermore provided with means to manually manipulate it and immediately explore the effects on the current sample set.

3.5.1 Algorithm

Suppose we have a set of messages T_g, called the *wanted results*, which in our case is the messages from the user-gathered topics, and another set T_b, called the *unwanted results*. Furthermore, we have a set of all regular words, i.e., terms and hashtags, that are contained in messages of the union $T_g \cup T_b$. In a naive approach we could just try to build a Boolean query, e.g., as a disjunction of conjunctive clauses of negated and non-negated words, that covers exactly all messages from T_g and none of the messages from T_b. We could then use a method for Boolean function minimization, such as the method of Quine [1952] and McCluskey [1956], in order to find a shortest query that imposes a less restrictive behavior when applied to a larger corpus, and that does also not exceed limits for request length of the service APIs.

This solution, however, suffers from a range of problems. First, computing a minimized form of a Boolean query is considered an intractable problem. Since we have a very large set of variables, even fast algorithms like the *Espresso Logic Minimizer* [McGeer et al., 1993] would hinder fluent interactions in the workflow. Second, depending on the messages, the optimal solution would often still be quite large, as there might be outliers that have to be covered with indispensable query clauses. Third, and most importantly, in some cases there might be no correct and complete solution at all, e.g., if $T_g \cap T_b \neq \{\}$.

TreeQueST therefore uses a heuristic solution, which might accept some unwanted results and also miss some wanted results, in order to allow fast computation and powerful queries. Furthermore, since the set of wanted results

T_g is partitioned into topics $L_1, \ldots, L_m$, one should adhere to the additional constraint that the query has to cover as many of these topics as possible.

Assume we have a large *idf*-dictionary IDF_{global}, as it has been discussed in Section 3.4.2. Let $W(T)$ be the set of all terms and hashtags that are used in a given message set T, and let $\|T\|_w$ be the number of messages in T that contain term or hashtag w. Furthermore, let q be our initially empty query, and let q_0 be the seed query, i.e., the one that was used to initially load the sample dataset. The query creation algorithm then works as follows:

1. Find the word $w \in W(\bigcup_{i=1}^{m} L_i)$ that maximizes the weight

$$priority(w) = IDF_{global}(w) * \frac{\left\|T_g\right\|_w}{1 + \|T_b\|_w} * \sum_{i=1}^{m} \frac{\|L_i\|_w}{|L_i|}$$

2. **If** $IDF_{global}(w) \geq IDF_{global}(q_0)$,
   - set $q := q\ OR\ w$,
   - **else**, set $q := q\ OR\ (q_0\ AND\ w)$.
3. Remove all messages from $L_1, \ldots, L_m$ that contain w.
4. **If** $\left|\bigcup_{i=1}^{m} L_i\right| > 0$,
   - continue at 1.,
   - **else**, terminate.

The final result will be a query that reasonably generalizes to unknown messages, prefers relevant topic messages over non-relevant ones, and usually tends to reduce the set of messages that are considered within the hidden corpus. The latter condition mimics the general Scatter/Gather behavior of rather reducing the document set in every iteration. Furthermore, due to step 2, the algorithm tries to add topic-relevant query words to cover messages missing the initial seed query. If the query is too long for the service API, it is cut between two conjunctive clauses right below the maximum allowed length.

3.5.2 Evaluation and Manipulation

Sometimes the result of automated query generation still requires refinement, as the algorithm chose keywords that users consider misleading, or the coverage of topics does not reflect their preferences. They can thus manually manipulate the query in the search bar, and the corresponding results in the local sample

are immediately highlighted, as described before. They can then further explore the topic hierarchy and try to understand the coverage. The lens, introduced in Section 3.4.4, can be a valuable tool in this process. Sometimes only certain areas of a display node are covered by the query, and the lens can be used to investigate their contents before zooming into the area and leaving the current context.

3.6 Case Study

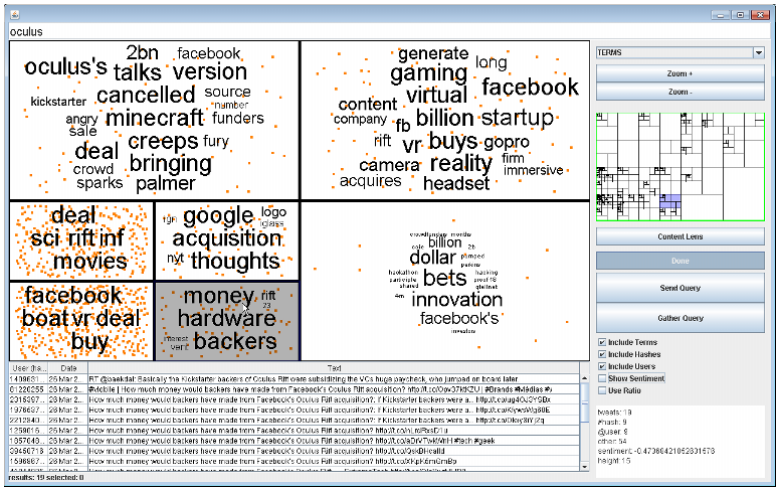
To demonstrate the applicability and workflow of the *TreeQueST* approach, a case study in media awareness has been conducted. An analyst assumed the role of a journalist tasked with assessing the impact of an ongoing event. He had to find related information, investigate the communities reactions and opinions, and identify news content that could dominate further media reaction. In order to evaluate the real-time capabilities of the approach, this case study was actually performed as the event unfolded. The following is therefore the true account of an exploration session with *TreeQueST*.

In addition to this case study, an experiment with multiple users and a similar topic area was also carried out. While the following case study is supposed to illustrate the functionality of the approach, the focus of this study was on statistical results about the usability and practical usefulness of the concept. The results of this study will be reported in the evaluation part of this thesis (Chapter 7).

Session Report

Oculus VR is a California-based start-up that specializes in virtual reality devices. Using novel hardware components for fast head-tracking and large field of view, their intent is to revolutionize the gaming market with new head-mounted-displays (HMD). The company's founding through Kickstarter in 2012 received a lot of media attention, as the campaign raised almost 2.5 million US Dollars, making it one of the top 5 Kickstarter funded projects by that time. On March 25, 2014 (9.30 pm UTC) Mark Zuckerberg announced that the company had been acquired by Facebook Inc. for US$2 billion in cash and shares. Within minutes, this announcement had tremendous impact in news media, the web, and almost all social media platforms.

The analysis of this business event was started on March 26, about ten hours after the announcement, by entering the seed query `Oculus` into the search bar and hitting the *Send Query*-Button. In the default configuration, the system then collects about 1000 tweets from the Search API using the MIXED-parameter,



◀ **Figure 3.7** — Initial view in the case study after executing the seed query `oculus`. Various hot topics are already recognizable. The analyst can use this overview as starting point for further interactions.

which means that the results will consist of a mix of most recent and most popular tweets for the query. Next, the system computes the matrix of pairwise distances for the tweets in order to perform the clustering. To support a fast computation, parallel processing is used in the implementation, and *TreeQueST* is executed on a compute server with 40 Intel Xeon E7 Cores at 2.13 GHz clock speed. In this study, the distance matrix was computed in 2.1 seconds and the subsequent clustering was performed in 3.6 seconds.

The results are immediately visualized, and one could already recognize certain hot topics at the initial display distance (Figure 3.7). Since the analyst was interested in peoples reactions, he first applied sentiment analysis, which is activated by a check-box. The tweets are visualized in green, if the system recognized positive sentiments, red in case of negative sentiments, and neutral tweets are shown in gray. Based on this highlighting, the analyst observed certain areas that were largely dominated by negative sentiments. The mouse wheel was used to change the initial splitting granularity, and, at a deeper level, a topic area that was completely dominated by negative tweets was selected. The analyst zoomed-in on the area in order to investigate its contents. The large tags `kickstarter`, `backers`, `demanding` and `refund` in the tag cloud now gave an indication what the topic might be about, and reading one or two of the tweets confirmed that there was a certain outrage amongst Kickstarter backers that felt betrayed by the deal. The analyst then right-clicked on the topic to gather it as relevant for the inquiry. The node area is subsequently highlighted in green in the information space as well as in the minimap, which always shows all gathered, selected, and currently zoomed-in nodes of the global hierarchy. The analyst continued to explore the tree in order to find more topics that could further inform his investigation. After a short search, he discovered that the community was somewhat concerned about a possible

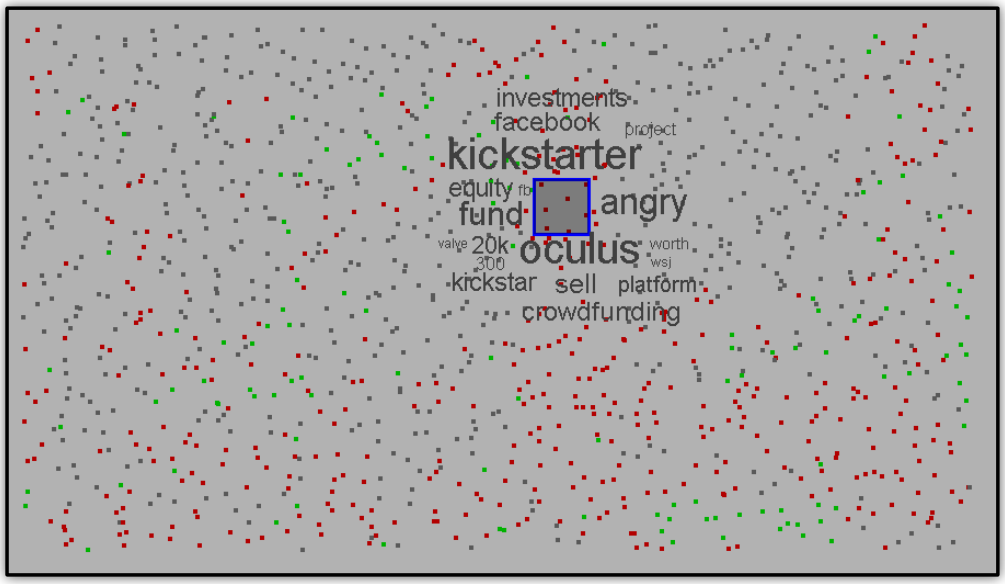


Figure 3.8 — Investigating sentiments using the exploration lens. The tags around the lens indicate that the negative tweets in that area are about Kickstarter backers feeling betrayed by Oculus.

redesign of Oculus products under Facebook's influence, and that the developer of the popular game Minecraft announced to cancel his existing cooperation with Oculus. This again lead to a lot of reactions. After the analyst had gathered some of the closely related topics, which seemed to have an impact on opinion building, he clicked on the *Gather Query*-Button to generate a new query more specific to the collected topics. The result combined the seed query `Oculus` with emotional tags like `anger` and `outraged` as well as prominent content tags from the gathered topics like `redesign` and `rebrand`. After applying the generated query, the results showed a large amount of negative sentiments, i.e., large portions of the information space were covered by red points. Starting at the root of the tree, the analyst therefore decided to investigate the content of these sentiments using the lens, as it can be seen in Figure 3.8. The negative tweets were again mostly about Kickstarter backers and reactions to the decision of the Minecraft developer not to cooperate anymore. However, there were further tags indicating that the gaming community was also quite outraged about the deal.

By further investigating the hierarchy of the new sample, the analyst found several new topics such as `creator, talks, reddit` and `plans, redesign, reddit` (cf. Figure 3.9). Exploring these topics revealed that Palmer Luckey,

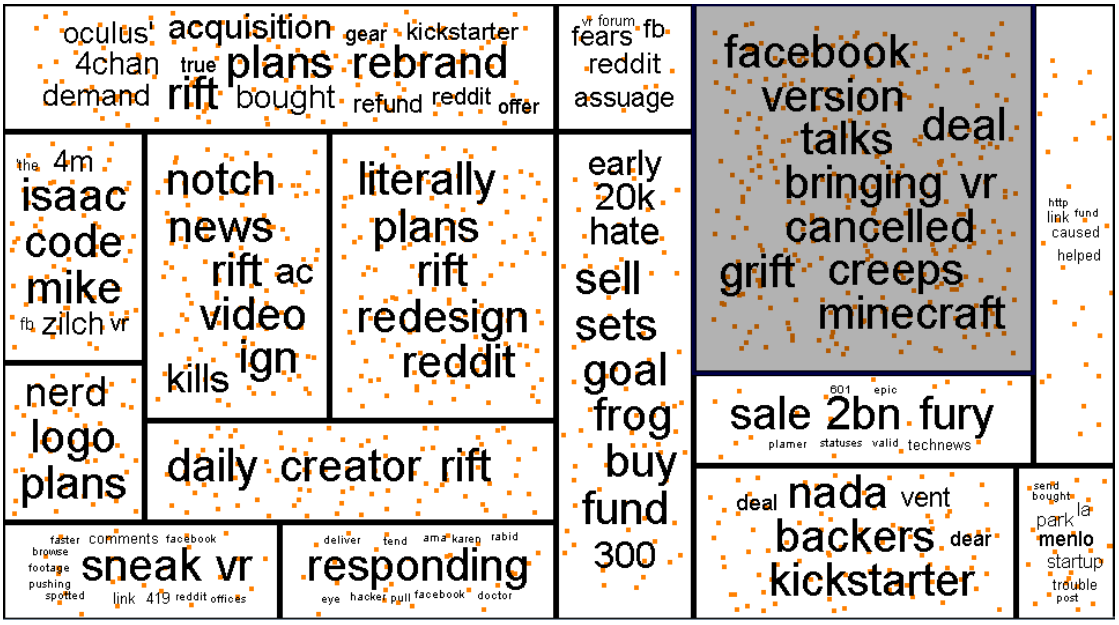


Figure 3.9 — The image shows part of a snapshot taken during the scenario study that investigated the Oculus VR acquisition. Several areas show easily recognizable topics, such as the Kickstarter backers being frustrated about the deal, that people are concerned about a possible rebrand or redesign, and that the creator responds to the community in Reddit.

the founder and mastermind behind Oculus VR, was reacting to accusations from the community in Reddit. By browsing through the hierarchy, the analyst quickly found a link to that Reddit entry, in which Luckey responded that people will not need a Facebook account to use Oculus Hardware, and that development will remain open. The analyst then gathered topics related to that new information and again generated a new query. By investigating the topic sentiments in this sample, the analyst was able to confirm that the message was positively received by some audiences, but the overall tone was negative towards it and dominated by disbelief. As the further development of the discussion was important to him, the analyst decided to use the last generated query as primary request for continued monitoring.

Discussion

Summarizing the observations of the case study, the *TreeQueST* Explorer allowed the analyst to assess the ongoing situation and find key information entities right as they appeared in the network. He was furthermore supported in understanding how different topics relate to each other and could greatly

benefit from using sentiment analysis and the spatialization of tweets to direct his investigations.

However, there also remained the concern that relevant information could have been missed, as it might not have been covered by the seed query. The subsequent iterations might have even directed the analyst away from it. Checking the news media on the following days did not support that concern, but it remains an existing danger to the solution. The analyst should thus sometimes move back to the beginning and try a completely new seed query to minimize that risk.

CHAPTER 4

Visual Event Discovery

Geo-referenced social media messages, such as tweets written with GPS-enabled mobile phones, have an outstanding value in disaster management and related situation awareness domains. This has three reasons: First, geolocations indicate whether users might have actually participated in or observed an ongoing situation. If they are actual eyewitnesses, their reports are much more important than rumors coming from remote locations. Secondly, available geo-data provides a simple means to filter the data to relevant geographical areas, and thus to cope with data volumes. Thirdly, and most importantly, if various users post about the same topic during the same time and at the same location, their posts are more likely related to an ongoing *event*, which is a central entity in situation awareness [cf. Matheus et al., 2003].

In various SA applications, the potentially relevant events are often reflected in a specific *spatiotemporal shape* in the geolocated data. Or it is at least possible to define a spatiotemporal hull, in which all possibly relevant events in a given situation can be fit. This observation provides a quite simple means of generating an overview and anomaly indication display - at least, if we can assume that this kind of information matters most. The following chapter presents methods that utilize the spatiotemporal shapes and patterns of data to enable the event-centered discovery element of the overarching analytics model.

In contrast to the query-based pre-filtering that was described in the previous chapter and the supervised classifiers that will be discussed in the next chapter,

these methods are not driven by relevance-presumptions of the analyst. Instead, unsupervised, data-driven algorithms are used to separate coherent patterns and anomalies from signal noise based on similarities in space, time, and content. By this means, the techniques not just help analysts to find and investigate known events, but also to highlight events they have not even been aware of. However, as the algorithms cannot understand semantics, analysts have to be provided with powerful visualizations that allow them to investigate larger numbers of automatically extracted findings using aggregation and relevance metrics.

The chapter is structured as follows: Section 4.1 will motivate and introduce the presented methodology based on the 2011 VAST challenge [Scholtz et al., 2012]. This scientific contest helped to identify some of the true problems of scalable social media analysis. At the same time, it uncovered how the characteristics of spatiotemporal data can provide a valuable basis for customized identification schemes. After a brief overview of related background in Section 4.2, a new algorithm and an interactive visualization scheme will be introduced in Section 4.3. Together they identify the anomalous patterns in the data based on stream-enabled cluster analysis and present them to the analyst with adaptively summarizing overviews. The outcome of this approach is a visual interface that shows large volumes of possibly relevant events as textual indicators on an interactive map.

However, several of these automated findings will often not be of relevance in situation monitoring, such as seasonal patterns, public gatherings, or general topics that suddenly become more popular in certain areas. Section 4.4 will therefore investigate additional relevance metrics that utilize geographic language models to decide which content is unusual or unexpected for a given region.

Parts and ideas of this chapter have been previously published in:

- H. Bosch, D. Thom, M. Wörner, S. Koch, E. Puttmann, D. Jackle, and T. Ertl. ScatterBlogs: Geo-spatial document analysis. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 309–310. IEEE Computer Society, 2011
- D. Thom, H. Bosch, S. Koch, M. Wörner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 41–48. IEEE Computer Society, 2012

- D. Thom, H. Bosch, and T. Ertl. Inverse document density: A smooth measure for location-dependent term irregularities. In *International Conference on Computational Linguistics COLING*, pages 2603–2618. Indian Institute of Technology Bombay, 2012

4.1 Spatiotemporal Anomalies

The VAST Challenge is a scientific contest held in conjunction with the annual IEEE VIS conference. Usually, participants are provided with an artificially generated dataset together with a list of tasks that have to be solved using visual analytics means. In 2011, one part of the challenge was titled "Characterization of an Epidemic Spread". It featured a synthetic dataset that combined real and artificially generated microblog messages, which were placed in the fictitious city of "Vastopolis". Based on this data, the challenge was to investigate an epidemic virus outspread that happened in the city, find out where it began and how it developed, and determine possible means of transmission [Grinstein et al., 2011].

As an entry point for the analysis, the participants were provided with a list of characteristic symptoms that had been reported by affected patients. The list included gastrointestinal symptoms, like nausea, vomiting, and diarrhea, as well as flu-like symptoms, such as fever, chills, sweats, fatigue, and coughing. A quite straightforward means to proceed from these initial hints was to use textual search engines to discover and investigate messages that mention these symptom keywords. Additionally, as all messages were provided with geolocation, one could highlight geospatial or even spatiotemporal patterns by plotting the respective messages in a 2D scatterplot or a 3D space-time cube. This approach has been taken by several participating teams at least as one part of their solution. An example result generated with an early *ScatterBlogs*-version [Bosch et al., 2011b] can be seen in Figure 4.1. The snapshot shows the 2D and spatiotemporal 3D view of the flu-related messages in green and the gastrointestinal-related messages in purple. One can easily recognize that the keyword searches resulted in two large spatiotemporal clusters of messages in the city center and alongside a river that moves through the city. In addition, there are several smaller clusters of flu-related messages distributed all over the city. With a closer look at the map labels, users were able to identify these locations as the different hospitals of Vastopolis.

The standard solution, which was published after the contest, revealed that the outspread began somewhere near the city center. Moreover, the gastrointestinal form of the disease was transmitted via water in the river, and the flu-like pathogens were blown over the downtown area by the wind. This corresponds

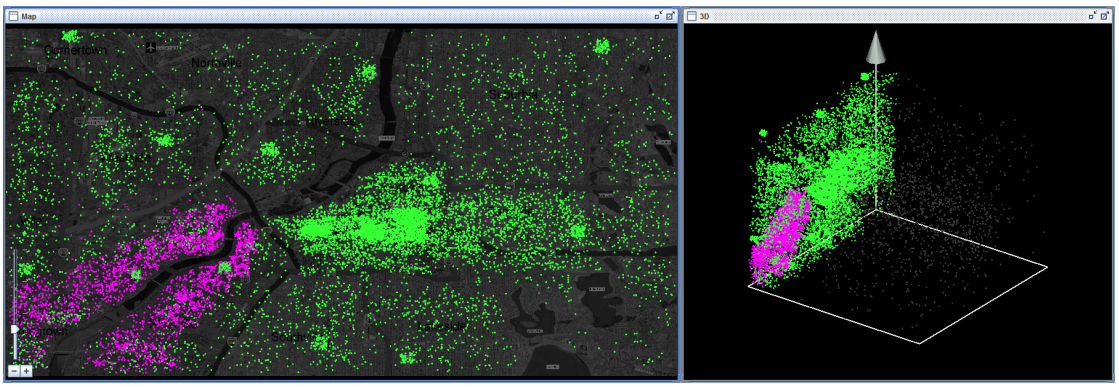
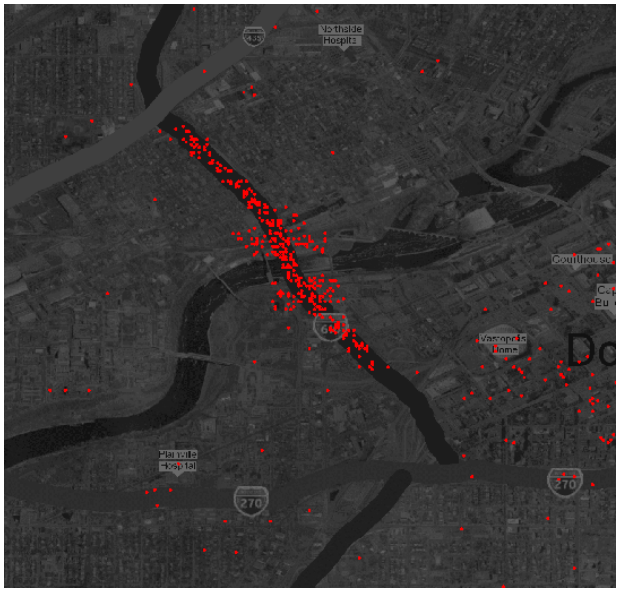


Figure 4.1 — Keyword-based visualization of messages during the VAST 2011 challenge. The data is filtered to show the last three days in the dataset. Green messages mention flu-like symptoms. Purple messages mention gastrointestinal symptoms.

to the two large clusters that seem to intersect at the location of origin. The solution furthermore explained that the people affected by the gastrointestinal disease recovered quickly, while the flu-related illness was more severe, and the affected people ended up in the hospitals during the last days of the scenario's timeframe. Here they wrote more of the messages mentioning their symptoms, which results in the smaller green clusters.

However, in addition to the symptom outspreads, the dataset contained further relevant events that were not so easily discovered. Here the challenge description gave no hint what to search for. The most important event was the actual cause for the virus outspread, which the solution revealed as a severe traffic accident that happened on a bridge near the city center. In this accident, a truck carrying hazardous biological agents was severely damaged. The agents were subsequently spilled into the river and carried along by the wind, which was blowing from West to East at that time. This event too was observed by the Vastopolis citizens, and the reports about it resulted in a distinct cluster as well. This cluster, however, was initially concealed in the data, as there were no keywords provided that would lead to its discovery (Figure 4.2).

From the symptom outspread events, one could learn that the right keywords led to the discovery of patterns concealed between the "noise" of the 1 million randomly distributed messages in space and time. As the distinct spatiotemporal shape and density of these message clusters indicated situation-related events, they can also be considered *spatiotemporal anomalies* of the data. However, the most important anomaly, the one that would explain everything, could



▶ **Figure 4.2 —** The traffic accident happened on a bridge near the Vastopolis downtown area. In this accident, a truck was damaged and spilled biological agents into the river beneath the bridge and into the air. These agents were the actual cause for the later epidemic in the city. The image shows the corresponding cluster of messages mentioning keywords like `truck`, `spill`, or `accident`. Without having the right keywords, the cluster is not easily discovered.

not be easily discovered, since analysts had initially no reason to enter keywords like `truck`, `spill`, or `bridge`.

The core idea of this chapter explains how the insight discovery process can essentially be *inverted* in order to tackle these unexpected anomalies. In the straightforward process, an initial hypothesis about the situation leads to keywords; keywords lead to the revelation of spatiotemporal anomalies; anomalies can turn out to be events; and investigating messages from the events can confirm the hypothesis. Accordingly, the inverted approach builds on the assumption that peoples' reports about incidents often generate spatiotemporal clusters of similar content. Using cluster analysis means, one can try to identify these patterns between noise and highlight them to the analyst. However, not only relevant reports may create such clusters, but also messages about all kinds of local events, such as concerts, street protests, or sports venues. The cluster analysis should therefore be complemented by an indication of the keywords that would reveal the anomalies in the straightforward process. From these keywords, analysts can then immediately infer what the anomaly might be about and form hypothesis how they might be related to the situation.

A rudimentary implementation of this method was already applied to the challenge dataset, as can be seen in Figure 4.3. The visualization not only highlights the truck accident in the middle of the city but also the clusters corresponding to the symptom keywords. Moreover, other events that were hidden in the data are now shown, such as a plane crash that happened at the

Figure 4.3 — Visual cluster discovery based on the inverted event discovery scheme. The tags on the map clearly indicate the truck accident, the symptom outspreads, and other events that were additionally hidden in the data.

airport in the north-western part and a bomb threat that was reported in the downtown area.

The following sections will explore how this basic visual event discovery scheme can be implemented for real-world streaming data and discuss possible extensions. Based on case studies with actual Twitter data, it will be demonstrated that the spatiotemporal anomalies are not just an artifact of the generated challenge data but an ubiquitous phenomenon in actual social media. They thus provide a pivotal element in tackling relevance indication.

4.2 Background

There have been numerous related works that also address the general topic of social media event detection. Existing approaches can broadly be categorized into plain data mining methods, which sometimes exhibit static result visualizations, as well as interactive analysis systems, which often enhance the detection with content exploration. This section provides a brief overview of

most important approaches and also highlights the background of previous research on spatiotemporal visual analytics, tag clouds, and geospatial term normalization.

4.2.1 Visual Analytics of Spatiotemporal Data

The general topic of spatial and temporal data analysis has since played an important role in information visualization. The relevance of addressing this domain with visual analytics methodology was highlighted by the VA research agendas [Thomas and Cook, 2005; Keim et al., 2010, chapter 3/chapter 5]. Keim et al. argue that space and time require special treatment and should not be considered as regular data dimensions. They highlight specific characteristics that render plain data visualization a challenging endeavor: Because of the dependency of the dimensions, the use of certain statistics techniques is restricted, but, at the same time, it allows spatiotemporal inference and interpolation. Moreover, inherent uncertainties of the data stem from problems with measurement errors or imprecision (e.g. GPS restrictions), data processing artifacts, missing separation between objects, and signal noise. Finally, the different scales of the data are constituted by the often fluctuating extent and granularity of measurements, which can comprise decades of observations on a global level or just few milliseconds in a Petri dish. Sometimes, these different scales can be observed in a single dataset.

However, despite this challenging nature, spatial and temporal coordinate systems also provide an easy way to facilitate exploration, filters, and selection. In the *TreeQueST* approach (Chapter 3), messages had to be artificially arranged in a spatial pane to allow exploratory interactions. However, if entries are already aligned according to spatial and temporal closeness, the visualization can employ a natural means of showing the data to the user. Numerous approaches have thus applied visual analytics techniques in this domain. For example, Andrienko and Andrienko [2011] employ cluster analysis to aggregate movement data and present overview visualizations. They employ a custom designed density-based algorithm, where parameters defining the neighborhood have to be configured, and show that it outperforms a simple K-Means scheme. Although the produced visualizations are very fast and allow spatial exploration, highly interactive analytics were not in the scope of this work. However, in a closely related work, they put more focus on the temporal aspects of the data and present a four step procedure of extracting events in movements, finding significant places, aggregating similar movements, and analyzing the data with visualizations, such as a 3-dimensional space-time cube [Andrienko et al., 2011]. Further important works in this domain were provided by Eccles

et al. [2008], who enable spatiotemporal narratives in a unified perspective, and by Aigner et al. [2008], who investigated specifics of time-oriented data. Based on their study, Aigner et al. identify three major requirements for event-based visualizations: communicate interesting findings, emphasize them compared to other data, and provide clues why the findings stand out. The methods presented in this chapter closely follow this recommendation by semantic indication, visual highlights, and spatiotemporal contextualization.

The more specific technique of using tag clouds on interactive maps was first presented by Jaffe et al. [2006], and it was later also employed by Slingsby et al. [2007] and Wood et al. [2007]. The latter work also presented the idea of applying spatial clustering to collect data into tags. However, none of these approaches is devised to scale with real-time streaming data or to address event discovery. Furthermore, they do also not exploit the temporal dimension for clustering, which leaves it to the user to explore tag frequency by interactive specification of time intervals. Wood et al. use a straightforward `binning` approach, which allows for hierarchical clustering but comes at the cost of having a fixed, potentially suboptimal grid structure that might represent spatial clusters inadequately at certain levels of detail. A Twitter-based tag map display was presented by the commercial system *Trendsmap*.[1] In this instance of the technique, all Twitter messages from predefined geographic regions, e.g., cities, are collected and aggregated as tag clouds. They employ no means for automated event detection, interactive examination, or semantically adaptive zooming.

4.2.2 Social Media Event Discovery

One of the most early experiments on social media event discovery was conducted by Sakaki et al. [2010]. They exploit patterns in Twitter to detect earthquakes and to track the trajectories of typhoons. Based on support vector classification, they first identify tweets related to critical events and subsequently employ Kalman and particle filters to predict their location and movement. This is also the first work that demonstrated the high probability that critical events can indeed be located in Twitter. Other notable approaches without visualization aspect include the works of Schulz et al. [2013], in which a highly sophisticated feature extraction pipeline was used to detect and rank incident-related tweets with supervised classification, and the works of Weng and Lee [2011], where Wavelet-based signal analysis is employed to filter away trivial words. In a subsequent step, Weng et al. also employ cluster analysis to group possibly event-related tweets. However, none of these fully automated meth-

[1] `http://trendsmap.com/`

ods provides perfect precision or recall, and analysts thus have to cope with the chance of missing relevant data objects or they have to manually inspect tremendous volumes of irrelevant data. Moreover, in these approaches, analysts have no elaborate means to further examine sub-events, to get an overview of the overall situation, or to better understand what they might be missing.

Visualization researchers have therefore integrated automated mechanisms with interactive representations of discovered anomalies. *Twitinfo* from Marcus et al. [2011] applies a custom peak detection scheme to identify and visually label unusual developments in Twitter debates. The Twitter stream is presented in a temporal overview, and the labels are placed as annotations over the data. The user can select the peaks, which will highlight the contents in a table and show geolocations on a map. However, the system only works with streams that are already pre-filtered to some topic, such as an ongoing football game. The detection mechanism only considers the temporal dimension of the data, and if larger or global data streams have to be analyzed, peaks of smaller events can easily be obscured by the unrelated message volumes. Dou et al. [2012] enhance the approach by first automatically dividing a larger message stream into topic-specific substreams based on LDA topic clustering (see Section 6.3.1). Subsequently, they identify bursty behavior in the topic streams to trigger an event alarm. The approach furthermore enhances event examination by showing related locations and persons based on entity resolution. However, in LDA, the number of possible topics has to be predefined by the user and the approach ignores information that can be drawn from spatial patterns of the messages. The technique presented in this thesis addresses smaller events through geographic and content-based separation. As the temporal and geographical closeness of related messages boosts their relevance, smaller sub-events have a higher chance to stand out between larger events.

A comprehensive overview of visual analytics approaches for event detection has recently been provided by Wanner et al. [2014]. Their statistical overview shows that no other social media VA system employs unsupervised partitional clustering to detect events.

4.2.3 Term Relevance Metrics

Work related to term relevance normalization, as it will be featured in Subsection 4.4 of this chapter, can be found in two areas. First, known geolocated resources have been used to establish meta-documents in order to assign non-geolocated resources according to their similarity to these meta-documents. This is basically the inverse problem to the one addressed here. Secondly, the geographic information of resources has been exploited to establish a *geo-*

ranking of search result lists. This kind of research was particularly prominent in the information retrieval domain.

An example of the first type was presented by Wing and Baldridge [2011], who cover the world with a geodesic grid and calculate the probability of term occurrences per grid cell using various geolocated document collections, including Wikipedia articles and Twitter messages. Given a new document, they calculate the similarity of term-distributions between the document and all cells to find the closest match. That basic approach was later enhanced by Roller et al. [2012], who use an adaptive space partitioning grid based on *k*-d-trees instead of a regular one. Although the coverage of distinct locations in smaller regions, such as cities, is increased, this discrete method of assigning one cell per document also leads to decreasing quality with increasing grid resolution. If cells become smaller, less documents are assigned per cell, leading to an overfitting of data when the pseudo-documents become too small and specific. Similar to Roller et al., the method presented in this chapter also employs an adaptive grid based on space partitioning to facilitate scalable processing and storage without loosing generality. However, in contrast to their approach, a smoothing kernel is used, such that the measure's quality is always increasing with higher resolutions.

In the information retrieval domain, geospatial variations of *tf-idf* were presented by Zhang et al. [2010] and Ahern et al. [2007]. Zhang et al. define their measure in the context of tag-related query processing for geolocated Web 2.0 resources. The approach tries to find regions where each of the query elements is covered by at least one nearby Web 2.0 document. Ultimately, the goal is to rank regions within the result set according to how characteristic the query is for each region. In these use cases, a term is characteristic for a region when it is *frequently* observed in the region but *infrequently* observed globally. The work presented in this chapter, however, contrasts term densities with long-term historic document densities at specific spots. A term can thus still achieve a high score for a specific region. Even if it is a globally common word.

4.3 *TagMap* Event Discovery

While the VAST contest dataset was a fixed corpus of messages limited to the boundaries of a single city, the situation is quite different if we cope with real-time data on a global scale. This section describes a visual analytics technique, called *TagMap*, that adapts the basic idea to real-world social media streams. It specifically addresses two challenges: that traditional cluster analysis algorithms were not devised for large and/or real-time streaming data, and that the visual interface should allow fluent level- and area-transitions on a global scale. To

tackle these challenges, we can build on two specifics of social media data steams, namely that the textual content is usually limited in length, and that new data will be incrementally inserted along the temporal dimension.

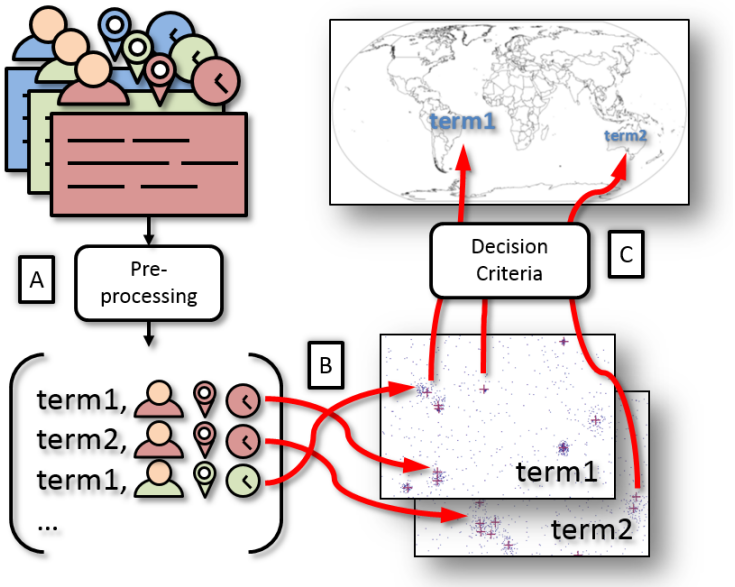
Compared to the *TreeQueST* component, which only has to cope with limited samples pulled from the search APIs, the *TagMap* is supposed to be applied to either complete or pre-filtered streaming API data once the stream has been opened. Due to its high computational complexity, hierarchical cluster analysis is thus not an option in this part. Instead, the *TagMap* analytics scheme is based on an enhanced K-Means clustering algorithm (see Section 2.1.4), which was specifically adapted to streaming data, as well as a novel visualization scheme that facilitates the scalable representation of identified anomalies. Utilizing the time-indexed nature of the data, the algorithm is optimized for scalability by employing incremental adaption to incoming elements. At the same time, it summarizes aging data and eliminates it from the active computation in order to reduce visualized content to a manageable amount. Incoming data is constantly clustered on a per-term basis, and anomalies are stored together with their time and location as well as an initially estimated significance score. Since the clustering is designed as a best effort heuristic, an overfitting of data can occur. However, this apparent drawback is subsequently turned into a feature by the adaptive visualization, as it automatically aggregates related clusters into larger highlights in zoomed-out views.

4.3.1 Procedure Overview

In contrast to the challenge data, actual social media streams contain thousands of clusters with varying density. And they are not just resulting from events. They are also generated in densely populated regions where large spatiotemporal message clusters happen to appear during every daytime. What we are thus looking for, are not any spatiotemporal message clusters, but clusters of similar or even identical *term usage*, which will frequently result from people collectively observing and reporting an event.

The real-time anomaly indication scheme of the *TagMap* combines three activities, as illustrated in Figure 4.4. The first activity comprises various NLP preprocessing steps, in which tokenization and optionally lemmatization as well as stemming are applied. Also, all stopword terms of the message, such as `the`, `and`, or `she`, are removed.[2] The remaining terms are then collected and each one is wrapped into an individual data object together with the message's user ID, timestamp, and geolocation. As these data objects play a significant

[2] More details on the complete preprocessing pipeline can be found in the *ScatterBlogs*-description in Chapter 6.



◄ **Figure 4.4** — Three activities to generate an overview of anomalies: A) Message terms are extracted and transformed to term artifacts. B) Quantization of term artifacts generates spatiotemporal clusters. C) Clusters selected by the decision strategy are considered as anomalies and represented as term-map overlay for exploration. © 2012 IEEE

role in the algorithm, they are henceforth named *term artifacts*. Due to the short length of social media messages, e.g., 140 characters in case of Twitter, the first activity would usually generate 5 to 10 term artifacts per message.

Following preprocessing, the second activity is a continuous cluster analysis of the extracted term artifacts based on the enhanced K-Means clustering scheme, which will be described in more detail in the following subsection. This component aggregates aging data iteratively. It thereby restricts the actively processed data to manageable amounts, e.g., to fit into main memory. By this means, one can achieve scalability in terms of document indexing. At the same time, the system is able to provide highlights for a fast and interactive overview even of large spatiotemporal windows. The output of the cluster analysis is a set of term occurrence anomalies. They consist of the term itself, the mean spatiotemporal location of the cluster (centroid), and a temporal histogram of the cluster's elements.

Based on this output, the third activity generates a graphical representation of the anomalies by placing visual labels on the map. The underlying visualization method tackles the problem of showing large numbers of the found clusters in a way that the most important anomalies as well as the terms that are semantically best suited to represent the underlying data at a given zoom-level are displayed. This component will be discussed in Section 4.3.3. Analysts can explore the generated overlay to get an overview of spatiotemporal anomalies that occurred in interactively selected timeframes and geographical areas.

4.3.2 Stream-enabled Cluster Analysis

The stream-enabled clustering scheme is closely related to the popular X-Means method [Pelleg and Moore, 2000]. Just like this approach, it is also a derivative of K-Means that tries to automatically estimate the best number of clusters. However, X-Means and other conventional cluster analysis methods do not facilitate real-time data processing or online analysis while the algorithm is still active. Usually, they are neither fast enough to support interactivity, nor do they scale to datasets of arbitrary size. When processing streaming social media data, we can, in contrast to other applications, discard old clusters at some point in time and store the location of their representing centroids permanently. Exploiting this characteristic of the data, the enhanced algorithm provides scalability to continuous stream processing of almost arbitrary duration. To this end, it basically performs a single, global, and continuous relaxation step of the regular K-Means scheme. That is, the clusters are adapted as new messages arrive, but globally the relaxation is not done repeatedly until an equilibrium is reached. Instead of using a predefined fixed number of centroids, a splitting mechanism is employed to identify new emerging anomalies and to accommodate noise - i.e. term artifacts that do not belong to any actual anomaly. These noise clusters are later discarded after examining the final cluster properties. The algorithm furthermore processes the messages on a per-term-basis. That is, for each and every term encountered in the messages, a new clustering branch is initialized, which covers only term artifacts referring to this specific term.

The algorithm works as follows: It begins by creating an empty list of cluster branches. Once the first message arrives from preprocessing, a cluster branch for each contained term is generated, and the term artifacts are assigned to newly created clusters in the respective branches. The centroids of these clusters are initialized with the spatiotemporal location of the message. As further messages arrive, it is tested for each term whether a corresponding branch already exist, and if so, the term artifact is assigned to the closest cluster of the branch. The centroid of this cluster is then adjusted by calculating the new mean location of all covered term artifacts. Conversely, if a term of the message is not already covered by a cluster, a new branch of clusters is added to the list and initialized with a cluster containing only the corresponding term artifact. Finally, it is tested for all modified clusters whether the average squared distortion of their elements is below some predefined threshold k. If the cluster exceeds the threshold, two new clusters are created in the branch and its elements are distributed between them using the basic K-Means (in this case 2-Means) cluster algorithm.

To put this more formally, the pseudocode for the procedure $handleMsg(T_m)$ in Listing 4.1 illustrates the basic algorithm. It does not include the storing and noise elimination as well as some implementation specific details, which will both be addressed in Section 4.3.2. The program variables are defined as follows: T_m is a set of term artifacts that have been extracted from a particular message m. A term artifact t is represented as a record consisting of a term $t.term$, a user ID $t.user$, and a location in a unified spatiotemporal domain $t.loc \in \mathbb{R}^3$. Furthermore, $C(t)$ is a hash table that uses terms as keys and maps them to sets of clusters. Each cluster c in these sets is a record consisting of its centroid location $c.loc \in \mathbb{R}^3$ and a set of associated term artifacts $c.reg$, called the *cluster region*. The *handleMsg* procedure is performed for each new message m arriving from the input stream after the term artifacts have been extracted.

Listing 4.1 — Basic Algorithm

```
1  procedure handleMsg(T_m)
2    Cluster: c, c_1, c_2 ← new Cluster()
3  begin
4    for t ∈ T_m loop
5      if C(t.term) = ∅ then
6        c.reg ← {t}
7        c.loc ← t.loc
8        C(t.term) ← {c}
9      else
10       c ← arg min_{ċ∈C(t.term)}(‖ċ.loc − t.loc‖_2)
11       c.reg ← c.reg ∪ {t}
12       c.loc ← (1/|c.reg|) * Σ_{i̇∈c.reg} i̇.loc
13       if D(c) > k then
14         c_1, c_2 ← split(c)
15         C(t.term) ← (C(t.term) \ {c}) ∪ {c_1, c_2}
16       end if
17     end if
18   end for
19 end
```

The distortion $D(c)$ of clusters is evaluated using a squared error distortion measure:

$$D(c) = \sqrt{\frac{\sum_{i \in c.reg} (c.loc - i.loc)^2}{|c.reg| * 3}} \tag{4.1}$$

The procedure $split(c)$ simply distributes the term artifacts of c between the new clusters c_1 and c_2 and adapts their centroids using a conventional 2-Means

procedure. The threshold k defines the expected spatiotemporal hull of a normal distribution that corresponds to what could be considered an actual anomaly. For example, the average size of a city district in the spatial dimensions and a duration of half an hour in the temporal dimension would be reasonable choices.[3]

Once an anomaly emerges, existing clusters are attracted by the high density of keyword mentions, and the closest cluster will absorb the new term artifacts with its centroid moving towards the center of the anomaly. Eventually, the distortion criterion will be met and the attracted cluster is divided into one cluster covering the old data and one covering the new anomaly. Using this strategy, one can also accommodate and later identify term artifacts not belonging to any actual anomaly. Since this kind of signal noise will also lead to clusters being split, the strategy prevents clusters from moving too far away from the actual anomaly.

Aging of Centroids and Noise Cancellation

Eventually, a cluster centroid will be so far away from the current moment in time that absorbing a new term artifact will inevitably lead to the cluster being split. If this happens, the resulting cluster covering the old data turns *stale*, meaning that it has no chance of receiving any new term artifacts.

At this point, we have to chose a strategy to decide whether the cluster represents an actual anomaly. If this is the case, we can store it in the database and let it appear in the visualization. If not, it should be discarded, as it probably just covers noise. To this end, a formal definition of *anomaly* comprised of features that can often be observed in real-word events is employed. A most important observation is that a cluster will most likely correspond to an actual event, if it has both a relatively low average distortion - i.e. densely packed term artifacts - and at the same time represents a relatively high number of elements. This attribute of clusters can be expressed by a significance function

$$significance(c) = \sum_{\dot{t} \in c.reg} 1 - \min(1, \frac{\|c.loc - \dot{t}.loc\|_2}{k}) \tag{4.2}$$

Here k has the same value as the threshold we check against $D(c)$. This means that the significance criterion is sensitive to normal distributions above a certain shape and diameter and ignores artifacts that are outside of radius k.

[3] For the sake of simplicity a single threshold regarding space and time can be used. In this case, the dates and geocoordinates of the original messages have to be mapped to the unified spatiotemporal domain such that the maximum extent of expected anomalies is covered by the threshold.

Another feature that is used in the decision strategy considers the number of distinct users that have been contributing to a cluster and relates it to the cluster's size. This enables elimination of clusters generated by a single user who was posting the same terms repeatedly. A phenomenon that can often be observed, as there are several automated agents in the social media services, such as bots and spam-distributors.

If the significance of a cluster is very low or if it fails to meet other decision criteria, the cluster likely represents noise and can be discarded once it becomes old. In contrast, clusters receiving a high significance rating and meeting all decision criteria can permanently be stored and removed from the active computation. A simulation run of the complete clustering procedure can be seen in Figure 4.5. Here the process of cluster elimination can also be observed. For the simulation, an artificial test set of 12 randomly placed clusters with varying Gaussian message distributions was used.

In an implementation, a very large FIFO queue (e.g. 300.000 entries) can be employed to manage active clusters and discard old ones. The cluster sets $C(t)$ of the presented algorithm can then just store pointers to these representations. Every time a cluster is updated using the $handleMsg(T_m)$ method, the corresponding centroid is removed from its position in the queue and inserted again at the beginning. By simply using a very long queue one can expect that only stale clusters arrive at the end of the queue and once a centroid reaches the end, it is evaluated and then either persistently stored or discarded.

In addition to the centroid position, represented term, and significance score of each anomaly cluster, one can also include additional information to enhance the visualization. For example, for each centroid one can use the associated messages $c.reg$ to generate a temporal histogram of messages that is divided into equal bin intervals (e.g. hours). To achieve scalability in terms of storage space, one can then discard the regions $c.reg$ of the clusters before transferring them to persistent storage.

To evaluate the performance of an actual implementation, several tests using a batch analysis of 400.000 Twitter messages have been performed during this thesis. The results showed that on a powerful architecture, as was introduced in Section 3.6, a message is processed in about 0,16 milliseconds on average. On this machine, the $handleMsg(T_m)$ method could thus process roughly about 540 million messages per day. Therefore, the algorithm could easily cope with the full daily twitter corpus of 500 million Twitter messages, assuming they would be provided with geolocation. Moreover, the architecture of the algorithm can easily be parallelized, as the terms are handled independently and could thus be distributed to different threads.

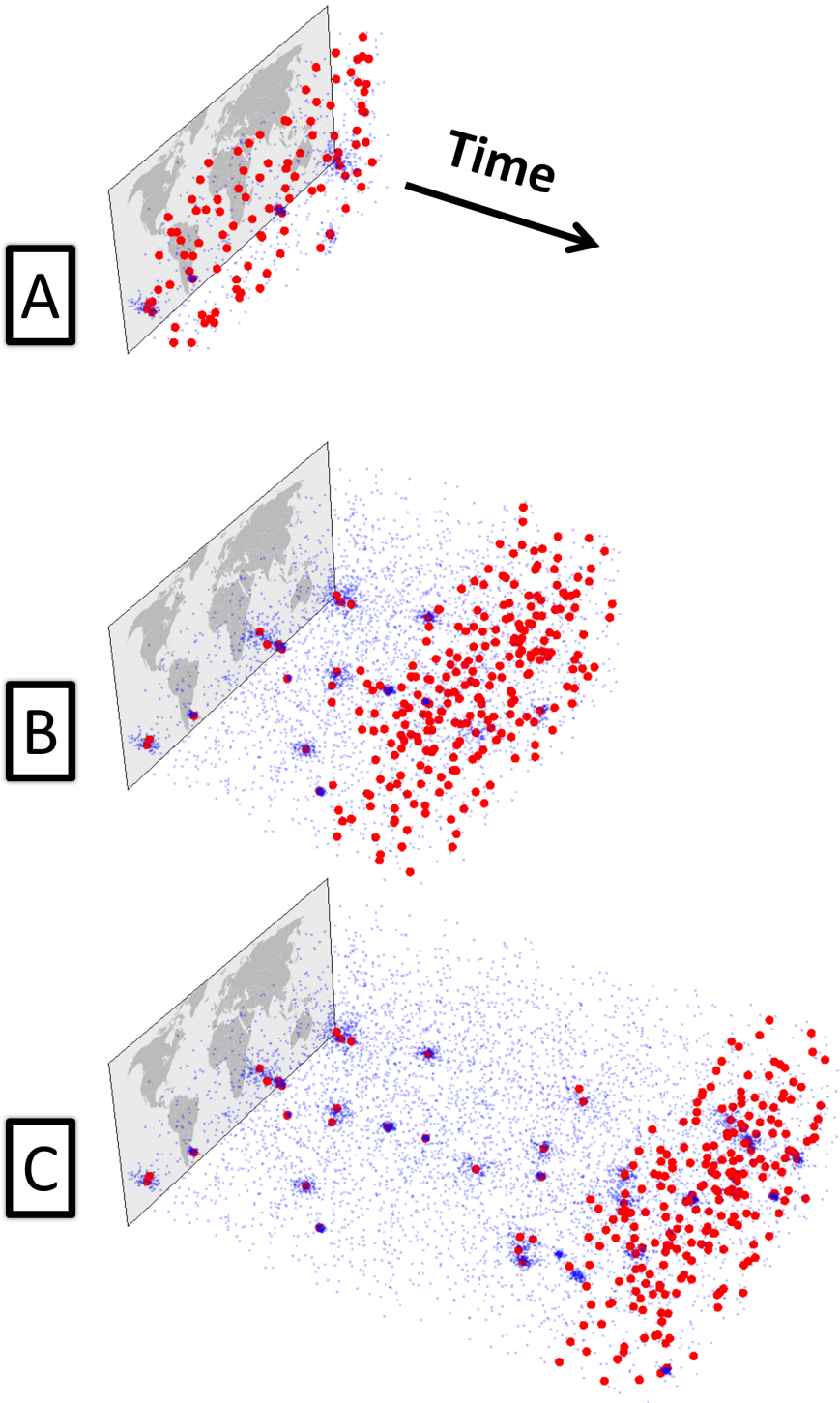


Figure 4.5 — Simulation of the stream clustering for an artificial test set with 30 Gaussian events (random $\sigma^2 \in [0,1]$) after A) 1000, B) 4500, and C) 8300 insertions. It can be seen how new centroids are created along the temporal axis. Eventually, each cluster is covered by at least one centroid.

4.3.3 Adaptive Visualization

The clustering activity results in a spatiotemporal map of centroids, i.e., locations of anomalies that potentially represent actual events. Based on this mapping, the *TagMap* then generates the overview that shows the anomalies as tag clouds on an interactive geographic map. The underlying label placement algorithm uses the significance and the temporal message histogram computed during cluster analysis to decide on the importance of individual anomalies. To this end, a weight is assigned that determines the label size logarithmically. Computing these weights and placing the labels comprises six steps:

1. Filter the anomalies according to the selected geographic area and timeframe
2. Calculate an initial weight for each anomaly having an overlap of its temporal histogram with the boundary of the selected timeframe
3. Determine the desired location and size of the labels based on the computed weights and the centroid locations of the anomalies
4. Aggregate the weights of overlapping identical terms and remove the ones with lower weights
5. Sort the labels by their weight to guarantee that significant anomalies are placed first
6. Incrementally place all labels and try to find the best available fit for each

In step (1) of the algorithm, a filter is applied to reduce the set of relevant anomalies to those located within the chosen timeframe and visible map area. This reduction can also be used to display less significant anomalies that would otherwise be dominated by more important ones in larger temporal or spatial frames. However, by considering only the centroids that lie precisely within the chosen timeframe, we would ignore anomalies extending into this range from the outside. To avoid this, the temporal histogram calculated during clustering is used to estimate the amount of messages lying in the intersection.

Step (2) thus generates an initial weight for each anomaly by summarizing the parts of the histogram covered by the intersection, which leads to a value between 0 and 1. Subsequently, this value is multiplied by the base significance of the anomaly and then used to assign an initial label size and position to it. However, particularly in zoomed-out views, there is a high probability that cluster centroids are located in close proximity to each other, which could lead to occlusion. For example, when different anomalies are co-located at the same city

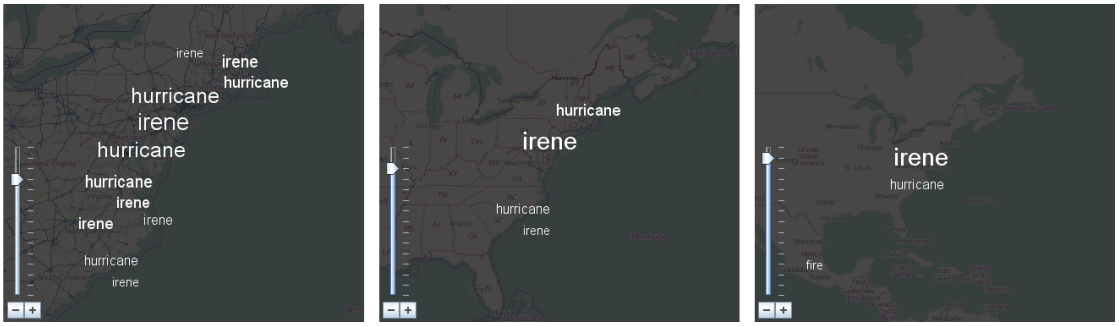


Figure 4.6 — Label Aggregation and zoom Levels. By moving from higher zoom to lower zoom we can observe how spacious distributed anomalies get aggregated into a big label representing what happens in the area. This way the overfitting that was generated by clustering can be exploited to easily align label distribution with zoom levels.

or when people reporting about an event use different terminology. According to Lohmann et al. [2009], circular tag cloud layouts are a suitable means to mediate multiple relevant terms to users. In order to avoid the occlusion, while at the same time conveying more than just the most important anomaly, it thus seems reasonable to rearrange the less important terms using this kind of layout around the target location. This is achieved in steps (3), (5), and (6):

In step (3), the initial label size is determined by normalizing the computed weight using a logarithmic function along with minimum and maximum size boundaries. After their font size and spatial extent are fixed, the labels are sorted according to their weight in step (5). In step (6), a tag cloud mechanism is used to finally arrange the tags. It is inspired by Luboschik et al. [2008], who allow a high-performance labeling of dense point clouds while, at the same time, maximizing the utilization of available space. Starting with the most important anomaly, each label is first placed at its desired location. If the required space is already occupied by a previously placed - and thus more important - label, the nearby area is explored by moving outwards in a circular fashion until a free spot is found or a maximum number of iterations is reached. In the latter case, the label will not be used, since the label position would be inappropriately far from the location of the anomaly.

Label Aggregation and Semantic Zoom

Since the clustering algorithm is designed as a best effort method, an overfitting of data is likely to occur, at least for anomalies of a certain magnitude. That is, clusters that are large in the spatial and/or temporal dimensions might be covered by multiple centroids. The primary reason for that is that the maximum

magnitude of message distributions fitting under threshold k is unlikely to match the one of each and every event.

This effect has both positive as well as negative effects. When zooming into the map, the spatial extent of anomalies can be better represented by multiple identical labels covering the affected area instead of one big label in the middle - e.g. when examining riots taking place in several parts of a city we will see several `riot` labels exactly covering the affected regions. However, at the same time, this leads to a problem in zoomed-out views. If a very important event is distributed over space and time, it is likely to be covered by many centroids having a relatively low significance value. Therefore, each of them could easily be dominated by a less important anomaly possessing a single centroid of relatively high significance.

The *TagMap* counteracts this effect by step (4) of the layout algorithm, which was not yet explained. Here the algorithm looks for overlapping labels and allows identical labels to supplement each other by accumulating the initial weights and removing the less important one. By this means, spatially distributed anomalies have a good chance of constituting a stronger label, which properly represents their importance. Of course, this principle not only applies to the spatial domain but also to temporal overlaps within a selected time frame. If there happens to be one long-lasting continuous event in the same location, it is likely to be covered by multiple clusters distributed over time. But since the corresponding labels would also overlap each other in the spatial domain, they would be aggregated inside the timeframe. Therefore, the importance of the anomaly concerning the selected timeframe can be properly reflected.

The combination of both effects - multiple anomalies per event and aggregated labels - works as a semantic zoom which aggregates anomalies in zoomed-out views and splits them to their individual locations when zooming in. This process can be seen in Figure 4.6.

4.3.4 Case Studies

To demonstrate the applicability of the approach, the *TagMap* was tested in the context of various real-world case studies based on actual Twitter data. This section illustrates how the technique is employed in such analysis sessions. Due to the prominence of the analyzed events, findings can be verified through later media cross-checks. All considered events were selected based on their relevance from an analyst's perspective, who aims at gaining situational awareness.

Earthquake hitting US East Coast

On August 23, 2011, the US East Coast was struck by a magnitude 5.8 earthquake.[4] The timeliness and distribution of tweets related to this particular event were already shown in Section 2.2.2. The event was prominently present in the messages because earthquakes of that intensity rarely occur in this part of the US. However, if an analyst had not searched for tweets containing the keyword `earthquake`, he or she would not have noticed that something unusual was going on. As the data is obscured by random daily chatter, the peak resulting from total increase in message volumes can hardly be measured.

With activated *TagMap*, the earthquake clearly stands out through prominent labels on the map, and examining lower zoom levels provides an overview of the places where people were tweeting about it (Figure 4.7). Additionally, interactive time-browsing gives an impression of the increase in related message volumes over time. It shows the typical distribution of a sudden, unforeseeable event with large labels in the beginning and smaller remote labels in the end due to citation by external observers. By further zooming into the map, smaller sub-events related to the larger event appear on the display. They indicate that buildings were heavily shaking, and that people conducted evacuation measures. The example also demonstrates the usefulness of allowing to switch between various degrees of NLP-preprocessing. In the default configuration, the prominent labels `cleared` and `evacuated` indicate that the people refer to evacuation measures that had already been *completed*. With stemming, which can be optionally activated, the tags read `clear` and `evacuate` and thus render the current state more unclear.

Nonetheless, to assess the severity and context of these events, it is still important to provide analysts with means to actually investigate individual message contents by selecting the tags. While the *TagMap* indicates anomalies and helps to come up with ideas to comprehend the situation, analysts still have to check exemplary messages in order to verify initial conceptions.

Besides the applicability of the technique, this example also highlights the dangers of misinterpreting or overlooking relevant information. For one thing, it quickly becomes clear that the *TagMap* not only shows event-related information, but also various irrelevant tags that result from arbitrary clusters. Additionally, one can also observe that the largest labels can be found in larger cities along the coast instead of the areas where the earthquake might have been first felt or were it was most severe. Obviously, the reason is that higher densities of Twitter users also create larger clusters, which has a significant impact on label sizes.

[4] http://earthquake.usgs.gov/earthquakes/eqinthenews/2011/se082311a/

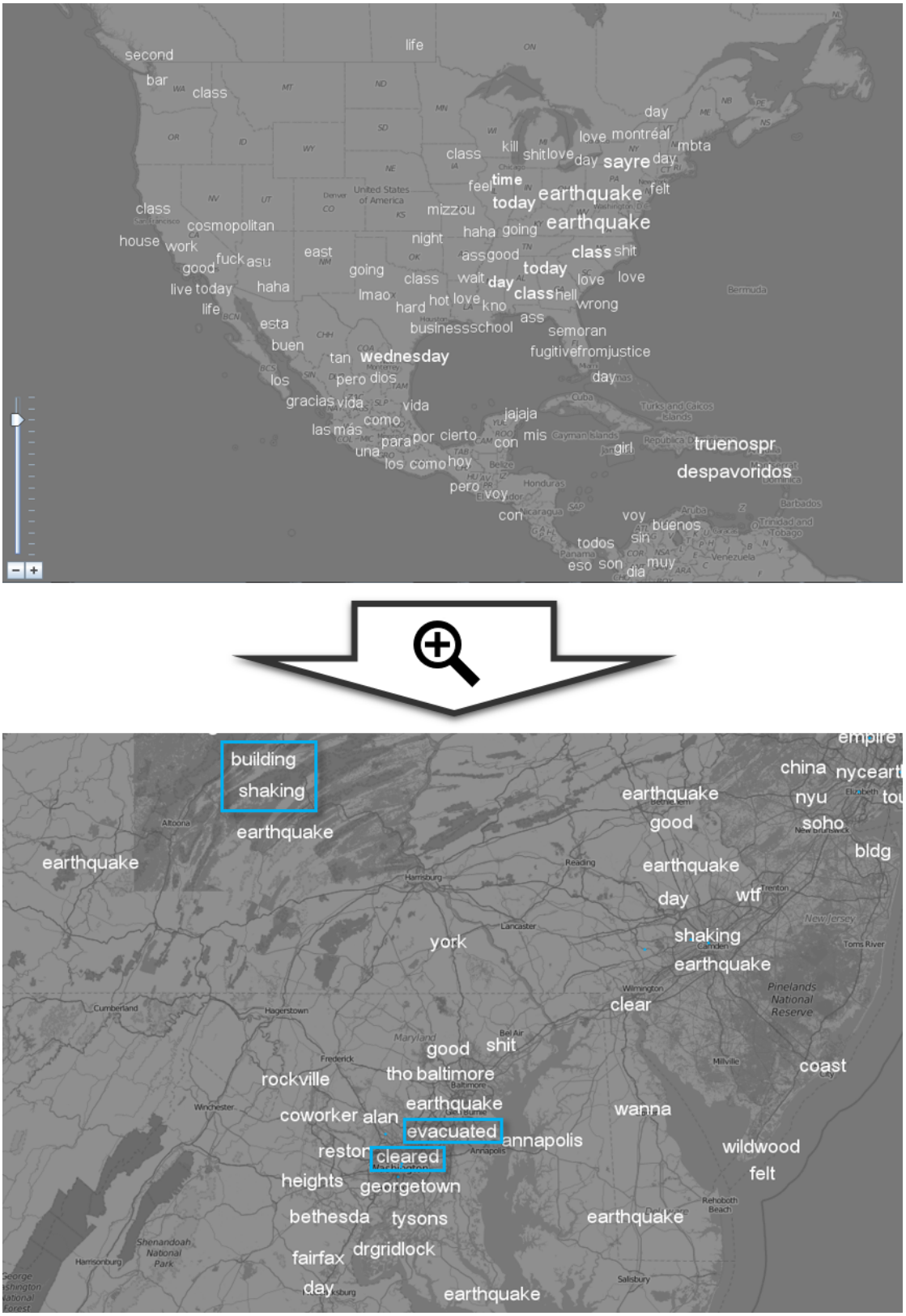


Figure 4.7 — *TagMap* analysis of the Virginia Earthquake. The upper screenshot shows an overview of the situation in the US on August 23 2011. The lower screenshot is a zoomed-in view of the East Coast area. Tags referring to evacuation measures are highlighted by the manually added blue boxes.

► **Figure 4.8 —** *TagMap* analysis of the London Riots. If display of frequently mentioned geographic entities is activated, Hackney and Peckham are highlighted in the overview because they were severely affected districts. The overview also highlights the police activities in the city. By zooming into the map, the labels are automatically split and distributed to individual hotspots.

These two issues will be addressed in Section 4.4, where population-density normalization and language-based normalization will be discussed.

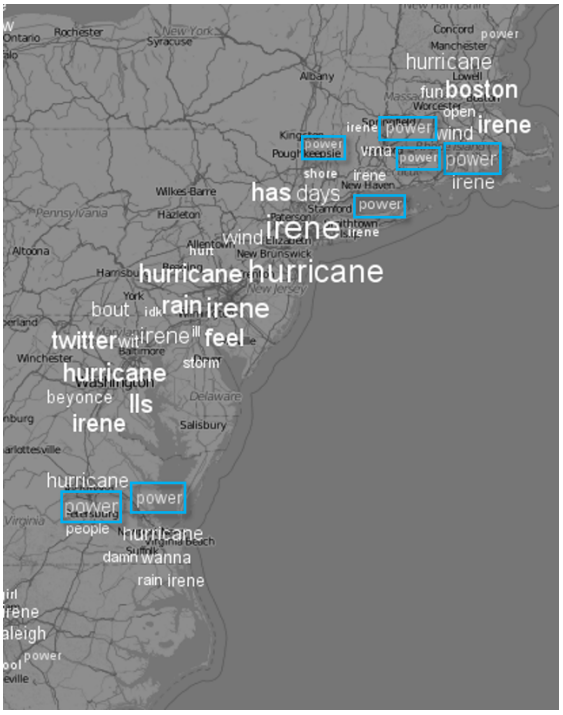
London riots

The second case study investigated the situation in London during the London Riots[5] on August 8, 2011. Here the *TagMap* overview shows that considerable numbers of tweets refer to the event, which is apparent from the very prominent label `londonriots`. Additionally, the names of some London districts, such as Hackney and Peckham, which were most severly affected by the riots, appear very salient. By zooming into the map, previously aggregated labels such as `riot` are split up to depict the individual locations of hotspots. Selecting some of the messages, all sent between 17.30 and 18.30 UTC in Hackney, indicates that the situation was escalating. Police forces seemed present at the scene, but the area appeared insecure to various observers. Tweets covered by the clusters read:

- **17.32 UTC:** *Great I,m stuck in the middle of the riots in hackney..*
- **18.07 UTC:** *Just leaving #bethnalgreen it's mad here. Deffo STAY OUT of the area. #riots #London [...]*
- **18.27 UTC:** *Fuck riots in Bethnal green seen fire cabt get home*

Similar procedures can be applied to other regions of London where labels indicate riots and looting. In the described case, it is helpful to have the district names as indicators for where to look for hotspots.

[5] http://en.wikipedia.org/wiki/2011_England_riots



◄ **Figure 4.9** — *TagMap* analysis of Hurricane Irene. Power outage related messages can be seen in the Richmond area and around Rhode Island. In later press releases these outages were confirmed by the power company. The blue highlight boxes were added manually.

Hurricane Irene

The last case study focused on Hurricane Irene, an event that is different from the previous two as it was already anticipated by weather experts and communicated to the public through news media. The hurricane struck the US East Coast in the morning of August 27, 2011.[6] Some coastal areas were already evacuated before the storm reached the mainland. However, it was unpredictable how severely it would affect East Coast cities. In this case, browsing through temporal ranges reveals that several regions are labeled by the *TagMap* with the term `power`. As this might be related to power outages, it is relevant to find out whether these are local or regional impacts of the storm. By taking a closer look at Richmond, Virgina, which has several `power` labels around it, and selecting one of the labels in this area, one can confirm by the messages that the event is indeed related to outages caused by the storm. This was also confirmed by the regional power supplier[7]. The *TagMap* thus allows to assess the severity and distribution of power outages on the map based on the presence and size of labels. However, these labels first have to be perceived by the analyst by recognizing their semantic abnormality compared to irrelevant ones surrounding them.

[6] `http://en.wikipedia.org/wiki/Hurricane_Irene`

[7] `http://wtvr.com/2013/08/27/richmond-revises-emergency-plan-after-hurricane-irene/`

4.4 Term Relevance Normalization

The case studies demonstrate that the *TagMap* is a powerful means to get an overview of the situation and discover relevant events. However, it can also be observed that the visualization produces visual clutter in form of several irrelevant tags. This can severely irritate analysts - particularly when using it for the first time - and hinders straightforward identification of relevant entities. The primary reason for superfluous visuals in the *TagMap* is that it does not consider semantics. From the algorithm's viewpoint, the spatiotemporal cluster resulting from the topic `class` during morning hours in schools can have the same significance as the ones resulting from the topic `earthquake` in the evening. They are both shown in the visualization in the same manner. Relevant anomalous keywords can thus be obscured by terms resulting from day-to-day chatter. However, there are certain characteristics of these terms that can help to discern them from more critical events. Most importantly: they are not unexpected or unusual.

This section investigates a model to evaluate whether terms are actually anomalous outliers or just perennially prominent terms frequently used in the examined region. Usually, tools like *tf-idf*, which was already featured in chapter 3, or similar measures are used in this type of tasks. They quantify whether a keyword is specifically relevant for a selected document set, or whether it is similarly frequent within the whole corpus of documents. However, when dealing with geo-referenced documents, these measures are not the optimal choice. Besides globally prominent terms, which can easily be identified by *tf-idf*, there are also many terms that are only frequent within a given region. Thus, when examining a certain geospatial area and timeframe of documents, the importance of certain prominent and unusual terms can still be obscured by frequently mentioned terms that are rather common just for the area.

To address that challenge, this section extends the *TagMap* with a scalable measurement technique for geospatial term relevance normalization. To this end, the basic *tf-idf* notion is combined with kernel density methods. A large corpus of recorded messages serves as a basis for a geographic language model that determines the a-priori probability that a given term is contained in a document composed at a given location. This follows a simple intuition: In the *idf* part of the measure, the number of documents in which a term appears is put in relation to the sum of documents in the corpus. Accordingly, the presented measure sums, for any given location, the derived probabilities that a document containing the term could have appeared at this point and puts it in relation to the sum of derived probabilities that any document could have appeared there. The outcome is the (im-)probability that a term is contained in

a message appearing at the given point, thus allowing to assess the abnormality of observed term occurrences in examined document sets.

An important part in the realization of this technique is a scalable method to calculate, store, and quickly retrieve the normalization values based on adaptive grid aggregation techniques. This method will be described in Section 4.4.2. The general performance of the measure in terms of precision and recall is then assessed in Section 4.4.3. Section 4.4.4 concludes with an application example, in which we will see how the *TagMap* is improved by visually highlighting unusual terms.

Although primarily developed to improve the *TagMap*, the technique can also be applied to all other forms of exploration tools that benefit from term weighting based on geographic regularities. For example, the exploration lens of *TreeQueST* (see Section 3.4.4) can also be applied to geographic message exploration, as will be shown in the *ScatterBlogs* chapter. Here one can use the technique to enable a relevance-weighted ordering of terms.

4.4.1 Geo-aware *tf-idf*

To illustrate how the metric can be constructed, it is useful to take a closer look at the idea of *tf-idf*. Common forms of such measures evaluate the relevance of a term for a given document in context of a given corpus [Jones, 1972; Manning et al., 2008]. To this end, the standard *tf-idf* works as follows: Given a document d and a term t, the number of occurrences of t in d ($=TC_{t,d}$) is determined and normalized by document length to compute the *term frequency* $tf_{t,d} = \frac{1}{|d|} TC_{t,d}$. It thus measures the relative prominence of t within d. A high *tf* value alone, however, does not indicate that the term is specifically relevant to the document, as it could be frequent in the whole corpus. Therefore, one additionally computes the *inverse document frequency* based on the corpus D of all documents, from which d was taken:

$$idf_{t,D} = \log \frac{|D|}{|\{d|d \in D \wedge t \in d\}|} \tag{4.3}$$

The *idf* then serves as an indicator for the a-priori probability that t appears in documents drawn from D - the higher the probability, the lower the value. The *tf-idf* is finally computed by multiplying the values:

$$tf\text{-}idf_{t,d,D} = tf_{t,d} * idf_{t,D} \tag{4.4}$$

In the domain of geolocated messages, we find very diverse contents ranging over regionally prominent topics, local characteristics, and specific language

use. For example, in most larger cities, the city's name as well as the names of individual districts are continuously mentioned in hundreds of messages. Because of the day/night-cycles in message frequencies, the *TagMap* might then falsely recognize one anomaly for each of these terms every day. At least, if they do not happen to be on the stopword-list. The same is true for regional points of interest or words from languages and dialects that are only used in distinct parts of the world. In order to estimate, whether a term is actually important or anomalous for a particular geographic area and timeframe, it is not sufficient to compare its local term frequency against the global *idf*. Instead, we need a measure that compares the term count of the examined message set with the estimated *idf* of all messages that have been written in the region. In order to allow this for arbitrary geographic points, kernel density estimation (KDE) can be used to move from term *frequencies* to geographic term *densities*. By this means, we can approximate the probability distributions from discrete point data and integrate them with *tf-idf*.

Kernel Density Estimation

There are well-known means to derive a continuous probability density f from a finite sample dataset $X = \{x_1 \dots x_n\}$. For example, X could be a list of locations of crime reports in a major city. With KDE [Rosenblatt, 1956; Parzen, 1962], the so-called density estimator for the dataset is constructed as follows:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{d(x, x_i)}{h}\right) \tag{4.5}$$

In this equation, $d(x, y)$ is a distance metric, e.g., the Euclidean distance in case of samples from $\mathbb{R}^n$. The function K is the kernel and it is used together with the bandwidth h to assign a weighted value to each x_i, depending on its distance from x. For the sake of simplicity, one can also write the equation using subscript notation $K_h(u) = \frac{1}{h}K(\frac{u}{h})$. Common choices for kernel functions are

- **Gaussian** - $K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right)$
- **Cauchy** - $K(u) = \frac{1}{\pi\left(1+u^2\right)}$
- **Epanechnikov** - $K(u) = \frac{3}{4}\left(1 - u^2\right)\mathbf{1}_{|u|\leq 1}$

All these functions have their maximum at $u = 0$, are rapidly decreasing with higher u, and integrate to 1. In the following definitions, it is assumed that a Gaussian kernel is used, but other choices will lead to similar results.

Ultimately, the purpose of using these functions is to reflect the probability that a given sample could deviate a given distance u from its actual location. For example, if a crime was committed somewhere along a street at point x, without prior knowledge, it could as well have happened 5, 10, or 100 meters further away - yet with decreasing probabilities. By summing and normalizing all these spatially decreasing probabilities from all samples - i.e. all crimes that happened in the city - the constructed function $\hat{f}(x)$ gives us a relative estimate of the a-priori probability that a crime can happen at any given location x in the city.

In this work KDE serves as a basis to assess numeric long-term densities. However, in other visual analytics approaches the technique is also used to directly visualize spatial probabilities as kernel-smoothed heatmaps [e.g. Maciejewski et al., 2010].

Measure Definition

Let m be a social media message that contains term t. Similar to the crime example, if the message was written at some location x, it could as well have been written further away with decreasing probability. Based on a sample of term occurrences at the respective message locations, we can thus use the KDE principle to inspire the estimation of term occurrence densities at given map locations. In the following definition, KDE is employed to establish the geospatial version of *idf* as basis for the geospatial *tf-idf* measure. As a prerequisite, it is assumed, that a large corpus G of geolocated social media messages collected over a large temporal range is available - i.e. it should be robust against seasonal characteristics. First, we have to define a measure for local term density. It is then normalized by the corresponding local document density to finally calculate the local inverse document density:

For a given term t, let $G_t = \{m \in G : t \in m\}$ be the subset of messages from G that contain t, and let $loc(m) \in \mathbb{R}$ be the location of message m in the coordinate space. Furthermore, let K_h be a kernel with fixed bandwidth h. For a given location $x \in \mathbb{R}$ we call

$$td_t(x) = \sum_{m \in G_t} K_h(d(x, loc(m))) \tag{4.6}$$

the term density of t at x. Note that these are absolute and not relative densities, as they are not normalized by $\frac{1}{|G_t|}$. In order to allow a cross-comparison of different terms, the term's densities are normalized at every location using the term independent document density:

$$dd(x) = \sum_{m \in G} K_h(d(x, loc(m))) \tag{4.7}$$

Finally, and analogous to the *idf*, we call

$$idd_t(x) = \log \frac{dd(x)}{td_t(x)} \tag{4.8}$$

the inverse document density of t at x.

In these equations, the distance function must be matched to the coordinate space that has been chosen to represent message locations. For the example in Section 4.4.1 a uniform grid coordinate system was assumed and thus the Euclidean metric was an appropriate choice. However, since GPS locations are usually given in graticule coordinates (e.g. latitude, longitude), one should transform them to a uniform grid or use the Haversine formula [Sinnott, 1984] to approximate the distance.

For the term frequency *tf*, there is a natural analogue in forms of the number of messages in an anomaly. If a set of messages M is to be examined - e.g., resulting from a detected cluster or a user-selected region and timeframe - one can build a localized *tf-idf* value for any term t by calculating the sum

$$\sum_{m \in M} TC_{t,m} * idd_t(loc(m)) \tag{4.9}$$

This equation properly reflects the relation of current prominence of the term versus its commonness at the message locations. For the sake of simplicity and computational cost, the value can be approximated by calculating the term frequency for a pseudo-document generated by concatenating all documents in M, and multiplying it with $idd_t(\frac{1}{|M|}\sum_{m \in M} loc(m))$. In case of the *TagMap*, this corresponds to multiplying the size of an anomaly, i.e., the number of term artifacts in the cluster, with the *idd* value of the term at the cluster centroid.

4.4.2 Scalable Implementation Strategies

For the traditional *tf-idf* measure, it is expensive to compute the *idf*-part, as the whole corpus has to be analyzed. Therefore, the values are usually precomputed at once by iterating through the complete set of documents and terms within the corpus. A *tf-idf*-vector for any given document can then quickly be generated by computing a term frequency value for each term $t \in d$ and multiplying it with its precomputed $idf_{t,D}$ value.

The computation of $idd_t(x)$ values is even more expensive than *idf*. For the given point x, we have to compute the sum of kernel-weighted distances between x and the location of all messages $m \in G_t$ and $m \in G$. Furthermore, as we are looking at a theoretically continuous and thus infinite coordinate space, it is not feasible to precompute and store an $idd_t(x)$ value for every possible x.

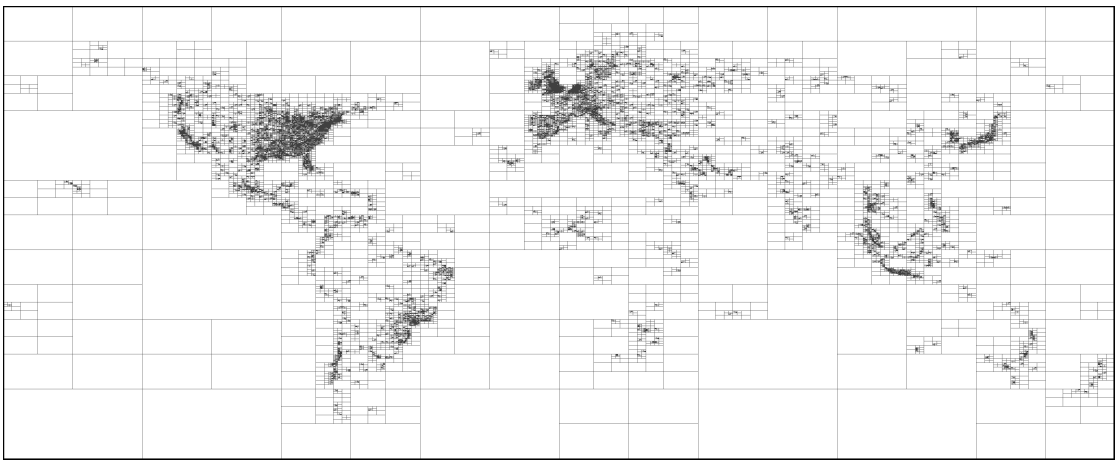


Figure 4.10 — The result of adaptive grid creation. Cells were divided if they contain more than 50 messages and up to a maximum depth of 16.

In practical applications, however, there is no need to have an infinitely high spatial resolution. Instead, we could work with a high resolution regular grid that is laid over the globe. The $td_t(x)$ and $dd_t(x)$ values could be computed at every cell center or vertex. Missing values between these points could then be calculated at runtime through interpolation. Nevertheless, to achieve high resolutions (e.g. cell-sizes below 0.5 kilometer at the equator), a grid resolution of at least 80 000 $\times$ 40 000 cells would be needed for global coverage. If we assume that each value takes 4 bytes of storage, this amounts to approximately 12 gigabytes of data for every single term in the corpus.

To tackle this problem, one can adhere to a grid construction strategy that is adaptive to regional requirements resulting from population density. This will be detailed in the next subsection. Subsection 4.4.2 then explains how the *idd* values can be quickly precomputed for each generated grid cell by approximating KDE based on grid splatting. Finally, Subsection 4.4.2 explains how the values are stored and quickly retrieved once they are needed by the *TagMap*.

Grid Creation

There are several geographic regions in the world, from where only few social media messages are produced, such as oceans, deserts or large rural areas. In contrast, the volumes of messages written in major cities by many times exceed the volumes from other populated areas. Instead of a regular grid with a fixed resolution, it is therefore reasonable to use an adaptive grid with high

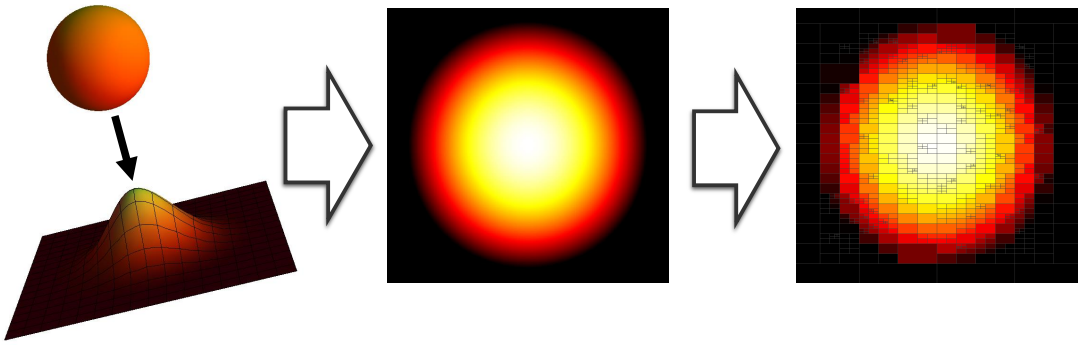


Figure 4.11 — Left: The splatting process can be imagined as throwing "ink balls" onto the grid, resulting in a Gaussian signature. Middle, Right: The continuous splat is applied to the discrete grid. Each cell value is computed based on the distance from cell center to splat center.

resolution in densely populated areas and lower resolution elsewhere. This grid can be constructed by a recursive splitting strategy, such as quadtrees [Finkel and Bentley, 1974].

For example, in case of Twitter data, the relevant "Twitter population"-density can be directly derived from message densities in the observed corpus. Figure 4.10 shows an adaptive grid that has been generated based on 10 days of recorded Twitter messages that were uniformly distributed over the world. It was generated as follows: Initially, the algorithm created a single cell that comprised the complete lat/lon-coordinate space, such that $lat \in [-90, 90]$ and $lon \in [-180, 180]$. As long as more messages than a fixed threshold fell within one cell, it was divided into four equally sized sub-cells, and the algorithm was recursively applied to to each. A recursive path was terminated as soon as a predefined minimum cell-size of 0.5 kilometers or a maximum depth of 16 was reached. For further computation, the complete recursive tree structure with the leaves representing the grid cells can be stored. This way, one can quickly ($O(\log |leaves|)$) find cells containing a given location x by recursively searching through the tree. In the following subsections, c_i is used to denote unique cell IDs in such grids, and $loc(c_i)$ refers to the center-location of a cell.

Fast Value Precomputation

In most kernel functions, the majority of the area beneath the corresponding curve is inside a bounded radius from the center (e.g. more than 99% are inside a $3h$-radius in case of a Gaussian kernel). For the computation of the $td_t(x)$ and

$dd(x)$ values, messages that are further away from x than this radius can thus be ignored as they add almost nothing to the sum. To precompute a good approximation of the KDE-density of a sample, one can thus adhere to grid splatting, a technique that basically inverts the process by accumulating bounded kernels. The concept originates from volume rendering for 3D graphics [Westover, 1991], where it was proposed as the metaphor of "throwing ink balls" onto the grid, resulting in a Gaussian footprint, a so-called *splat*. In our case, the local sums of the footprints at each grid cell add up to the *td* and *dd* values. The basic concept is illustrated in Figure 4.11.

Based on this idea, an algorithm to quickly precompute *idd* values can be defined: Let $grid = \{c_1, ..., c_i\}$ be the set of cell IDs of a grid data structure as explained in Section 4.4.2. Furthermore, let $DD : grid \rightarrow \mathbb{R}$ and $TD_t : grid \rightarrow \mathbb{R}$ be initially empty hash tables that map cell ids to computed *dd* and td_t values. A given corpus G is then processed according to the `splat`-procedure shown in Listing 4.2. Instead of iterating through the whole corpus for each grid cell, it iterates through G just once and adds a Gaussian splat value for each $m \in G$ and $m \in G_t$ to all affected hash table entries $DD(c)$ and $TD_t(c)$, if the center location $loc(c)$ is within a $3h$-distance from $loc(m)$.

Listing 4.2 — Splatting Algorithm

```
1  procedure splat(G,TD,DD, grid) is
2  begin
3    for m ∈ G do
4      impact_area ← {c ∈ grid : d(loc(c), loc(m)) ≤ 3h}
5      for each c ∈ impact_area do
6        if DD(c) =empty then
7          DD(c) ← K_h(d(loc(c), loc(m)))
8        else
9          DD(c) ← K_h(d(loc(c), loc(m))) + DD(c)
10       end if
11     end for
12     for t ∈ m do
13       for each c ∈ impact_area do
14         if TD_t(c) =empty then
15           TD_t(c) ← K_h(d(loc(c), loc(m)))
16         else
17           TD_t(c) ← K_h(d(loc(c), loc(m))) + TD_t(c)
18         end if
19       end for
20     end for
21   end for
22 end
```

As mentioned in Section 4.4.2, this "impact area" can be found quickly using the quadtree data structure. Assuming a constant upper bound for the number

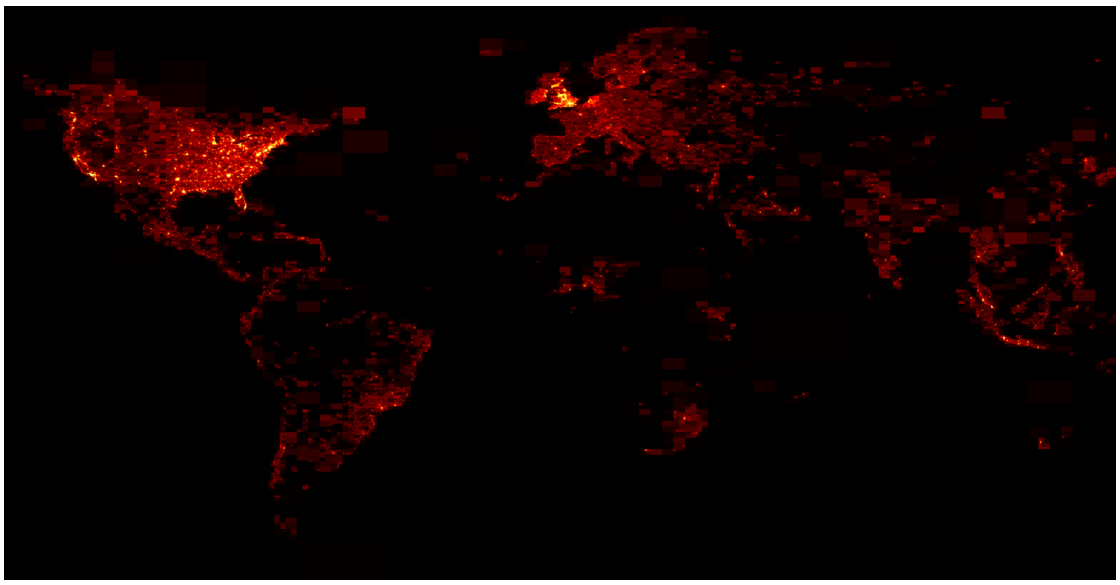


Figure 4.12 — The image shows an example result of adaptive grid splatting for the term `love`. The visualization maps *td*-value to a color-palette ranging from black for low values, over red for medium values, to yellow for high values. It can be recognized that major cities receive higher volumes of term-mentions, and that few mentions happened in low-resolution areas.

of terms inside a message as well as the number of cells within a splat radius, the algorithm's runtime can be estimated by $O(|G| * \log(|grid|))$. Also, in terms of memory management the hash tables provide an efficient means to store the data, since large volumes of grid cells in oceans and rural areas will be unaffected by the splats of most terms. For these areas, the redundant information $idd_t(c) = 0.0$ does not need to be stored. An example result for TD_{love} can be seen in Figure 4.12. To limit the length of the precomputation phase, it is reasonable to restrict it to terms that have a certain minimum frequency. For example, one can only include terms that have at least a certain amount of mentions within a year. In the application, all other terms are then handled as if they occur for the first time by assigning a default minimum term density of $td_t(x) = 1 * K_h(0)$.

Fast Value Retrieval

The output of the splatting procedure will be a large set of filled hash tables TD_t for each term and DD for all documents. These tables can now be used for ad-hoc interpolation of the $idd_t(x)$ values at given points on the map. To actually apply the measure, different modes can be chosen. In case of the

TagMap, one way is to compute the *idd* weights for the labels once they are placed in the visualization. In this case, the retrieval should be fast enough to allow fluent interactivity.

However, to achieve more accurate results, one could also retrieve the values for each individual message and compute averages for the aggregates. In case of Twitter, a current maximum of 10 million messages would have to be handled per day if only geolocated data is considered. Assuming that the number of specific terms (i.e. no stopwords/urls/usernames) per message can be bounded by 2 to 4, this means that the computation of one $idd_t(x)$ value should be achieved in between 2,1 and 4,3 milliseconds.

4.4.3 Measure Performance

If integrated visual analytics solutions involve novel data processing algorithms or modified algorithms that were tailored to a given challenge, they can be detached from the complete system to be evaluated based on quantitative performance indicators. This method resembles the process of unit testing [Runeson, 2006] in software engineering. Based on labeled ground-truth data, the algorithm can be compared to existing algorithmic solutions or to means of solving the problem without algorithmic support. This approach was chosen to evaluate the *idd* measure as it has a measurable input-output behavior, ground-truth data can easily be generated, and the method can be compared to less specialized solutions. In this section, it is thus evaluated how the measure performs on the more general information retrieval task of finding anomalous terms in Twitter message sets.

As scalability of the precomputation process and fast retrieval times were a key requirement of the developed algorithm, the section first discusses computational performance of a reference implementation.

Computational Performance

To benchmark the *idd* algorithm, it was prototypically implemented and tested on the same powerful machine that was also mentioned in previous sections - i.e. four Intel Xeon processors totaling to 40 physical cores, 128GB RAM, and SAS hard drives in a RAID 50 configuration. The *idd* values were precomputed based on approximatelly 730 million geolocated Twitter messages collected between August 2011 and August 2012, which is a subset of the data described in Section 2.2.3. The splatting algorithm was implemented to allow parallel execution to fully benefit from the multicore capabilities of the test system. For a maximum depth of 18, the adaptive grid was created in less than 30 minutes and has about 300,000 cells. To further limit the total computation time, terms

that occurred less than 1000 times during the year were ignored, as described in Subsection 4.4.2. Based on this configuration, the complete precomputation process for the *td* and *dd* tables was performed in less than 35 hours, and it took approximately 200 gigabytes to store the raw output. This demonstrates that the solution easily scales to very large datasets and that the *idd* dictionary could also be updated on a daily basis.

A set of 1000 terms, drawn according to their overall frequency in the corpus, as well as 1000 randomly chosen cells of the grid were used to measure the retrieval speed of different storage solutions. The adaptive grid tries to keep the amount of documents in each cell relatively constant; therefore the random cell node selection roughly reflects the document distribution. Because all terms share the same grid, only one instance of the quadtree needs to be held in memory for computing the cell ID for a given point x. The actual data is stored as mappings from cell IDs to term density values.

First, a straightforward approach was tested that stores the map for each term in a separate file, resulting in file sizes up to 8 megabytes for the most popular terms and much less for others. The content of the file is sorted by cell ID to allow for fast failing of the search if a value does not exist. The results show that accessing a stored $td_t(x)$ value takes 213 ms on average, with 221 ms for hits and 149 ms for misses.

Secondly, Apache Lucene[8] was used as an alternative approach for large-scale indexing. Each combination of term, cell ID, and *td* value is indexed by Lucene as a standalone *document*.[9] In order to access a value, the index for the term and cell ID combination is used to retrieve the document containing the value. This process takes 46 ms on average, with 48 ms for hits and 35 ms for misses. Using parallelization, which was tested with twenty cores, the described implementation achieves an average of 4.8 ms per value retrieval, thus almost matching the performance requirement to process all geolocated tweets per day. However, because of the simple means to parallelize the algorithm, the speed can be further increased to fully match the requirements.

Test Setup

For the information retrieval evaluation, the location-aware *idd* measure was tested against raw frequency of terms and global *tf-idf*-based term rankings. To this end, two past events, the 2012 Comic-Con conference[10] and the already

[8] `http://lucene.apache.org/`

[9] Lucene stores documents as individually indexed string or numeric fields.

[10] `http://en.wikipedia.org/wiki/San_Diego_Comic-Con_International`

investigated 2011 London Riots were chosen. Twitter data from corresponding timeframes and areas was extracted.

This data was then manually labeled by deciding for each of the 1000 most frequently appearing terms in the tweets whether they can be considered specifically relevant to the respective event. For example, in case of the riots, terms like `police`, `looting`, or `burning` were labeled with `YES`, and terms like `london`, `people`, or `think` were labeled with `NO`.

Following the labeling, the three measures raw frequency (*tf*), *tf*idf*, and *tf*idd* were computed to rank the terms accordingly in three respective result lists. The retrieval performance is evaluated by interpolated precision/recall-curves, which is a common IR-measure to assess ranked retrieval results [Manning et al., 2008, pp. 145-150]. Terms of a computed ranking are incrementally added to a retrieval set, and the resulting development of precision versus recall is plotted. The plot illustrates the volumes of correct hits in the top-ranked results as well as the drop in accuracy until the complete set of relevant items is retrieved.

Retrieval Results

The diagrams in Figure 4.13 show the experimental results. It can be seen that the location-aware *tf*idd* measure performs better in terms of precision for the top-ranked terms and equally well or better for the lower ranked terms. In case of the Comic-Con, the measure shows almost perfect precision until the first 20% of the relevant items are retrieved. As there were 96 relevant items, this result indicates that the 19 top-ranked results are all considered relevant in the ground-truth.

Such a result is specifically important if the measure is used as term ranking tool for visual aggregation schemes like the *TagMap*. Usually, these tools find hundreds of significant terms that could be used to represent data in a given geographic area. In such cases as the Comic-Con, the *tf*idd* ranking can make sure to show and/or highlight just the terms that are specifically relevant during the event.

It can also be seen that there are noticeable differences for the two events. While the measure performed significantly better than the other measures for the Comic-Con event, the difference is slightly less distinct for the London Riots. This can be explained by the very high amount of proper names used during events like the Comic-Con, such as artists, related venues or activities. During the convention, these names experience a significant increase in usage compared to other times. Certain terms like `police` are overall more common in normal

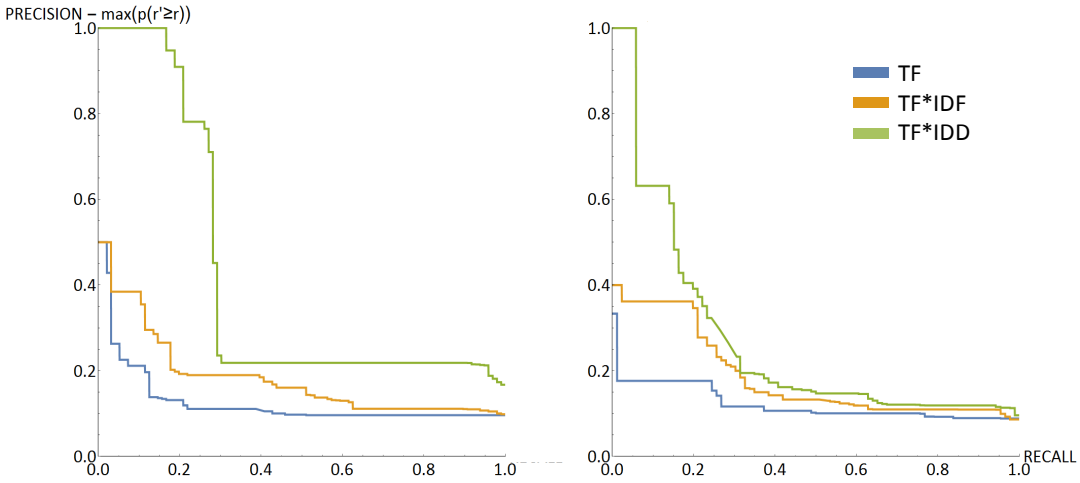


Figure 4.13 — Left: Interpolated precision ($p_{interp}(r) = \max_{\dot{r} \geq r} p(\dot{r})$) versus recall for the Comic-Con 2012 event. 96 terms were labeled relevant. **Right**: London Riots 2011. 88 terms were labeled relevant.

usage, but there is still a significant change in ratio that is detected by the metric.

As the available screen space usually limits the number of terms that can be displayed or highlighted at the same time, the first 10 to 20 retrieval results are most important for map-based aggregation. The superiority in these upper ranges thus justifies the usage of *idd* compared to other measures.

4.4.4 *TagMap* Integration

After having evaluated the statistical performance of the measure, we can now have a look at the actual applicability to support event weighting and term highlighting in the *TagMap*. It has already been indicated that there are different ways to employ the *idd* values here. To generate aggregated values, we can either compute the values already at the locations of individual term artifacts and later build averages of all items in a cluster, or we can just compute one *idd* value at the cluster's centroid and multiply it by the cluster's size or significance score. These values would then again have to be aggregated if multiple clusters are merged in the visualization. While the second method is significantly faster, it can also be less accurate, especially if clusters are displaced from existing term-frequency hotspots. However, experiments have shown that this rarely happens if the splitting parameters are well configured.

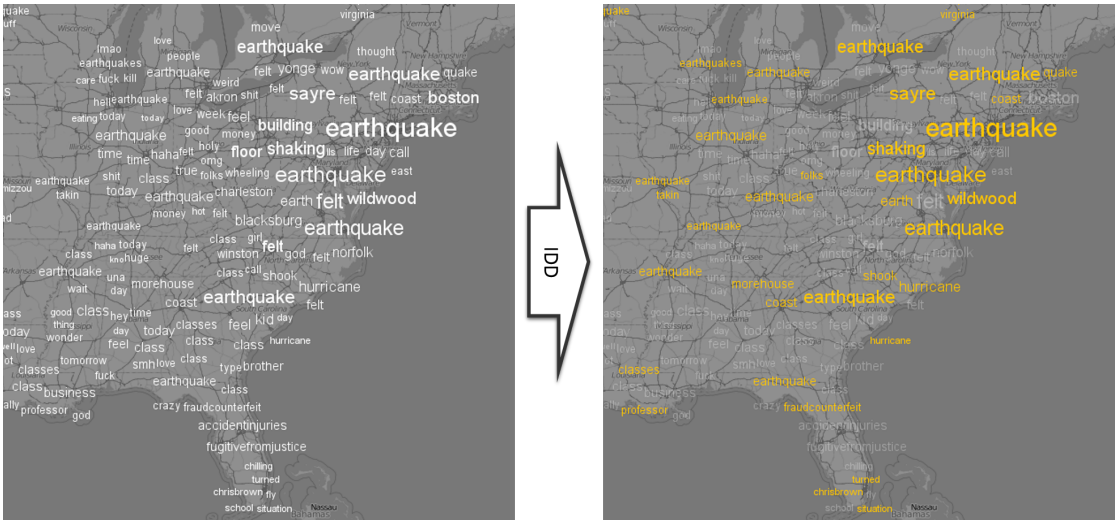


Figure 4.14 — *TagMap* enhanced by *idd* evaluation. The right image shows how sudden, unusual terms related to the event are highilighted while others are grayed out.

The result of applying the second method to the Virginia earthquake case can be seen in Figure 4.14. In this case, a simple color coding was used that highlights topic clusters with high average *idd* values in orange. The threshold for highlighting can be computed based on the z-score of the *idd* value for all clusters. However, the analysts can also configure it to their needs using interactive controls (e.g. if nothing is highlighted, they can incrementally lower the threshold to get indications). One can observe that the relevant terms, like `earthquake` and `shaking`, which are related to the sudden event, immediately stand out, while other prominent terms are grayed out.

By further exploring different time periods on this day, one can also find other events that were not easily identified with the plain *TagMap*. They include a larger concert that happened in the Indianapolis area (e.g. performing artists `britneyspears` and `djpaulyd` are highlighted) and a Drupal conference that was held in the UK. Although these are not critical events, they are highlighted because the corresponding term usage frequencies are unusual for the area. Whether it is important or not to further investigate such mass gatherings depends on the analyst's use case.

► **Figure 4.15** — Different events during August 23 that can be easily discovered with enabled *idd* normalization. The first two images show the most prominent tags with and without *idd* support if the complete time range is selected. The third and the fourth image show the situation with activated temporal filters for two different timespans during the (UTC-)day. One can now spot the earthquake as well as the concert and the Drupal conference through the color highlighting.

CHAPTER 5

Task-Adaptive Detection and Drill-Down

While traditional web search is mostly precision optimized, analysts in situation awareness also have high requirements for recall (see Section 2.3). Relevant reports with low frequency, such as the two or three messages informing us that an embankment has been broken during a flood disaster or that a building has collapsed must not remain undetected. However, if the time to investigate large volumes of possibly relevant messages is just not available, methods are needed that allow for an acceptable trade-off between both, precision *and* recall.

While the last chapter demonstrated how events and related entities can be indicated without prior information, this chapter investigates what we can do if we already know what might be going on and want to reliably monitor or drill down on information according to previously identified topics. In principle, Chapter 3 showed how recall-optimized queries for retrieving and filtering data can be created. However, precision can be increased at the same time once the pre-filtered data was retrieved to allow the application of more powerful algorithms. In this case, keyword queries often fail to deliver perfect results. They are usually either too broad, as they contain generic or ambiguous terms, or too narrow, as they lack vital terms that were not considered.

A common solution to achieve both, high precision and recall, is to apply supervised machine learning. Various researchers have already employed such models in the realm of social media data to categorize messages with regard

to their relevance [Aramaki et al., 2011], textual content [Dilrukshi et al., 2013], and sentiment [Go et al., 2009]. In these approaches, linear binary classifiers, such as *support vector machines* or *naive Bayes* play a particular role, as they are easy to handle and evaluate, usually have few or even zero configuration parameters, and - most importantly - are fast enough to cope with the data volumes in real-time. Based on a comprehensive corpus of recorded messages, such models can be trained via ground-truth labeling or active learning (AL) to reliably detect and classify messages with accuracies that usually outperform keyword-based approaches.[1]

It is a problem, however, that once trained, these models are usually static in nature, and they are applied without letting analysts understand their true behavior. In the fast-paced world of situation awareness, analysts can incorporate filters and listen to alerts, but they have no clue what the model is actually detecting, what they might be missing, and it is very hard to react to sudden changes not covered by the model. The question thus remains how to cope with the dynamic nature of the data, explain model behavior, and allow an exploration of the model's effects.

This chapter tackles the challenge and complements our analytics equipment by enhancing classifier creation, evaluation, and application by means of interactive visual interfaces. It comprises two stages that tightly integrate with each other to establish a workflow of interactive visual classifier training and real-time orchestration of these classifiers. The first stage opens the black box of model training and testing by letting analysts create linear classifiers in a visual, exploratory fashion based on recorded data from well-understood previous events. The created models then serve as task-tailored building blocks in the second stage, where they can be interactively re-combined and re-configured in a filter/flow-metaphor to create more complex ad-hoc filters during ongoing monitoring.

A specific requirement, which informed the design of both stages, was to achieve cost-effectiveness in terms of users' effort versus the amount of insight they can draw from the analysis [cf. van Wijk, 2005]. While the classifier creation phase might be resource-intensive, it is mostly conducted as an initial effort to produce a comprehensive library of classifiers. From then on, classifier creation is only done occasionally to improve and adapt classifiers or to extend the library with more specific modules according to the analysts' observations.

[1] See [Androutsopoulos et al., 2000] for a study on automatically generated keyword lists versus Bayesian classifiers.

The approach presented in this chapter has previously been published in:

- H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Krüger, M. Wörner, and T. Ertl. ScatterBlogs2: Real-time monitoring of microblog messages through user-guided filtering. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2022–2031, 2013

5.1 Background

Basic filter/flow metaphors as a means for interactive, Boolean filter combination were introduced by Shneiderman [1994]. A recent variation, called DataMeadow, was proposed by Elmqvist et al. [2008], who also employ multivariate data. Furthermore, there exist powerful visualization toolkits, such as VTK [cf. Schroeder et al., 2000], that employ pipeline metaphors to let the user compose toolchains. Although other approaches, such as KNIME [Berthold et al., 2007], have employed machine learning operators as elements in such chains, none of the existing approaches has tackled dynamically streaming data and the support for interactively re-combining filters during real-time analysis. Similar to VisGets [Dörk et al., 2008], the approach presented in this chapter integrates meta-data-based filters, i.e., spatiotemporal restrictions, directly from multiple coordinated views.

Visual approaches on classifier training have been previously published by various researchers. Seifert et al. [2010] conducted a study on user-based active learning. They allow efficient labeling of larger areas of the dataspace by presenting it as a document landscape created from unsupervised clustering. They conclude that the user-steered method has the potential to outperform machine-driven active learning and leads to more robust results on different datasets. They also highlight the need to further assess active learning with information visualization methods. Similarly, Moehrmann and Heidemann [2012] use visual means to allow labeling for classification of large image datasets based on self-organizing maps. They visualize data elements clustered by similarity and employ overview+detail techniques to navigate inside the landscape. Moreover, Höferlin et al. [2012] visualize video classification results and enable the user to directly influence the classification model.

However, an immediate visual representation of a linear classifier has not been shown by other approaches. As streaming social media data was not a focus of previous takes on visual classifier training, high-performance classification was not a major requirement as well. So far, no existing visual analytics approach

integrates post-analysis of social media documents to leverage filter creation for online application.

5.2 Visual Active Learning

In the domain of text retrieval, linear classification models are frequently used to separate two classes of documents using their representations in the term frequency vector space. For social media messages, a representation can use the same features as discussed for *TreeQueST* in Section 3.4.2. A message is then defined by a sparse vector that reflects the presence or absence of terms, hashtags, and usermentions with the corresponding *idf* values at respective index positions.

If active learning is applied, users are repeatedly presented with documents that they have to label as belonging or not belonging to one of the given classes (see Section 2.1.4). Based on the results, the algorithm incrementally adjusts the model until an optimum fit between the classes is found. There exist several strategies for deciding which data item should be queried next, e.g., informed by the impact on the model or on how much the item amounts to the overall classifier error [Settles, 2009]. In case of SVM-based classifiers, one can always choose the unlabeled data item with the smallest margin to the hyperplane, because the model will be most uncertain about it. In a figurative sense, active learning thus employs users as *oracles* that can be queried to incrementally improve the model.

Although the method can be more efficient than corpus labeling, the users still have no insight how the training progresses, how their decisions influence the model, and how their own conception of the classes is currently reflected. Especially in situation awareness, it is relevant that analysts know what they can expect from the model, and how it might perform on the data. In addition, a once well-working model sometimes suddenly fails to cover all relevant messages or delivers too many irrelevant results. In traditional AL, errors in the training process often require re-training of the model from scratch, because the analyst has no direct means to highlight these errors.

With the notion of *visual active learning*, as presented by Heimerl et al. [2012], the traditional AL process can be improved with visual interfaces to tackle these problems. To this end, the analyst is presented with a graphical representation of the classifier's hyperplane, which shows the data items separated in opposing scatterplots. Their distance to the hyperplane is mapped to the horizontal axis. Their position on the vertical axis is determined based on the first principal component of the corpus, which tends to group more similar documents more

closely to each other. Based on this central visualization, their system provides multiple tools to explore, filter, and examine the classes, and to eventually label user-selected groups of data items at once. After each labeling step, the user can re-iterate the training and immediately observe its effects in the visual representation.

A study conducted in the same work of Heimerl et al. showed that the resulting user-generated SVM-classifiers are equally or only slightly less accurate in terms of F_1-scores compared to a classifier trained with simulated active learning based on a perfect labeler [Tong and Koller, 2001]. Three findings of their evaluation showed that it would be promising to employ the method in decision-critical environments: First, the accuracy of visual AL was found to be better in the initial training phases. Interactive learning enabled participants to find meaningful labeling actions, and high-quality classifiers could thus be generated more quickly. Secondly, results of a questionnaire indicated that the participants had higher confidence in the classifiers created with the visual method, at least for coherent datasets, and that it was also less stressful and boring to work with than traditional AL. Thirdly, the profiles of the participants indicated that there was no correlation between their knowledge in machine learning and the quality of the classifiers they created. The visualization is thus suitable to communicate the abstract idea of classification to analysts with no background in the field. In conclusion, these three features make visual AL a perfect tool for the realm of social media monitoring, as the technique offers a quick, simple, and accessible way to understand and continuously adapt powerful models. The concept of visual active learning was thus adapted as part of the approach presented here.

The resulting tool (Figure 5.1) further advances the approach of Heimerl et al. with a user interface that accommodates geolocated social media messages.[2] Here, the goal of visual AL is to replace the task of finding good keywords by finding good sample messages and label them as related or unrelated to a given event or topic type. To begin the training, the analyst has to provide a seed query that could indicate good samples for the positive class by using the search box (Figure 5.1.A). The query is used to bootstrap the classifier by automatically labeling the top 50 search results of an Apache Lucene query as positive examples and 50 randomly selected low-ranked results as negative examples. In the backend of the system, a support vector machine implementation from liblinear [Fan et al., 2008] is used to create the binary classifier. However, the actual choice of binary classification scheme is interchangeable and not important to analysts. Once bootstrapping is done, analysts can explore the

[2] This UI was provided by Florian Heimerl.

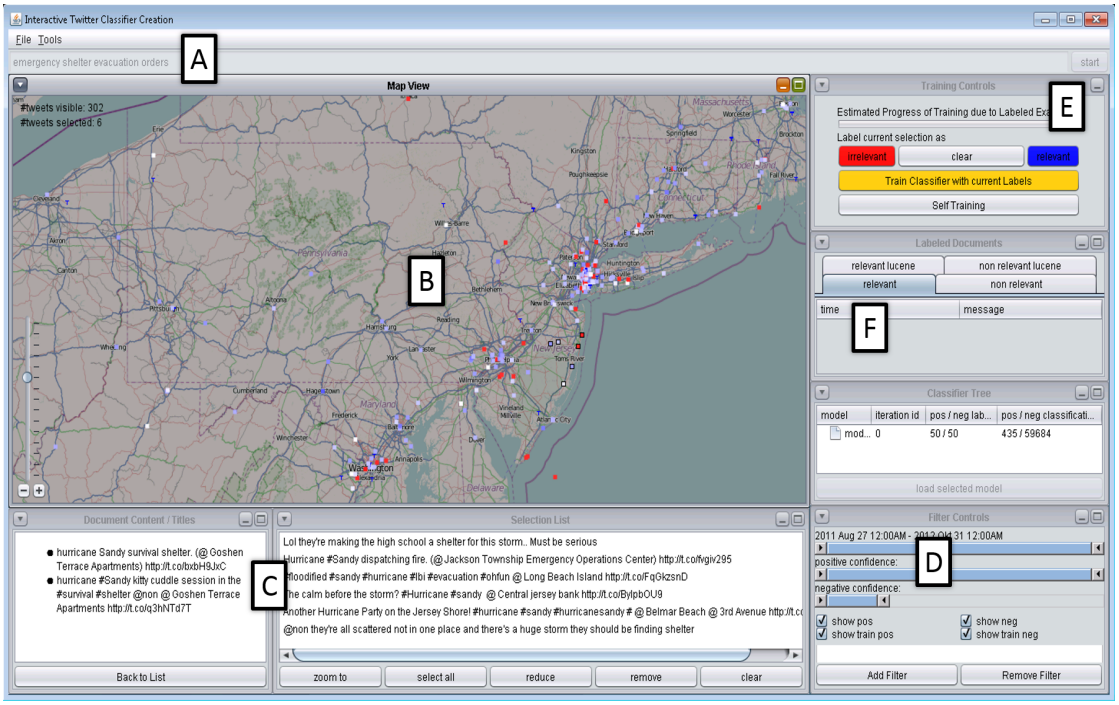


Figure 5.1 — The prototypical user interface for visual classifier training. It comprises A) bootstrap search, B) interactive map showing labeled results, C) list of selected messages, D) time-range and confidence filters, E) labeling controls, and F) labeling history. © 2013 IEEE

results on an interactive map showing the individual messages as glyphs at their respective geolocations (5.1.B). Messages considered relevant by the current model are highlighted in blue, and messages considered irrelevant are shown in red. The glyphs also indicate recently labeled messages as triangles, and messages labeled in previous iterations are shown as T-shapes. Furthermore, the confidence of classification is encoded by the brightness of glyphs - i.e. higher brightness means lower confidence.

Note that this central view differs from the one shown by Heimerl et al. [2012], which was using a graphical hyperplane representation and scatterplots of messages as exploration space for primary interactions. It was already discussed in Chapter 4 that geolocation plays a crucial role in situation awareness, and it is thus already an important indicator of the relatedness of messages in this realm. The information how the classes are comprised and separated is thus communicated through the color mapping instead of an artificial spatialization. However, it would still be possible to integrate the graphical hyperplane

representation as an additional view that the user can switch to or that can be arranged side-by-side to accommodate non-geolocated messages.

The messages on the map can be explored with an exploration lens similar to the one shown in Section 3.4.4. Moreover, users can draw selection boxes on the map to further investigate message contents. The corresponding messages are then shown in a table and can be examined in detail by clicking on the corresponding list item (5.1.C). Temporal filters can be applied to narrow or widen the range of visualized messages using an interactive slider (5.1.D). Analysts can furthermore filter for messages situated in a specific confidence range of the current model with two separate sliders for the positive and negative class respectively. This is particularly important because labeling messages with low confidence, i.e., messages near the decision boundary, has a high impact on model changes. Once analysts have found a good set of samples, they can use the controls in the upper right corner (5.1.E) to label them as irrelevant (red button) or relevant (blue button) and to re-iterate automated training based on the assigned labels (yellow button). The re-classified messages are then again shown on the map and the analysts can iteratively improve the process.

Instead of letting users non-transparently label large numbers of messages, the training environment provides feedback on the evolution of the classifier in each training iteration. This saves time compared to traditional labeling, where the performance is only evaluated afterwards by applying the result to a test set. A history of the performed training iterations is shown in an additional view (5.1.F), which allows analysts to undo labeling actions that they considered misleading.

5.3 Classifier Orchestration

Although visual AL helps to better understand and accelerate the process of classifier creation, the activity is still time-consuming and requires the full attention of an analyst. In time-critical situations, the necessary resources for adapting or extending the existing classifier set are often not available. However, if additional details about an ongoing event unravel or if sudden anomalies appear, such as trending hot topics, it might be necessary to exchange classifiers or add more specific restrictions. Analysts might be only interested in hits that contain specific keywords, are located in their region of interest, or were produced after a specific instance in time. Additionally, it can be useful at times to reconfigure existing classifiers to adapt their behavior. In order to facilitate such means, this section describes how classifiers can be interactively combined, modified, and enhanced with ad-hoc restrictions during real-time monitoring.

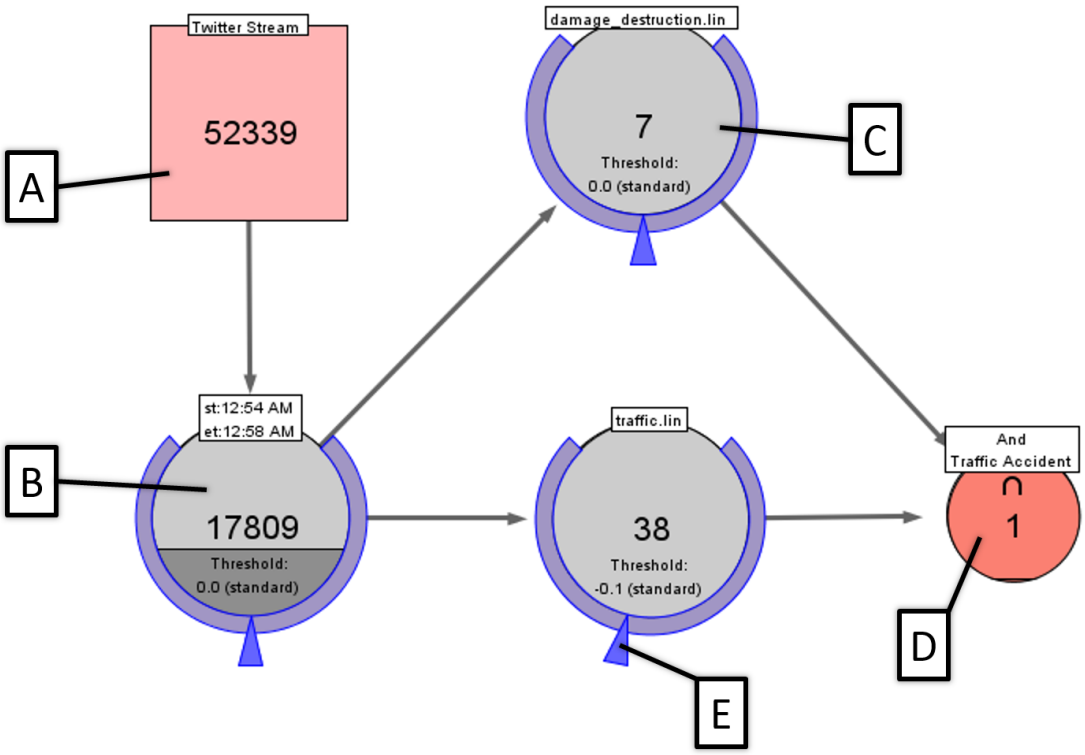


Figure 5.2 — Filter orchestration tool. Nodes represent sets of messages. They can be cascaded and connected to operator nodes through edges.

5.3.1 Interactive Filter Management

To enable classifier orchestration, a module for interactive management of regular filters has been adapted from previous works [Koch et al., 2011; Bosch et al., 2011b]. It has been enhanced with features to handle real-time data as well as to integrate linear classifiers in form of self-contained filter modules. A screenshot of this tool can be seen in Figure 5.2. The basic interaction metaphor is a node-link diagram, in which nodes represent filtered message sets. Edges between them represent combinations of these sets with each other and with set operators. Moreover, the number displayed in the node and its visual fill level illustrate the number of represented messages. While all regular nodes have a circular shape, one initial node has a rectangular shape. It stands for the complete set of currently analyzed messages (Figure 5.2.A) - e.g. messages arriving from a data stream or collected at once from an archive. Beginning with this initial node, called the *root*, the user has four different means to interact with the graph:

- **Adding/Deleting Nodes** - By right-clicking on the root node or other already existing nodes, a context-menu opens and filters can be added based on temporal, spatial, or textual restrictions. A new circular node connected by a directed edge is then created (5.2.B). This new node represents the set of messages that result from applying the filter on the old node. Users can load pre-trained classifiers as filters (5.2.C), which will work in the same fashion.

- **Combining Nodes** - All nodes can be combined with standard set operations, such as union, intersection, and symmetric difference. These operators are also established via the context-menu of an existing node. After selecting a set operation, a smaller node with the corresponding operator symbol (5.2.D) is created and connected to the parent by a directed edge. Further nodes, including the root node, classifier nodes, or other operator nodes, can be dragged to this node to also be included in the operation. The operator is applied to all nodes connected to it, and represents the resulting message set - i.e. all messages in the intersection, union, or symmetric difference of the parents.

- **Adapting Thresholds** - If a node represents a classifier, analysts can use it to interactively configure the trade-off between precision and recall. To this end, a circular slider around the node (5.2.E) is used to adjust a configuration parameter in a normalized $[-1, 1]$ range. The implementation of the classifier can then translate this value into a meaningful threshold for the classifier. For example, in case of a SVM-classifier, higher precision translates to only accepting messages with larger distance to the decision boundary.

- **Tagging Nodes** - If the messages filtered by a node are of specific relevance to the analyst, they can be associated with a user-defined name, color, and symbol. This can be used to highlight message sets related to specific events or topics, and to visually represent a mental model of the ongoing analysis.

The classifier orchestration tool is supposed to be employed as part of an integrated monitoring and analytics system. Filtered message sets can then be shown in different views, such as a map and timeline, and exploration can be enabled with overview and indication techniques like the *TagMap*. This kind of integration will be described in further detail in the *ScatterBlogs* chapter. In this case, the tagging of nodes with color and symbols should consistently highlight corresponding messages in linked views.

5.3.2 Tasks and Capabilities

Based on the intuitive filter management interface, the visual approach particularly supports three different tasks. First, it allows a quick and easy validation of classifiers in the current context. For example, analysts can easily compare the results of a classifier with a broader keyword filter by intersecting them. They can then explore whether the classifier accurately detects all relevant messages or tends to ignore important ones. In this case, they can either try to adapt the classifier's precision/recall-slider and explore the results - which is done quickly - or they can advance the classifier further in the visual AL component - which takes more time.

Secondly, the incremental cascading of filters and classifiers facilitates targeted monitoring and drill-down into larger message sets. For example, by adding a spatiotemporal filter to a classifier, one can investigate only classifier hits that arrived after some pivotal instance in time or within some relevant region. Furthermore, analysts can incrementally increase the precision of single or combined classifiers and successively investigate the narrowing result sets.

Finally, the filter management can be useful to react to unexpected changes in monitoring events. This can be done by combining several broader classifiers into more specific ones through intersection, or by broadening classifiers with ad-hoc filters through union. For example, one can intersect a classifier that detects traffic-related messages with one that detects damage-related messages to create a filter that detects *traffic incidents* (This exemplary combination was shown in Figure 5.2). To broaden an existing classifier, one can, for example, unite it with an ad-hoc keyword filter of terms that suddenly became relevant in an ongoing event.

5.4 Monitoring Workflow: Doing by Learning

Together, the visual AL component and the filter management component serve as complementing parts of an integrated *training* and *monitoring* workflow, as sketched in Figure 5.3. The training activity is performed when more time is available, and analysts can decide, based on their experience, what type of classifier they might need in the future. For example, to prepare for earthquakes, as encountered in the previous case studies, they might want to enable detection of messages related to certain sub-events such as the indicated power outages or problems with evacuation measures.

Based on a database of historic messages, they would thus first retrieve a dataset of messages that were written during related disasters of the past. Once these messages are shown in the visual AL user interface, they can bootstrap

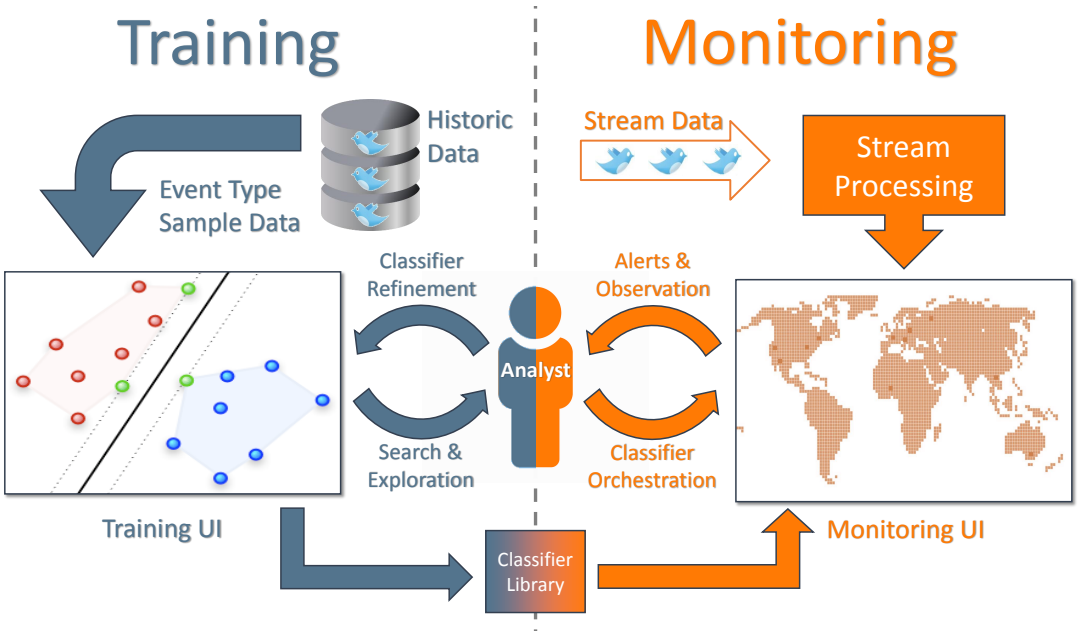


Figure 5.3 — The figure illustrates the integration of visual active learning and classifer orchestration in real-time monitoring.

a classifier based on rather broad keywords, such as `earthquake`, `outage`, or `traffic`. The visual AL component will then create an initial classifier and provide the analysts with means to search and explore classified messages to evaluate its accuracy. The analysts can select significant sample messages that were falsely categorized as relevant or irrelevant, label them correctly, and re-iterate the training process. The output of this refinement are well-trained classifiers that are stored in a classifier library for later application.

The monitoring activity is usually performed during or shortly after ongoing events, where not much time and manpower can be invested for deeper investigations. A continuous stream of messages from the social media APIs or from other message repositories would first undergo preprocessing, and they would then be visualized in an interactive monitoring environment, such as *ScatterBlogs*. This environment should at least provide search and exploration tools similar to the capabilities of the classification UI. At the beginning of a monitoring phase, i.e., a working day, the analysts might already load a standard set of classifiers from the library. They can then pre-configure and orchestrate them according to their experience. From this point on, the analysts can initially leave the tool running, and if something important happens, they can be alerted by color and symbol highlighting based on tagged nodes. If the situation further unfolds, they might have time to improve and combine

filters, validate their performance, and investigate ongoing events using search and exploration. Later, the observations made during the event can be used to inform the orchestration, adaption, or creation of classifiers to be applied on the following days or in future events.

The process shows that two roles should ideally be fulfilled by the same analyst: One of the roles exists in the slow-paced world of classifier training, where enough time is available to actually work with and understand the data to create accurate classifiers. The other role exists in the fast-paced world of real-time event monitoring, where one can just load and orchestrate the classifiers to be alerted if deeper investigation is useful or necessary. The two worlds are thus not just connected through the classifier library itself, but also through the knowledge and experience of the analysts. What they learn in the monitoring phase will give them new ideas how to adapt and expand the library in the next training phase. And the intensive work with historic data in the training phase can subsequently teach them what to expect from the data in future events.

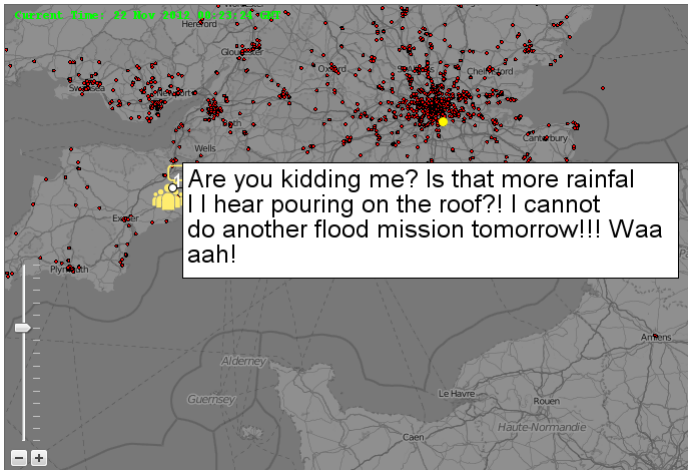
5.5 Case Study

In 2012, Great Britain and Ireland were periodically affected by a series of severe weather events such as torrential rain and heavy winds that subsequently lead to floodings and landslides. These events caused severe damage to buildings and transport infrastructure. At the end of 2012, nine lives were claimed by the events, and the total damage was estimated at 1.2 billion Euros.[3] The first series of storms occurred in June, July, and August. At that time, Sussex, Lancashire, Cumbria, and Belfast were most heavily affected by the flash floods. They caused casualties and property damage.

Six days of Twitter messages from this first timeframe were collected to conduct a case study. Based on the data, a linear SVM-classifier was created with the visual AL method as described above. A half-day training session was conducted that aimed for labeling first-hand observations related to the events. The resulting classifier enables comprehensive detection of flood-related messages. Together with other general purpose classifiers, which were generated from past events, it served as part of the classifier library in a later analysis session.

In November 2012, a second series of flash floods and heavy winds struck many parts of England and Wales. This time, the situation was even worse in terms of infrastructure damage and fatalities. In order to resemble an actual disaster management setting, classifiers created from the previous events were

[3] `http://en.wikipedia.org/wiki/2012_Great_Britain_and_Ireland_floods`



► **Figure 5.4 —** The screenshot shows the first flood-related tweet that can be detected (yellow symbol) on November 22 using the medium precision classifier. The points in the background (red) show the locations of other tweets that had been written until then.

used to conduct analysis in these later events. To enable simple exploratory analysis, the classifier orchestration was combined with means to highlight detected messages as color-coded symbols on an zoomable map as well as filter functions to restrict shown messages to specific timeframes. Based on a simulation that incrementally streamed the actual data in real-time, the analysis was conducted as if the situation currently unfolded. However, the analyst was provided with means to fast-forward the streaming process if nothing relevant happened. The following report is therefore once again the true account of an actual analysis session.

Session Report

On November 22, 00:00 UTC, the analyst began the monitoring phase by creating three different instances of the flood classifier and configuring them with different precision/recall rates. By this means, three different urgency levels are established. Checking the increasing numbers at low, medium, and high precision rates, the analyst could decide which level to investigate. This decision should be influenced by the time-criticality and severity of the ongoing situation. As tweet volumes were still manageable in the early morning hours, the analyst selected the medium precision classifier and tagged it to highlight detected messages as symbols on the map. The first detected message on this day already signalized the increasing severity of the weather situation (Figure 5.4):

- **00:22 UTC** *Are you kidding me? Is that more rainfall I hear pouring on the roof?! I cannot do another flood mission tomorrow!!! Waaaah!*

The rainfall commenced during the night. On the next morning, large volumes of reports about flooded rivers and areas started to appear. At about 14:00 UTC

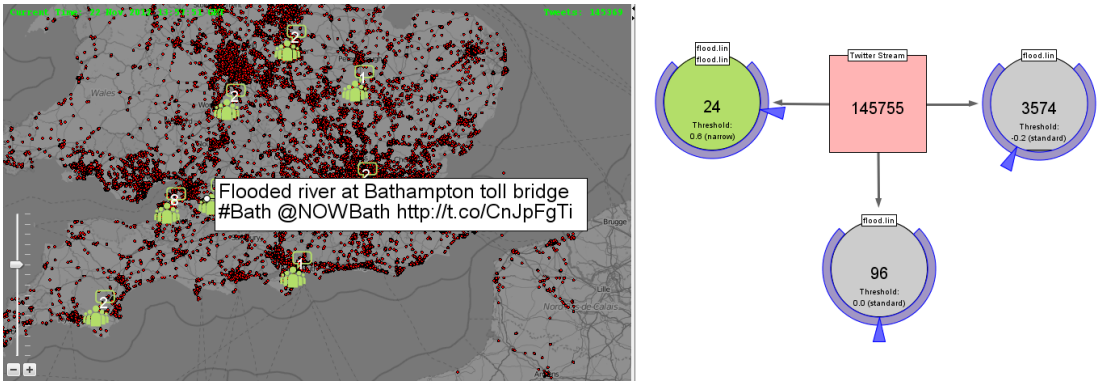


Figure 5.5 — The classifier orchestration tool on the right currently shows three instances of the flood classifier configured to high, medium and low precision. The high precision classifier is labeled with green symbols and corresponding message locations are highlighted on the map.

the medium precision classifier instance had detected more than 96 messages. However, by reading various sample tweets, it became clear that just a small portion were first-hand accounts. The other messages were just rumors and references to news media:

- **13:19 UTC:** *77 Flood warnings across England this lunch time. Hope everyone stays safe!*

However, the flood classifier was specifically trained to detect observer information, which is often indicated by keywords like 'I','we','saw','see','felt', etc. The analyst thus removed the tag from the medium precision classifier and tagged only the high precision instance to restrict the overview to this kind of messages. This revealed reports from actually affected regions (Figure 5.5):

- **07:37 UTC:** *@lisa_marie76 Most of this area is flooded in one way or another - railway line is flooded between Bristol and Swindon....*
- **08:11 UTC:** *Flooded river at Bathampton toll bridge #Bath @NOWBath* `http://t.co/CnJpFgTi`
- **13:18 UTC:** *@GBarlowOfficial hope the drive is good we are flooded down here #flood*

Since the situation seemed to become more serious every minute, the analyst now wanted to further drill down and check if there had been any reported damages to people, property, or infrastructure. In the training phase, an

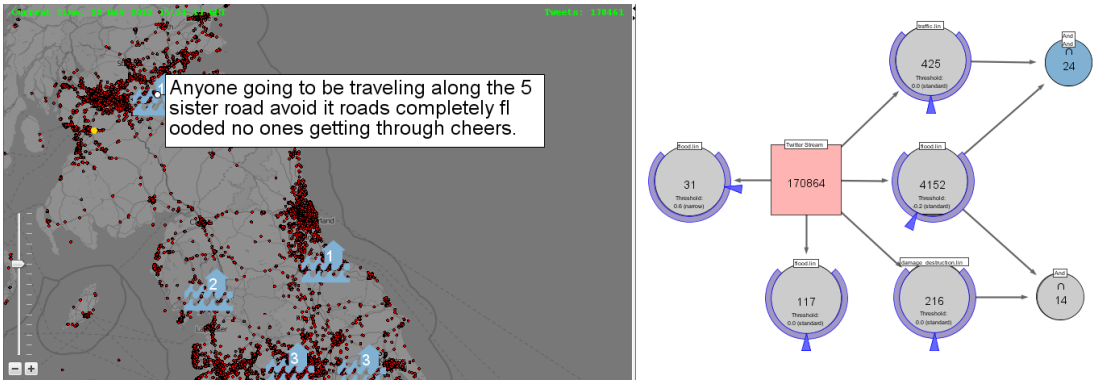


Figure 5.6 — By using intersection nodes, the flood classifier can be combined with classifiers that detect damage- and traffic-related messages. This reveals flood-damages and problems with road conditions caused by the flood. In the screenshot, the combined classifier of flood- and traffic-related messages is tagged with blue symbols.

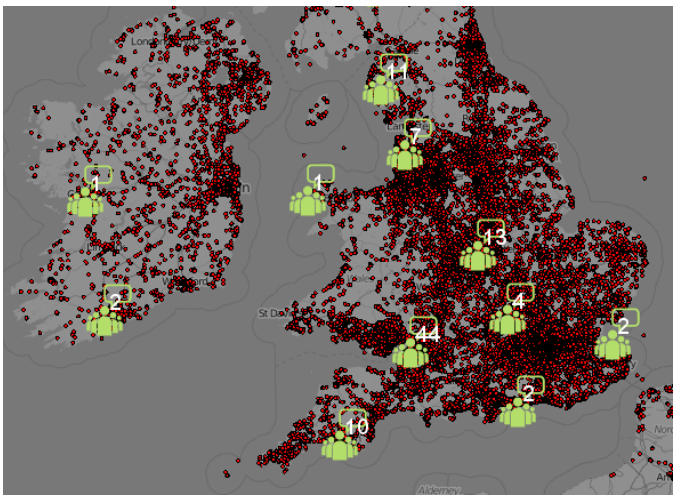
additional classifier was created to find messages that report any kind of damages or destruction. Loading an instance, however, highlighted more than 200 messages. Many of them were not directly related to the flood:

- **13:43 UTC:** *All the whole I was thinking...I wonder how much damage I could do to this guy using only a Cajon and tambourine*

At this point, classifier combination came into play. In order to achieve high recall rates, the analyst created an intersection of the low-precision flood classifier with a low-precision damage classifier. The result covered all flood-related damage reports and still significantly reduced both sets. This combined classifier then detected various flood-related damage reports:

- **07:30 UTC:** *@BBCWiltshire Now have a dead car, was caught in a flood yesterday at Bradford Rd nr Stonar School!*
- **13:54 UTC:** *Salisbury flooded car park http://t.co/4FpMGryl*

As these messages indicated possible threats in traffic situations, the analyst additionally loaded a classifier that reacts to traffic-related reports. Doing so immediately highlighted more than 400 messages. The classifier was thus also intersected with the flood classifier to reveal more specific results. The combined classifier highlighted messages related to affected traffic infrastructure and drivers (Figure 5.6):



◀ **Figure 5.7** — The final overview shows all detected eyewitness reports provided during the day. It was created by applying the high-precision flood classifier to the complete set of November 22 tweets. Overlapping symbols were aggregated and the corresponding number is shown at the midpoint.

- **14:26 UTC:** *Anyone going to be traveling along the 5sister road avoid it roads completely flooded no ones getting through cheers.*
- **18:26 UTC:** *And my street is closed off due to flooding :-(*
- **21:25 UTC:** *Lots of fallen trees, debris and surface water on the roads, big up to @WiltshireRoads for all their hard work tonight!*

Such messages can help to direct relief measures, issue warnings, and report the severity of the situation to other authorities. At the end of the day, the analyst concluded the investigation by producing an overview of how the floodings affected different UK regions. Visualizing the situation based on the medium and high precision flood classifiers showed that most flood-related messages as well as most actual eyewitness reports were provided in the south-west, particularly in the Somerset region (Figure 5.7). This corresponded to the actual severity of the floodings in this region, which was later also reported by the media.[4] Two men died at that day - one because his car was washed down a brook[5] and another one because his car overturned in the torrential rain.[6] Although these two events were not directly observed and reported by social media users, the first-hand accounts from surrounding areas clearly indicated that such incidents had to be expected.

[4] `http://www.metoffice.gov.uk/education/teens/case-studies/november-2012-flooding`

[5] `http://www.bbc.com/news/uk-20457526`

[6] `http://www.telegraph.co.uk/news/weather/9699120/Weather-three-men-feared-dead-but-downpours-will-cease-next-week.html`

CHAPTER 6

The *ScatterBlogs* Platform

To evaluate the performance and interplay of the techniques discussed in this thesis, the *ScatterBlogs* visual analytics platform has been created. An initial instance of the system was introduced at the VAST 2011 Challenge [Bosch et al., 2011b], as has been demonstrated in Chapter 4. This early prototype has since been advanced into a multi-purpose framework for social media monitoring and analytics, which provides scalable data management, stream-enabled real-time visualization, basic filter capabilities, and plug-in integration of highly interactive tools. In its three-year development process, it has served as a platform not just to develop own approaches, but also to adapt, implement, and evaluate recent ideas from ongoing research in the domain. Each of the components presented in Chapters 3, 4, and 5 have been implemented as plug-ins that enhance the system's basic functionality and make it more applicable in situation awareness domains. Together these approaches establish a set of tools at the analyst's disposal, complementing each other to provide a comprehensive picture.

In many cases, research prototypes in visual analytics serve as a proof-of-concept demonstrator, showing that a certain class of data and/or tasks can be handled by a given visualization approach. They are thus by no means ready to be deployed in real-world application settings. With *ScatterBlogs*, however, a significant effort has been invested to advance the system's reliability, simplicity, and ergonomics to a near production-ready quality. This ensures that the interaction and visualization methods can be evaluated by domain experts without obscuring their performance by usability issues.

ScatterBlogs is completely developed in Java. It employs Apache Lucene[1] as search engine, Prefuse[2] for graph visualizations and color mapping, Swingx-ws[3] and OpenStreetMap[4] for geographic maps, Geonames[5] for geocoding toponyms, SentiStrength[6] for sentiment analysis, MALLET[7] for statistical topic extraction, Google Translate for automated translation [8], and RabbitMQ[9] for client/server-communication. Furthermore, it uses a range of clients to collect data from the social media APIs, including Twitter4j[10], the YouTube Client Library for Java[11], and Flickr4Java[12].

The following subsections give an overview of the basic features and tools and highlight similarities to related systems and approaches in the domain. We will then have a look at the particular instances of the *TreeQueST* query creation, the *TagMap*, and the classifier orchestration techniques and how they have been implemented as part of the system.

Previous *ScatterBlogs* versions were introduced by:

- H. Bosch, D. Thom, M. Wörner, S. Koch, E. Puttmann, D. Jackle, and T. Ertl. ScatterBlogs: Geo-spatial document analysis. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 309–310. IEEE Computer Society, 2011
- D. Thom, H. Bosch, S. Koch, M. Wörner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 41–48. IEEE Computer Society, 2012
- H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Krüger, M. Wörner, and T. Ertl. ScatterBlogs2: Real-time monitoring of mi-

[1] `http://lucene.apache.org/core/`
[2] `http://prefuse.org/`
[3] `https://java.net/projects/swingx-ws`
[4] `http://www.openstreetmap.org/`
[5] `http://www.geonames.org/`
[6] `http://sentistrength.wlv.ac.uk/`, see also [Thelwall et al., 2010]
[7] `http://mallet.cs.umass.edu`
[8] `https://developers.google.com/api-client-library/java/apis/translate/v2`
[9] `https://www.rabbitmq.com/`
[10] `http://twitter4j.org/`
[11] `https://developers.google.com/youtube/`
[12] `https://github.com/callmeal/Flickr4Java`

croblog messages through user-guided filtering. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2022–2031, 2013

Parts of this chapter have previously been published in:

- D. Thom, H. Bosch, and T. Ertl. Inverse document density: A smooth measure for location-dependent term irregularities. In *International Conference on Computational Linguistics COLING*, pages 2603–2618. Indian Institute of Technology Bombay, 2012
- D. Thom, R. Krüger, T. Ertl, U. Bechstedt, A. Platz, J. Zisgen, and B. Volland. Can Twitter really save your life? A broad-scale expert study of visual social media analytics for situation awareness. In *IEEE Pacific Visualization Symposium (PacificVis)*. IEEE Computer Society, 2015, to appear

6.1 Data Structures and Index

Coping with large volumes of real-time data is a key challenge for many systems that center on web content analytics. Recent approaches, such as Nanocubes from Lins et al. [2013], have demonstrated how space- and time-referenced data can be structured in a fashion that allows fast queries and visualization of millions of items in linked geospatial and temporal views. Also, big data aggregation strategies from online analytical processing (OLAP) have specifically been adapted to support interactive visualizations. In that vein, imMens from Liu et al. [2013] advances OLAP-datacubes [Harinarayan et al., 1996] to the notion of multivariate data tiles. They build on the observation that a maximum of four dimensions is needed to support brushing and linking in one- and two-dimensional binned plots - e.g., clicking a bar in a histogram and showing affected cells on a map grid. Standard datacubes, suffering from the combinatorial explosion of data dimensions, can thus be decomposed into multiple smaller cubes that only link combinations of up to three or four dimensions over limited ranges of the coordinate space. If a respective area of a plot, e.g., a map area, has to be visualized based on selected data, values can be quickly accumulated and rendered from precomputed tiles.

However, a problem with such pre-aggregation strategies arises if we want to handle textual queries, where the challenge results from the inherent high dimensionality of the data. ImMens is suitable for brushing that happens in one plot and highlighting corresponding bins linked in the other plots. It thus only has to consider mappings of the dimensions of the first plot to the maximum

of two dimensions visualized in each other plot. If brushing should also be allowed by entering (multiple) keywords and linking messages that contain it, we would basically need individual data tiles for any possible combination of words in the given language and plotted dimensions. This would again lead to a combinatorial explosion. Furthermore, in addition to that problem, real-time changes and selection of arbitrary bin intervals are also not handled well by such pre-aggregation strategies, as they usually assume fixed bin interval sizes and bounded value ranges to define the aggregation structures. While the coordinate space of a map grid is usually limited in width and height, the temporal axis in situation monitoring is not bounded by a definite start- and end-date.

To tackle the challenges specific to the domain of social media monitoring, *ScatterBlogs* thus mostly refrains from pre-aggregation. Instead, it relies on powerful index structures that allow quick accumulation of data in arbitrary granularities and ranges in time, space, and content. The custom-built data management is based on a tight integration of Apache Lucene and time-sequenced spatial quadtrees that are both associated with fixed time intervals (e.g. one day). The messages are stored as contiguous records in Lucene tables, which, among others, include message id, username, timestamp, content, geolocation, usermentions, urls, and hashtags. Keyword queries and retrieval of items by id can thus quickly be performed based on this storage type. In addition, the quadtrees only store message id, timestamp, and geolocation to support fast temporal and geographic range queries.

Given a temporal range together with a geographic bounding box, the data management first collects all fixed time intervals that intersect the temporal range. Each of the associated quadtrees is then recursively queried with the bounding box to collect geolocations and/or message ids of covered items. If more message details are requested or if additional textual filters are applied, the collected message ids are used to retrieve the records based on Lucene queries.

6.2 System Architecture

ScatterBlogs is designed as a client/server-system, and the data and index structures are similarly used in both components. An overview of the system architecture can be seen in Figure 6.1. While the client contains the graphical user interface, the server provides access to larger archives of previously collected messages as well as auxiliary data, such as precomputed *TagMap* and *idd* values. It delivers the data to the client on request.

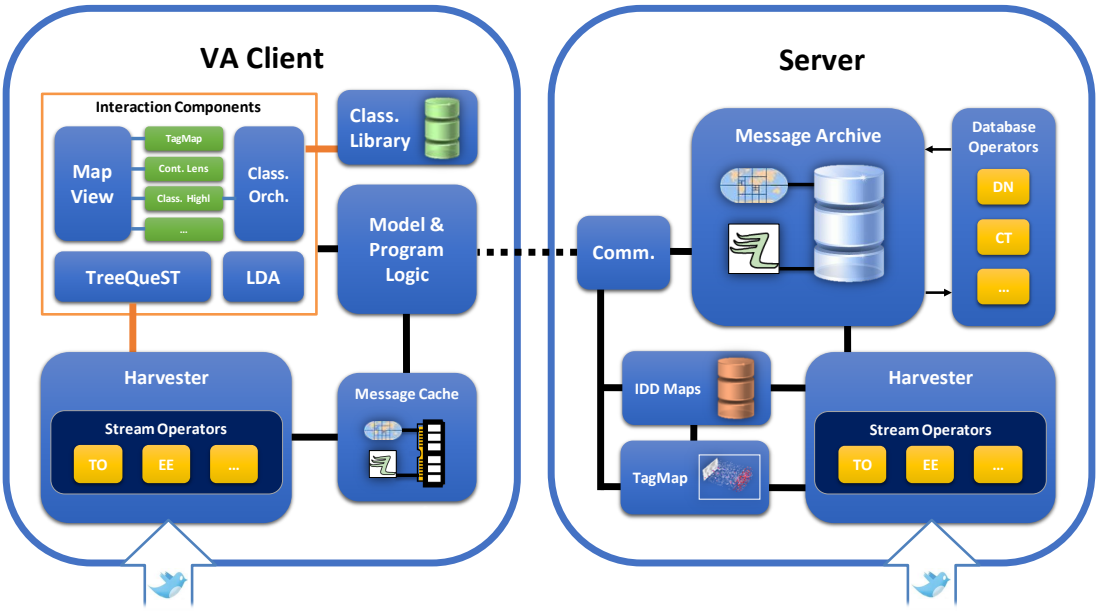


Figure 6.1 — The image illustrates the *ScatterBlogs* system architecture. Components that directly interact with each other are connected by thick lines. Data transmission between client and server is highlighted by the dashed line. (Database icons made by Freepik and Softicons, licensed under CC BY 3.0)

The system provides two means of collecting data from the social media services. Using an integrated harvester module, the server collects data from streaming APIs, which happens in a continuous fashion, and from search APIs, which is done in fixed request intervals. That server-side data collection is supposed to be based on broad filter conditions, such as multiple large geographic bounding boxes or lists of common and frequently used keywords. Although the server still misses a majority of the data, this strategy ensures that at least a large sample of historic messages can be provided for post-analysis of events and for classifier training.

For the second way to collect data, a similar version of the harvester module is included in the client. It allows immediate integration of live data, which is initially stored in a volatile message cache. On user request, data that was specifically collected by the client can be added to the server storage. Both instances of the harvester module can be configured to perform various NLP-preprocessing steps. These sequential processes are called *stream operators*:

1. **Tokenization** - Decompose message text into a set of words

2. **Entity Extraction** - Extract URLs, hashtags, and usermentions
3. **Stopword Removal** - Flag common words like `the`, `and`, `she`
4. **Stemming** - Find the root form of words
5. **Polarity Analysis** - Assign a sentiment score to the content
6. **Index Insertion** - Update the quadtrees and Lucene index

The extracted information is added to the plain records as annotations and meta-data. The server furthermore houses precomputed *idd* maps (Section 4.4) and continuously computed cluster centroids of the *TagMap* (Section 4.3), which are also updated by the message monitor. This will be further explained in Section 6.4.

In addition to the stream operators, which have to guarantee live processing of the data at all times, the server also features *database operators*, which are allowed to process the data at a slower pace to employ more powerful algorithms. Available database operators include language detection and translation, extraction of location names, and automated location discovery [Thom et al., 2014a]. Although the processing speed of these operators may fall short of message throughput during peak times (e.g. daytime in the US), they can usually catch up during troughs. Also, they can be configured to skip messages if the gap becomes larger than one day of data.

Once processed and stored, the social media data, as well as the *idd* and *TagMap* data, can be requested by the client via a RabbitMQ-based communications module. To this end, the client can use temporal, geographic, and keyword-based filter conditions, as well as other request parameters, to define and limit the data volumes. The model and program logic component of the client receives the data, stores it on the local message cache, and distributes it to the modules of the user interface. Depending on the view or visualization overlay, this is either done via polling by the module or via event-based push notifications.

The harvester module of the client is not controlled by the central program logic but directly by the *TreeQueST* component. *TreeQueST* uses the harvester to collect and visualize samples of the data from search APIs and to define the current real-time streams of the client. New data received by the harvester is immediately pushed into the memory cache. The cache then informs the central program logic about updates, and these notifications are also forwarded to selected interactive components, such as the *TagMap*. Moreover, queries that have been defined in the client with *TreeQueST* can also be submitted via the interface to enable continuous collection by the server.

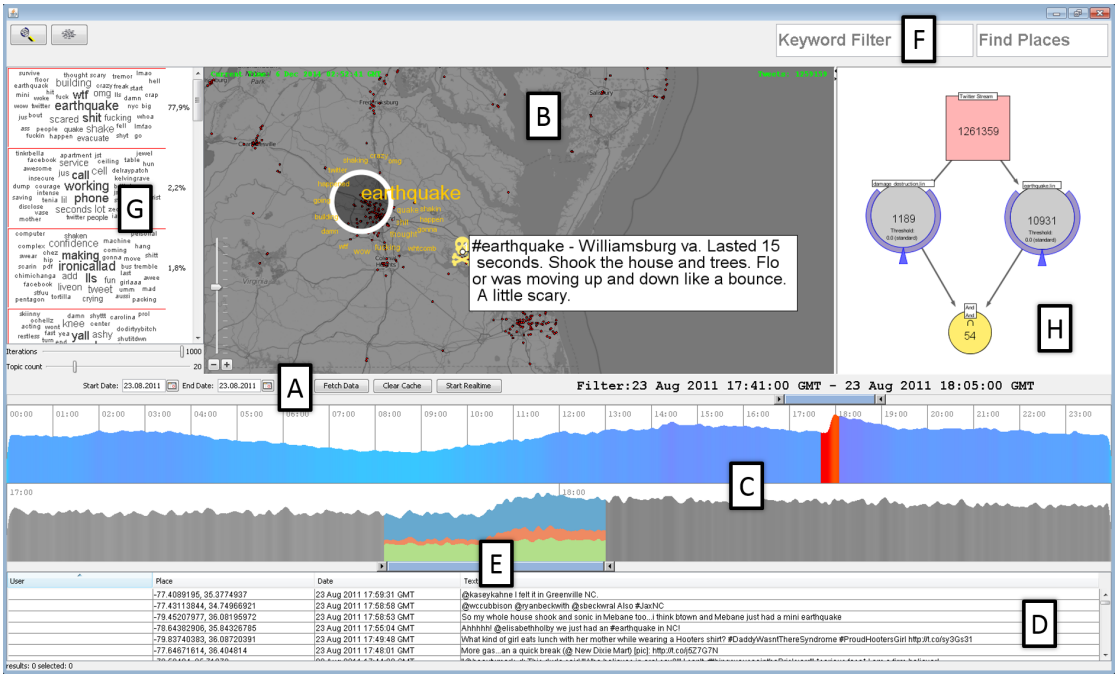


Figure 6.2 — The primary *ScatterBlogs* UI with activated *ContentLens* and classifier orchestration. It consists of A) calendar control widgets, B) an interactive map showing message locations as red dots, C) hierarchical temporal overviews, D) a table of messages details, E) timeline controls, F) textual and geographic search, G) LDA topic view, and H) classifier management.

6.3 User Interface

The *ScatterBlogs* graphical client interface can be seen in Figure 6.2. The index and data structures of the client are suitable to visualize data of up to 5 million messages at the same time. Data beyond that scale has to be stored in the server's archives and can be re-loaded into the main UI on user's request. Loading data from the archives is done via calendar widget-controls, which are also visible in Figure 6.2 (A). In addition, the user can draw geographic polygons and provide keywords to limit the request. The UI will load filtered historic data until the fixed limit of 5 million messages is reached and it will inform the user if the requested data exceeds that limit. Cached messages that provide geolocation are initially all shown as points on an interactive world map (6.2.B). The messages are additionally highlighted by color and animation if they were recently received. The map can be moved with the mouse and allows fluent zoom-level changes from a global overview down to the city and street level. The visualization of messages is based on threaded tile-rendering that also utilizes multicore hardware if present. For each visible tile, the corresponding

geographic bounding box is queried from the spatial index to find covered message locations. However, if no additional filters are applied, this process can be accelerated by storing the message volumes of subtrees at the inner nodes of the quadtree. If, for example, all messages of a subtree would fall into the same pixel of the current viewport, there is no need to continue the recursion, and the number of messages in the subtree can be represented by color coding the pixel. *ScatterBlogs* thus allows to switch between showing all messages as individual dots or an aggregated view based on color-coding the pixels of the current viewport.

All messages, including the ones without geolocation, are also shown in a temporal overview of message volumes (6.2.C). The upper pane is showing the complete overview of data in the message cache. The lower pane is only showing data covered by the current user-defined filter conditions. The plot is rendered based on one-dimensional kernel density estimation to achieve a smooth curve without predefined bin intervals. It furthermore uses the sentiment scores assigned by the harvester to illustrate the polarity of messages. Positive message volumes are once again shown in green, negative message volumes are shown in red, and neutral ones are shown in blue.

Every visual entity can be interacted with by brushing selections on the map and timeline, which will present the user with more details such as the textual content, authors, and timestamps of messages in a table (6.2.D). The user can apply local filters to the data by means of geospatial bounding boxes, the sliders above and below the temporal overviews (6.2.E), and textual keyword lists (6.2.F). All views are linked and will be immediately updated when the filter conditions change.

6.3.1 Basic Exploration Tools

Despite the availability of basic selection and filter means, analysts still have to read many messages by themselves, e.g., if they were identified as potentially relevant, or if they may add information through exploration. In addition to the map, timeline, and message list, *ScatterBlogs* therefore provides two simple means to investigate selected groups of messages in a summarized fashion. Lohmann et al. [2009] have demonstrated that circular tag clouds with decreasing popularity are a suitable means for finding the most popular topics in a document set. Both aggregation tools thus use this technique as a basis for quickly conveying the polarity, content, and coherence of selected message sets to the user.

Figure 6.3 — The *ContentLens* consists of circular shapes that can be moved over the map to examine message contents. In the snapshots, the lens is used to investigate messages during the San Diego Comic-Con (cf. Section 4.4.3). The figure illustrates, from left to right: default application, activated stopword removal, and *idd*-based weighting. In the *idd* version one can observe recently popular topics, such as autograph signings by popular authors and talk about co-located events.

ContentLens

Previous chapters highlighted that exploration lenses have been a useful focus+context technique in visual analytics (Chapter 2). To quickly assess the content and overall tone of messages in a geographic area, users of *ScatterBlogs* can thus additionally activate geographic lenses similar to the one presented as part of *TreeQueST* in Section 3.4. They can freely be moved over the map and timeline, and dynamically generate tag clouds of terms that were frequently used by messages in the respective ranges.

By default, the scale and order of tags is defined according to their frequency, and stopword removal can be applied to eliminate common words. Alternatively, the system allows to employ the *idd* measure (Section 4.4) to determine the weights based on the specific relevance of the terms for given geographic areas. (See Figure 6.3 for the different versions.)

To monitor current developments at selected geographic locations, the user can also place multiple lenses on the map. They will continuously adapt the tag clouds to new messages from real-time streams, and the user can apply any of the client's filters to restrict the evaluation to specific subsets.

Topic Models

LDA topic modeling is a popular statistical inference technique in natural language processing [Blei et al., 2003]. It is commonly used in data visualization to aggregate and organize large document collections. Put simply, LDA builds generative models from the given corpus in an unsupervised process and

delivers a list of prevalent topics together with their probability of occurrence and a bag of words characterizing their content.

Several approaches from data mining and NLP have developed variations of the basic model that specifically adapt to Twitter and other sources of social media data [Quercia et al., 2012; Ramage et al., 2010; Zhao et al., 2011]. Visual analytics approaches have then employed these models to aggregate, organize, and label groups of related messages and to detect relevant outliers [Dou et al., 2012, 2013; Pozdnoukhov and Kaiser, 2011].

To find coherent topics in relevant message sets, *ScatterBlogs* also features a simple visualization based on LDA topic models [Chae et al., 2012]. The topics are extracted from selected message sets using the MALLET toolkit. In this case, Gibbs-sampling is employed to estimate topic posteriors [Casella and George, 1992]. The extracted topics are shown to users as a list of concise tag clouds ranked by their probability to appear in messages from the selection (Figure 6.2.G). Users can adapt the granularity and computational effort of the topic extraction using the interactive sliders below the list. In most cases, the tag clouds will seem rather random and do not show any meaningful combination of words. However, if a coherent discussion is included in the messages, it can quickly be identified by contextually related tags in the clouds.

6.4 Implementation of the Analytics Cycle

While the basic user interface sets the stage, the three main techniques discussed in this work compose a workflow that enables users to achieve ongoing situation awareness and to visually model a situation overview. The structure of this scheme implements the analytics cycle initially outlined in Section 2.4.1.

6.4.1 Step 1: Query Optimization and Retrieval

ScatterBlogs currently harvests data from the Streaming and Search API of Twitter as well as the Search APIs of Youtube and Flickr. However, in the latter cases, the system only uses the textual descriptions provided with the posts for search and analysis. Images and videos can only be shown when entry details are requested by the user. To find optimized request parameters, *TreeQueST* can be opened by the user as an internal view (Figure 6.4) or within an external frame on a second display. It can be used to establish a data stream from the APIs and start pulling entries in real-time, or to retrieve sets of keyword-related messages and import them into the primary view at once. If data has been retrieved with *TreeQueST*, the corresponding entries are immediately highlighted on the worldmap and in the timeline of the main view. The user

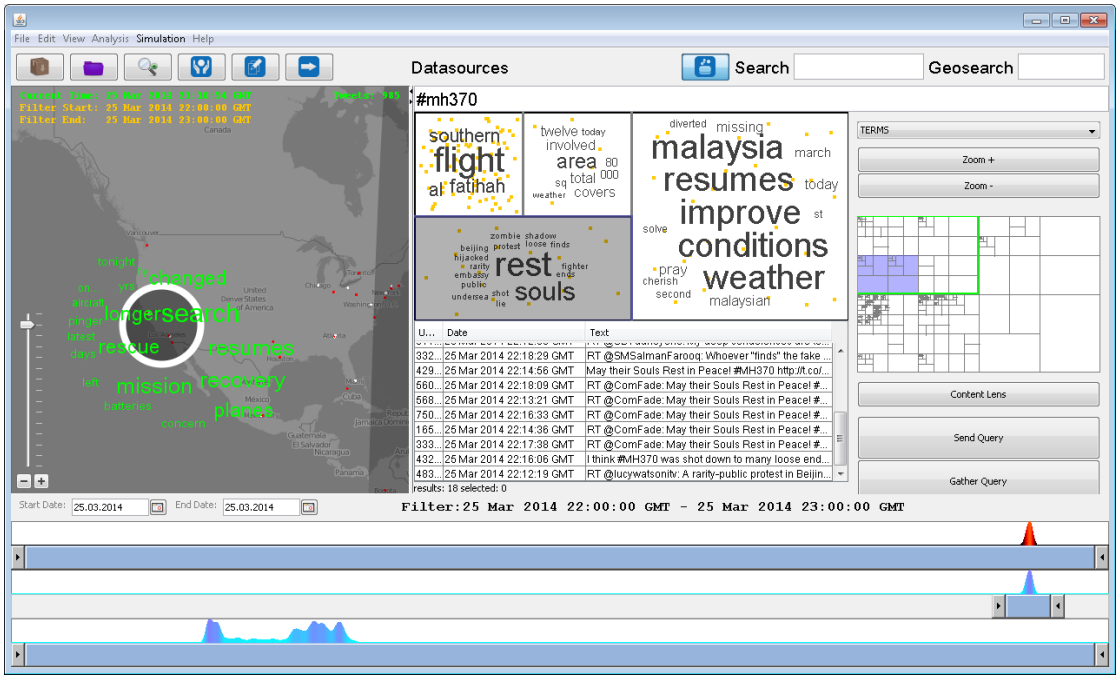


Figure 6.4 — Opening *TreeQueST* as integrated *ScatterBlogs*-view. The screenshot illustrates how the tool was used to collect news about Malaysian Airlines Flight 370, which suddenly disappeared on March 8, 2014. © 2015 IEEE

can thus also quickly understand the geographic and temporal coverage of a sample request. Used in a two-display setup, *TreeQueST* additionally serves as a data exploration and aggregation tool that not just calls data from APIs, but also allows to hierarchically examine message sets selected from the primary view. This mode of operation has been shown and evaluated as part of the VAST 2014 Challenge [Thom et al., 2014b].

6.4.2 Step 2: Overview and Indication

The most challenging part for the *TagMap* integration is the relevance normalization done by the *idd* model, because this component requires large volumes of precomputed term density maps. These maps are thus completely stored on the server, which can dynamically compute and transmit *idd* values for given sets of terms by request. Once received, the client caches values for resumed utilization.

Both components, client and server, integrate an individual *TagMap*-module, as can be seen in Figure 6.1. The server continuously updates a global *TagMap* of all messages that were collected with the broad request parameters. On client's request, a temporal section of this *TagMap* can be delivered either together with

a corresponding set of messages or as a preview of larger data ranges. The second mode is especially useful if the corresponding set of messages would be too large to be delivered by the network or to be stored in the client's memory. In this case, the *TagMap* is transmitted as a set of spatiotemporal centroids together with their corresponding term, significance score, message volumes, and temporal extent. The data volumes are thus significantly smaller than that of the actual message volumes. They can even be further capped by transmitting only a fixed number of top-weighted centroids. The server-side implementation of the *TagMap* only contains the anomaly detection part without the algorithm for adaptive visualization.

The second *TagMap*-module, which is integrated in the client, contains both algorithms. It is used to enable visualization of anomalies as well as ad-hoc anomaly detection. It can either just display the data retrieved from the server, or quickly compute a smaller *TagMap* based on the harvester's inbound message stream, intermediate samples collected with *TreeQueST*, or filtered subsets from the classifier orchestration.

6.4.3 Step 3: Task-adaptive Filters

The interactive filter management, as described in Section 5.3, is directly integrated as one of *ScatterBlogs* client views (6.2.H). Here, the root node represents all messages that were received by real-time streams, ad-hoc requests defined by *TreeQueST*, or data retrieved from the archive. The classifiers are evaluated in real-time, and the numbers on all nodes update with each new message in a streaming pipes-and-filters fashion. When using the system in daily operation, it is assumed that a base library of classifiers has already been created with the visual AL tool. This library is delivered together with the client, and each classifier is stored in an individual model file. For example, if the SVM-model is used, these classifiers are represented as sparse term-weight vectors. However, other formats are also available and will automatically be interpreted by the module.

In addition to the classifiers, the user can create nodes from any filter configuration created in the primary views. The view can thus be used to combine classifiers with ad-hoc temporal, spatial, textual, or other filters. And it can also be used to combine complex filter constructs with each other. The user can furthermore select nodes and set them as the new primary filter of messages that should be shown in the other views.

As described before, the user can tag nodes to identify them as representing a relevant stream or subset of messages. In this case, a menu pops up to define the symbol, color, and name of the tag. From that point on, all messages covered

by the filter or classifier are highlighted in the other views, e.g., as symbols on the map and with the corresponding color in the table. In zoomed-out views, overlapping individual symbols on the map are merged and represented by an aggregated symbol at the mean location. This symbol also highlights the total number of detected messages in the area, making it easy to assess which region has most classifier hits, as was shown in the case studies.

6.4.4 Closing the Loop

Together with the *TagMap*, the classifier orchestration can be used in a tighter loop, where initially all messages are aggregated. Based on this first overview, the analyst decides what could be relevant, uses the classifiers for drill-down, and again aggregates the result with the *TagMap*. That process is repeated until all relevant messages or patterns have been found and/or a better situation understanding has been established. If this loop reaches the point where no new information is added, or where hypotheses suggest an additional information need, the analyst can turn back to the larger foraging loop with *TreeQueST*.

CHAPTER 7

Evaluation

Various studies and experiments have been conducted to evaluate the methods presented in this thesis. The *ScatterBlogs* system has been used as a platform to evaluate the integrated analytics scheme with disaster response and critical infrastructure experts as well as to collect their opinions on visual social media analytics in general. This overarching study thus also served as a means to see what the research can accomplish today, and what challenges it needs to address in the near future. The results of this larger study will be presented in this chapter.

Some components of *ScatterBlogs* have been evaluated independently from that larger study, insofar as it seemed advisable or useful. Because of the limited technical and temporal resources available at the domain expert sites, *TreeQueST* was not tested as a *ScatterBlogs* component in the larger study. In this case the necessary training and real-time application for the tests would have exceeded given constraints. It was therefore evaluated in a separate user study. These results will be presented in the second part.

Parts and results of this chapter have been previously published in:

- D. Thom and T. Ertl. TreeQueST: A treemap-based query sandbox for microdocument retrieval. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1714–1723. IEEE Computer Society, 2015

- D. Thom, R. Krüger, T. Ertl, U. Bechstedt, A. Platz, J. Zisgen, and B. Volland. Can Twitter really save your life? A broad-scale expert study of visual social media analytics for situation awareness. In *IEEE Pacific Visualization Symposium (PacificVis)*. IEEE Computer Society, 2015, to appear

7.1 Evaluating Visual Analytics

Visual analytics constitutes a quite novel form of scientific conception. Approaches usually not just discuss an individual technique, method, or research question. Instead, they frequently present a conglomerate of components, capabilities and methodologies - sometimes informed by various other scientific areas - to facilitate problem solving in one or multiple application domains. Researchers in this area thus also require novel means to comprehensively evaluate their approaches. Before the study results are presented, this section therefore discusses various views on the general challenge of visual analytics evaluations.

7.1.1 Evaluation Design and Benchmarks

Catherine Plaisant [2004] highlights that a major goal of information visualization is to make unexpected discoveries. Tasks in controlled user studies, however, have to be simple and should allow measurable success criteria. They are thus sometimes contradicting independent discovery. Plaisant therefore recommends to let users freely explore the data on their own before and after the controlled tasks and to let them report their observations. In this study she also highlights the path to successful evaluations as a three step process of building sophisticated data repositories, investigating case studies and recording success stories, and creating generic toolkits and frameworks. Further comments on insight-based evaluations have been provided by Chris North [2006]. He compares two types of procedures. While in one type, complex analytics tasks are assigned to the users and answers have to be given in a measurable form, in the other type, benchmark tasks are eliminated completely and it is observed how the users use the system on their own. He concludes that both types of measures are needed, as complex benchmark tasks help to identify low-level effects, while eliminating them provides a richer view of the system's insight capabilities. The evaluations presented in this chapter try to cover both areas by presenting the analysts with tasks that have to be solved to proceed, but also by designing these tasks open-ended and with a focus on free exploration.

More recently, Smuc et al. [2009] suggested a three level methodology that combines summative and formative evaluations. In three phases of participatory tool design, they propose to compare different tools, variations, and user groups, test the fit of designer's intentions, and to align the tools with the insight generation process. In terms of testing integrated visual analytics systems, Laskowski and Plaisant [2005] describe three levels of evaluation: the component level, in which individual techniques and algorithms are evaluated, the system level, in which the interplay of components and the utility and accessibility of the platform and interface are tested, and the work environment level, in which the chance of adoption by domain users is assessed based on field studies. The evaluations presented in this chapter try to cover the three levels by designing tasks that encourage to start with the use of specific techniques, work with multiple tools at more complex stages, and that are furthermore embedded with familiar information gathering processes of the analysts.

However, they do not follow a fully standardized procedure of visual analytics evaluations as such a scheme or template does not exist so far. Initial efforts in this direction have been undertaken by Plaisant et al. [2008]. They promote the creation and adoption of a visual analytics benchmark repository, which can then be used to develop, evaluate, and compare solutions as well as to enable long-term experience. To this end, they recommend to further establish and popularize scientific contests, such as the InfoVis contest or the VAST challenge, and to make the created data and solutions freely accessible. By this means, researchers can test their methods on real-world inspired datasets, clearly defined tasks, and with available ground-truth. Moreover, they can compare their method's performance with all solutions that were initially submitted to the contest.

Although such benchmark repositories provide useful means to evaluate a method's performance and usability, one can only make limited assumptions about its real-world applicability. For example, the VAST challenge 2011 data helped to demonstrate the possibilities and detection capabilities of the *TagMap* (see Chapter 4.1). This dataset, however, only comprised 0.2% the size of the current daily Twitter volumes, and real-world scalability was therefore not a true issue. Moreover, observations made in the course of this thesis demonstrated that actual communication and publishing behavior in social media significantly differs from the rather simple model that was employed for the synthetic data. Although the basic appearances may share some similarities, the underlying detection algorithm and visualization method of the *TagMap* thus severely differ from the ones that were used in the challenge. One can conclude that while the challenge and benchmark repository can serve as a valuable motor to inform novel visual analytics designs and to allow iterative

testing during development, the actual challenge of formally benchmarking visual analytics under real-world conditions must still be considered the subject of an ongoing discussion.

7.1.2 Practical Considerations

It is an underestimated challenge of visual analytics evaluations that there are frequently not enough time and personnel resources available to thoroughly familiarize users with the system, the tasks, and the data [see also Plaisant, 2004]. This is particularly the case if the evaluation is conducted with professionals, whose expertise is at the same time needed for other tasks. However, in contrast to easy-to-use interfaces in the consumer domain, such as web clients or mobile apps, sophisticated systems for professional use often require considerable amounts of training before they can be of actual benefit. Well-known examples of such systems include computer aided design (CAD) or integrated development environments (IDE).

It has also been a major problem of past evaluations that study participants were handicapped by the low usability of research prototypes. Such prototypical systems are frequently created by small teams in short development cycles. Since they primarily serve as proof-of-concept platforms, they are prone to limited simplicity, interaction fluency, and ease of use. In visual analytics evaluations, the underlying concepts are then often obscured by the difficulties of handling the system. First-time users might not even reach the point where they could make informed comments. Instead, they are frequently locked out by the difficulties of coping with the controls. Poor study results may then suggest that a visual analytics technique fails to meet its goals, while the results are actually just a consequence of insufficient software engineering.

In the following evaluations, both challenges have been carefully addressed early on in the study design. In its three-year development process, *ScatterBlogs* has been iteratively enhanced to a near production-ready software quality. Various case and user studies conducted underway helped to inform the structure and capabilities of the interface. Moreover, the system's performance as well as the possibility to implement powerful and fluent interactions are facilitated by a sophisticated backend. In general, the challenge of short training phases in the evaluations is hard to come by, but it has been addressed by designing domain-oriented tasks that are suitable for beginners, and by centering them around real-world data of popular and/or well-understood events. Additionally, in case of the expert evaluation, significant efforts have been invested to thoroughly design a tutorial-style introduction that quickly conveyed the basics of the system.

7.2 *ScatterBlogs* Domain Expert Study

A visual analytics system is a combination of three elements: algorithmic methods, visual interfaces, and the user that applies it. Its performance can thus only be assessed if the plain software is complemented by an analyst that contributes his experience and intuition to correctly interpret insights and to steer the analytical process. To investigate the actual usefulness of the social media analytics scheme of this thesis, the following study was therefore conducted with domain experts. It investigated its real-world applicability and practical relevance using *ScatterBlogs* as reference implementation.

The results provide multiple different perspectives on the challenge, which have been acquired by contacting various large companies and institutions with sophisticated experience in command, control, and interoperability environments. All of them see situation awareness as a pivotal element in achieving their goals. In total, twenty-nine domain experts participated in the study. They came from eight different institutions, half of which are commercial and deal with critical infrastructure management, the other half are government authorities in disaster response from different regions of Germany.

Based on Twitter data collected during a recent disaster, the 2013 German Flood[1], a task-oriented study was designed, in which the experts were asked to apply the integrated system for gaining situation awareness in a real-word crisis. These tasks thus served a means, both to give the experts a feeling how they could benefit from social media in their efforts, and to help understand how they would perform with *ScatterBlogs*. The system was used as a device to drive the discussion and to collect their comments about the individual tools and techniques, the analytics scheme as a whole, and visual social media analytics in general.

The study was conducted as a collaborative effort of user interface experts from Siemens AG, disaster management experts from the Academy for Crisis Management, Emergency Planning, and Civil Protection (AKNZ), and researchers from the University of Stuttgart, including the author of this thesis.

7.2.1 Experimental Setup

Based on the collected Twitter messages, the experts were enabled to evaluate the techniques and tools under real-world conditions and could get a feeling how they would perform on actual data. The details of the employed dataset as well as four tasks that were created for the study will be described in the

[1] `en.wikipedia.org/wiki/2013_European_floods`

following paragraphs. Based on these tasks, the study was conducted as a combination of think-aloud task assessment, open review stage, and a questionnaire about the system's usability. A specific strength of this evaluation are the twenty-nine experts with different backgrounds and professions as well as the large number of institutions and companies that participated in it. This resulted in multiple different accounts and opinions on the problem. The participating institutions and experts will therefore also briefly be introduced.

Data and Tasks

The following analytics tasks were based on events that happened during the 2013 German Floods. This disaster heavily affected multiple regions in the southern and eastern parts of Germany. Back then, almost all geo-referenced Twitter messages written in Germany between June 1 and 10 were collected for the study, comprising a dataset of about 250,000 messages. Four tasks in crisis intelligence were created that had to be solved by the domain experts using *ScatterBlogs*. The tasks were based on the actual events, which are reflected in the data. Their design encouraged the use of different capabilities to discover and detect relevant anomalies and messages.

Task 1: The first task asked participants to obtain a general overview of the situation in Germany and to find possibly relevant events. A straightforward way to address this was to use the *TagMap* on a low zoom-level to find larger events through zooming and panning. Temporal navigation using the time-range filter was furthermore helpful to explore how the situation developed. Once an overview was acquired, the participants were asked to analyze the most prominent events in detail and extract more information such as the number of Twitter users involved. As the *TagMap* projects cluster size to tag size and indicates unusual tags by color highlighting, inspecting the largest and colored tags was a good way to proceed. The analysts could then identify catchwords like `Hochwasser`, which means `flood`, but also other events like `Rock im Park`, which indicates a larger German rock festival. Besides these actual events, there were further frequently used tags like `Neuschwanstein` and other tourist hotspots shown as white tags in the data. The situation can be observed in Figure 7.1.

Task 2: In the next task, participants were asked to drill down to investigate prevalent topics in the city of Magdeburg between June 7 and 10. A good solution was to first activate the *ContentLens* and use it in combination with the time-range filter to examine the messages in the respective spatiotemporal frame. Doing so revealed topics like `flood`, `dikes`, `sandbags`, and `evacuation`. By selecting the corresponding messages in the area, the analysts could further

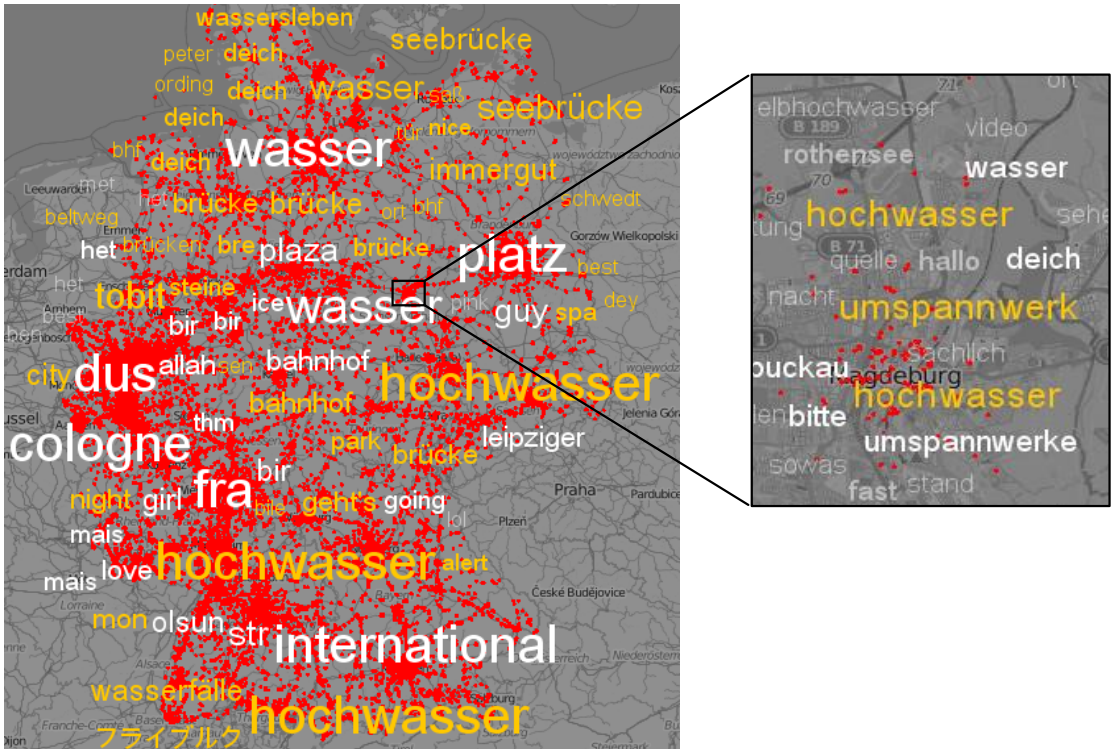


Figure 7.1 — Events of the floods are automatically detected and visualized on the *TagMap*. Unusual topics are highlighted by the *idd* measure. Right: Zooming into the map shows smaller sub-events (`hochwasser` = flood, `umspannwerk` = transformer station). © 2015 IEEE

investigate detailed content in the list, which revealed that more help was needed in building sandbag barriers, and that the situation was indeed severe. At this point one could again try to activate the *TagMap*, which would show additional smaller events that happened in the area. This was particularly helpful as the tag `umspannwerk` (=transformer station), which then appeared, indicated that a critical infrastructure might have been affected by the flood, as can be seen in the right part of Figure 7.1. This hypothesis could be further validated by examining the messages in detail, which revealed that the structure was half submerged and thus indeed severely in danger.

Task 3: The third task centered on messages in the area of Frankfurt. Participants were asked whether there was a period between June 1 and 10, in which the term `police` was frequently used. This could be solved by using a geographic keyword search for Frankfurt in combination with the textual search for `police`. By looking at the temporal overview, analysts could then immediately observe a peak of police-related messages largely dominated by negative

(=red) sentiments. Investigating the respective timeframe and area with the *ContentLens* revealed a demonstration with police presence that happened in the city center and selecting the messages revealed information about the usage of pepper spray and water cannons (see Figure 7.2).

Task 4: For the last task, participants had to examine in which area the flood was most severe on June 8, and they were asked to provide a reliable number of related messages written in the area. In this case, the message classifiers were a good way to proceed, as they would comprehensively detect all flood related messages regardless of the keywords that they used. After selecting a flood classifier from the provided library, the system asked experts to apply a suitable symbol and color for detected messages. Doing so immediately showed multiple symbols with high numbers of flood-related messages on the map. By zooming into the map, the aggregated symbols were split and distributed to more precise locations. After exploring the map and the numbers of aggregated symbols, it would quickly became apparent that Magdeburg, with 45 messages, was indeed the most severely affected region that day.

Procedure

The evaluation sessions were comprised of four stages and were conducted in a situation room or a seminar room at the respective sites. The base duration for the procedure, including only one analysis session, can be estimated at 2 hours. All studies were audiotaped, and the experimenters took additional structured notes in prepared spreadsheets. After introducing the team (1-2 developers, 1-2 usability and/or crisis response experts), *ScatterBlogs* was presented to the group at the respective institution by explaining the usage of the tools and capabilities using real-time data from provided internet connections. This presentation took about ten minutes and followed a standardized procedure, thus ensuring the same knowledge base.

Subsequently, the participants were asked to take control, and the tasks were solved in consecutive order. For each task, the participants first had to decide which tools (e.g. *TagMap*, *ContentLens*, Classifiers) they considered suitable to solve it, and afterwards they had to use it to answer the questions. At the critical infrastructure groups, the system was used to solve the tasks by **6 individuals** themselves. In these cases, only the participant and the experimenters remained in the room. A usability expert from Siemens AG, who was not involved in the system's development, served as the participant's contact person. This experimenter introduced the tasks and collected immediate comments. The participant was instructed to only talk to this experimenter, while the other experimenters just observed the session and took notes. The participant was

furthermore allowed to ask the contact person for help, but was encouraged to do so only if he or she had absolutely no idea what to try next. Right after each task was completed, these participants were asked to rate the usefulness of the tools on a scale from 1 (very useless) to 10 (very useful). At one of the sites (WVV), **3 additional** participants investigated the tasks collaboratively together with the experimenters because of time constraints. Their ratings were not counted.

In contrast to these more rigorous sessions, the groups from disaster response (**20 individuals**) usually just observed one volunteer, who was found at the beginning, or an experimenter using the system. In this case, the task was solved in a collaborative effort by the whole group of participants at the respective site. Participants were encouraged to ask the analyst to apply distinct tools and provide further recommendations, ideas, and comments to advance the analysis. The primary reason for this procedure were the more constrained parameters appointed with these institutions. Individual sessions were not an option in this case.

The task stage was followed by an open review stage, in which the complete group at every site was asked to comment on the tools, the integrated system as a whole, and social media analytics in general. To conclude the session, a questionnaire that collected ratings on the usefulness, perceived performance, and hedonic quality of the system had to be completed by every participant.

Domain Experts: Critical Infrastructures

The experts from the critical infrastructure domain mostly had a background in energy supply and distribution infrastructures. Most of the participants were employed in senior and higher service and had at least 8 years of experience in the field. In addition to decision makers and analysts, administrators and information systems specialists that were familiar with the IT infrastructure of the existing control environments were also interviewed. For the critical infrastructure domain, the interviews were conducted at four different sites:

- **EnBW Energie Baden-Württermberg AG (1 participant)** is one of the four large energy supply companies in Germany and has about 20,000 employees. Their responsibility covers electricity, gas, and water supply for the south-east of Germany. The study participant from EnBW was employed in the R&D branch and specialized in innovative electrical systems and equipment for smart power grids.
- **Stromnetz Berlin (2 participants)** is a subsidiary of Vattenfall, one of the five largest energy supply companies in Europe. SB is particularly

responsible for the electrical grid of Berlin. The two study participants were the department head of high voltage electrical grid management and the information and control systems coordinator.

- **DB Netze (2 participants)** is a subsidiary of Deutsche Bahn AG, the German national rail company, and provides infrastructure and operations support for the German railways. The participants were the department head for energy distribution systems and a control systems administrator at the power systems branch (DB Energie) of the company.

- **WVV (4 participants)** is an infrastructure and energy supply company responsible for gas, water, district heating, and electricity in the city of Würzburg. Here, a control systems administrator and an electronics engineer participated in the study. Two department heads were furthermore available to give comments after the system's initial presentation and participated collaboratively in task assessment in one of the sessions.

Domain Experts: Crisis Response

Experts from the crisis response domain usually had a quite practical background in firefighting or rescue engineering. But there where also participants from higher administration/civil service, public relations, press officers, and volunteers. Six of the experts had more than twenty years experience in the field, and most experts had at least eight years of experience:

- **Gemeinsames Melde- und Lagezentrum (GMLZ) (3 participants)** The German Joint Information and Situation Center is part of the German Federal Office of Civil Protection and Disaster Assistance (BBK), which was established in response to new threats, such as the 9/11 terror attacks and the flood catastrophe of 2002. The GMLZ hosts the central emergency operations center of Germany, where representatives of the local and federal response and relief organizations gather in case of a federal crisis. Its tasks involve the gathering and analysis of situational information relevant to Germany and its states as well as the joint coordination of rescue and support efforts in case of a crisis or catastrophe. Besides that, the GMLZ serves as a contact point to European and international partners. *ScatterBlogs* was presented to a larger group, of which three attendees participated in the study: two dispatchers of the situation center and one consultant.

- **Crisis management groups from Aachen (5 participants), Goslar (8 participants), and Viersen (4 participants)** are mainly staffed by employees

► **Figure 7.2** — The *ContentLens* could be used in the third task to assess the ongoing situation in Frankfurt. The three largest tags say `police`, `encircle/empocket`, and `protest`. The view below shows the volumes of positive (=red) and negative (=green) sentiment tweets for the keyword `police` in the temporal overview. © 2015 IEEE

of the city council or administrative district. They are usually only summoned during critical events, and their tasks include planning of staff and resources, situation awareness, crisis communications, and public relations. They support decision-making and ensure fast and effective crisis response in collaboration with federal authorities. In these councils, firefighters in senior and higher service, administrative and press officers, as well as rescue engineers were interviewed.

7.2.2 Findings and Comments

This result section will first report how the experts rated the usefulness of the individual tools after applying them to the tasks. In addition, observations that were made about the performance of the tools and possible problems are also reported here. Results of the open review stage will be provided by summarizing the collected comments and suggestions.

Task Performance

Participants usually performed well on all tasks and solved them in a matter of minutes. If they asked the contact person about support, they only required help to choose the right tool, as they forgot about functions from the introduction. Once they applied it, they had no problem in getting the right insights from it. The results of the ratings given by the 6 critical infrastructure experts that used the tools themselves are shown in Figure 7.3. It can be seen that the participants gave mostly favorable grades with means ranging between 7 and 9.

However, several issues that occurred during usage were observed by the experimenters. For example, when first activating the *TagMap*, many participants

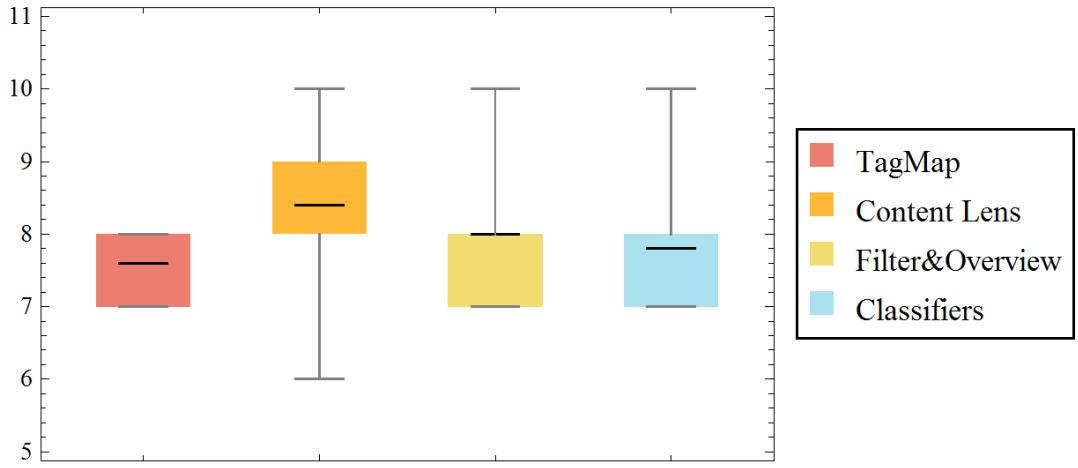


Figure 7.3 — Means and quartiles of ratings given for the usefulness of tools *ContentLens*, *TagMap*, and Classifiers. Grades could be given from 1 (very useless) to 10 (very useful). © 2015 IEEE

were still overwhelmed by the high amount of tags presented on the map, as not all tags corresponded to actual events. Although the relevant unusual tags, such as `flooding` or `sandbags`, were clearly visible as the largest *idd*-highlighted tags, some users needed quite some time of moving and zooming the map until they noticed them. This clearly demonstrated that highlighting relevant items based on color- and scale-mapping might sometimes not be sufficient if the surrounding visual space is over-allocated.

It was furthermore experienced that some users initially had difficulties to understand the distinct capabilities of the *ContentLens* and *TagMap*. While the lens just highlights the most frequently used or most unusual terms in selected message sets, the *TagMap* only shows tags if a spatiotemporal cluster of similar messages has been identified. This, however, became clearer when both tools were applied on the same task, e.g., to investigate the area of Magdeburg. In this case, the lens just visualized frequent concerns of people affected by the flood, while only the *TagMap* indicated the affected transformer station. Additionally, when used to highlight most unusual tags, the *ContentLens* might be missing capabilities to compare the actual situation to the normal situation, e.g., by displaying current vs. average message counts for the given tags.

Comments and Suggestions

The open review stage can be considered the most important part of this study. It consisted of two segments of questions. The first segment evaluated

the applicability and usefulness of social media analytics with *ScatterBlogs*. Participants were asked:

- **How they assess social media analytics with the system in general:** The experts from critical infrastructures generally commented that it was "fun to use", the techniques were "easy to handle", and that the idea to analyze social media in critical infrastructure management is "interesting and needed". However, two participants from energy power grid management also commented that they are not sure what kind of task they would address in their domain, and that there might not be enough human resources to employ such tools or to listen to insecure information channels. ("I cannot assess how credible the messages really are.") The experts from disaster response also commented positively about the usefulness of the system and social media monitoring in general. ("Every operations center would be thankful to have such a tool.") At the same time, they had even stronger concerns about reliability of the data and/or credibility of the social media users. They generally felt that social media monitoring should be seen as an additional means to provide information in areas where no other sensors and/or own observers are available and as a tool to harvest public opinions about response measures. Participants also raised concerns about the legal situation with respect to privacy rights and of decision-making based on uncertain and unreliable information.

- **What possible usage scenarios they see in their respective domain:** The disaster response experts saw the techniques particularly suited to provide an information backchannel during and after critical situations. They felt that they could better assess ongoing situations; especially if multiple observers report consistent information about the same event, which would enhance the reliability of data. One group also indicated the possibility to oversee evacuation measures through public comments. In terms of organizational structures, they saw the usage of such tools on the federal level, from where observations and reports from professional analysts could be distributed to local responders, or on the municipality level, where volunteers could also help in crowdsourcing analytical efforts. One participant also commented on the political dimension of social media channels that now demand from responders to directly communicate with the public. He therefore suggested the necessity of specialized teams that use the tools to oversee trends and aggregate public communication. The critical infrastructure experts considered possible usage in collecting observations about power outages, problems with distributed heating, and also a more direct way of collecting environmental information, such

as local weather conditions. ("I think it is faster than our weather radar.") They saw new possibilities to get explanations for events that they only observe through sensor data, and to collect vital observations that people falsely consider too irrelevant to make an emergency call. However, they also commented that it will be difficult to integrate the approaches into the well established IT systems and workflow, and that it might take some years until such tools could be implemented in daily use. ("Our power grid management system has significant security requirements.") In this context, they also considered it problematic that operators would often only apply the tools in critical situations and not in daily business, making it hard for them to stay familiar with their usage.

- **Which capabilities were most useful in solving domain challenges and why:** Here, several participants highlighted the means to cope with large volumes of data through classifiers and automated event detection. ("This is tremendously important because I can specifically highlight messages relevant to my work.") They particularly liked that both of these tools are real-time-enabled (as was illustrated in the tutorial), giving them a feeling to be more directly connected to ongoing events ("As this can be applied in real-time, I have to say this is really really impressive.") In that respect participants highlighted that gathering information with the tool seemed tremendously faster than administrative reporting channels. ("I'm not looking at something that happened half an hour ago, but right now - in real-time!") Almost all participants liked the simple and familiar means of geo-referencing messages directly on the map, which they considered a key factor in making the tool useful for their domains. They also liked it that the tool is similar to well-known applications, like Google Maps, as there would probably not be any specially trained personnel available to use it. In the best case, it should even be accessible to untrained volunteers.

The second segment of questions addressed the changes and improvements that would be needed to adapt existing research to practical usage. Participants were asked:

- **What kind of tools they would be missing in the system:** The emergency responders criticized that there would be no tools available that specifically address images and videos from the posts. Although the media associated with a post can be shown in *ScatterBlogs* by selecting it, the participants asked for more sophisticated means of aggregating and organizing this media, or even means to use them as a basis for event detection. As a particular example they recommended that the TagMap should also show

representative images from clusters in a sidebar or directly on the map. It was emphasized on several occasions that images and videos would be much more valuable to the responders than plain text, as it gives them both a better means to verify eyewitness data and a more easy way to truly assess the severity of the ongoing situation. The disaster response experts furthermore recommended to develop tools for credibility checks of event reporters, e.g., by automatically analyzing a user's profile, the content of his previous posts, and his standing within the community. The experts from critical infrastructures commented that it would be important to get a complete picture of public reactions and opinions. It would thus be needed to collect from multiple sources of social media posts, but they would have to be presented in a unified and integrated fashion. In addition they would like to see sensor data from their own infrastructure, such as the energy or gas distribution grid ("Overlays that additionally show data from our grid infrastructure would be helpful."), and other information layers, such as traffic data, to be shown on the same map on request.

- **What tools and visualizations of the presented system they consider dispensable:** For this question most participants did not give a definitive answer, and the few available comments were centered on issues with the implementation's usability. For example, one participant from critical infrastructures criticized that the temporal overview should be shown "less prominently", and that the map should be "shown in full-screen". By contrast, one of the emergency responders commented that the map should be smaller or it should even be possible to "deactivate it on demand", as he was more interested in trends and individual message contents.

- **What the participants considered too complex about the presented techniques, and what they would possibly change about them:** Experts from both domains commented that the symbols used to show messages detected by classifiers should resemble icons known from their existing control environments. For example, messages mentioning power outages should use the corresponding symbol conventions from grid management software, and messages reporting about evacuation or response measures could use tactical symbols from the BBK/SKK recommendations [SKK, 2010]. One expert from critical infrastructures indicated that she sometimes had difficulties understanding which filters and modules were currently active. She therefore recommended to provide textual highlights of what users are currently observing and what inputs they

had performed to reach that point. For example, some kind of search or filter history could be helpful. Some experts also commented that it was cumbersome to manually activate and deactivate overlays to get a better view. The system should thus automatically decide whether certain overlays should be hidden or grayed out based on the analyst's last actions. For example, if analysts have just created a combined message classifier, they will be specifically interested in detected messages, and the TagMap should thus be deactivated or at last reduced to the few most important tags. Some experts indicated that the TagMap sometimes shows "too much information", and that so many words are "initially confusing". However, they also commented that "the highlighting significantly helps [...] to identify the relevant elements."

7.2.3 Survey

In addition to its relevance, the perceived pragmatic quality, the hedonic quality, and the overall attractiveness of the *ScatterBlogs* system were also assessed in the study. The AttrakDiff questionnaire has been developed by Hassenzahl et al. [2003] to measure the perceived performance of tools, techniques, and products with regard to their usability and visual appearance. The perceived attractiveness of visual interfaces is gaining more and more importance in usability design. Traditional evaluation procedures, however, are only focused on usefulness and operability (pragmatic quality). AttrakDiff thus consists of twenty-eight items of semantic differentials, all ranging from -3 to 3, measuring qualities in these four dimensions:

- **Pragmatic Quality (PQ)** - The ability of the system to tackle domain challenges by providing useful and applicable features.
- **Hedonic Quality - Stimulation (HQ-S)** - The ability of the system to satisfy the need to improve own knowledge and capabilities.
- **Hedonic Quality - Identity (HQ-I)** - The ability of the system to communicate self-serving messages to relevant other people (e.g. co-workers, superiors).
- **Attractiveness (ATT)** - The overall appeal of the system's visual and interaction design.

The questionnaire was given to the participants right after the open questions session, and they were instructed to rate the complete system including all tools

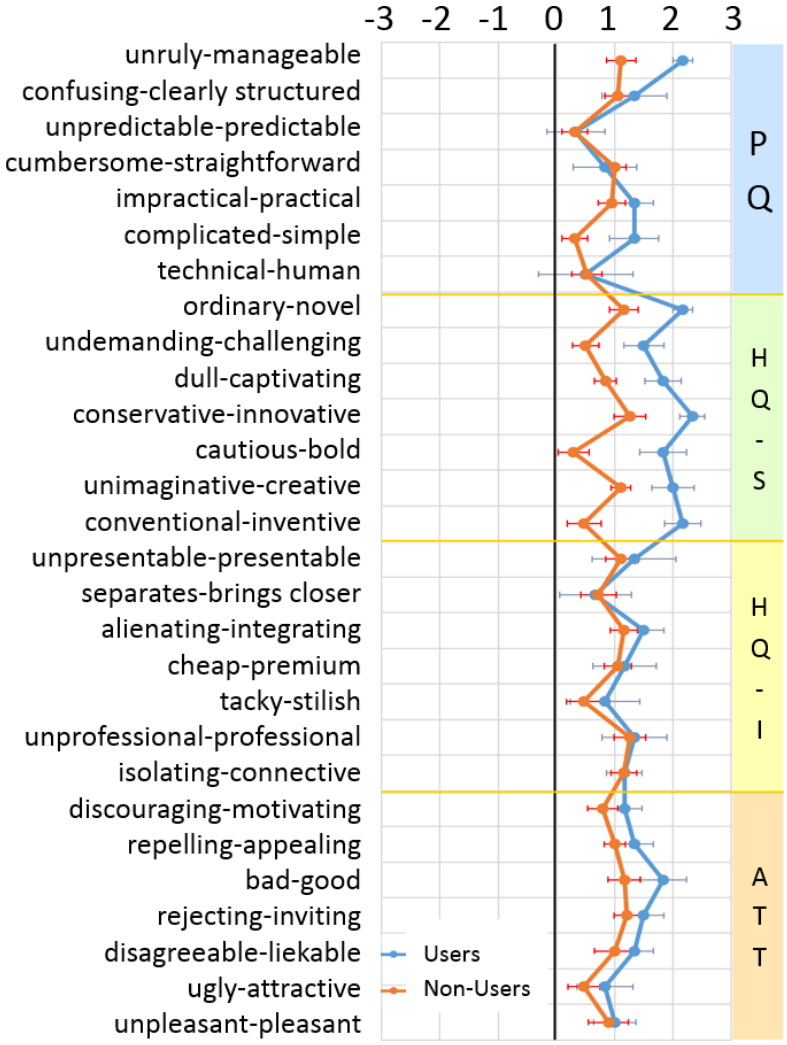


Figure 7.4 — Means and standard errors for the 28 items of the usability questionnaire. © 2015 IEEE

presented during the introduction, no matter whether they used them to solve the tasks or not.

The mean results for each of the items can be seen in Figure 7.4. Results of participants who were actually using the system themselves are shown in blue. These were primarily users from the critical infrastructure domain, as mentioned in Section 7.2.1. Results from participants who just saw the presentation and observed others solve the tasks are shown in red.

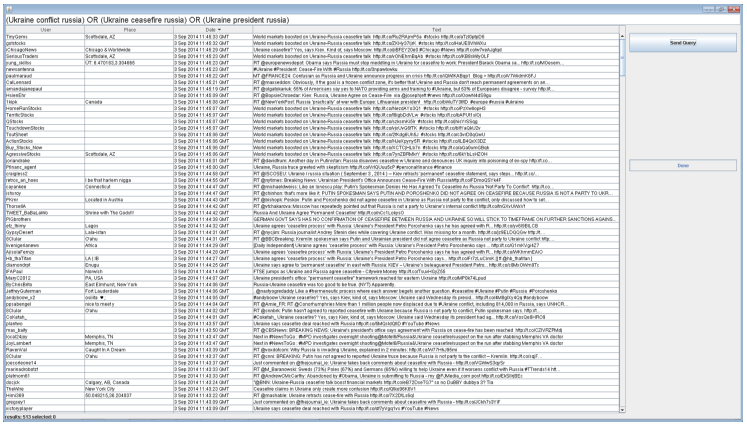
The results thus indicate that after using the system the participants were generally more convinced of its manageability, practicability, simplicity, and overall hedonic quality. It can be concluded that coping with the real-world tasks made participants more familiar with the actual challenges of analyzing the data, and gave them a better idea how the more advanced tools like the *TagMap* and classifiers can help them to improve their knowledge.

7.3 *TreeQueST* User Study

Assessing the capabilities of query optimization is a complicated endeavor due to the vast volumes and real-time nature of the *hidden data* that constitute the inherent challenge. The activities of the *TreeQueST* component are manifold. The ultimate goal is to create queries that match a given information need. However, these queries are not created by the system alone but iteratively informed by the analyst's ever-increasing understanding of the situation. *TreeQueST* is used to collect samples, organize them, explore them, tell the system what is important, and - based on this information - investigate the next sample. At the end of the process, analysts are also supposed to manually adapt the proposed query, and to have acquired a mental model of the corpus that hides behind the web interfaces. It is therefore relevant to assess how well *TreeQueST* supports the user in general information retrieval tasks.

However, performing a user study with a previously gathered ground-truth dataset has two major drawbacks. First, as the study would not be performed in real-time, the participants might have already acquired knowledge about the targeted event or topic, e.g., from rumors or news media. And secondly, this data would only comprise a small sample of the actual data volumes produced in the services. The whole point of the approach, however, is to cope with the problem of hidden vast data that only allows users to pick smaller extracts, and then use them to infer the complete picture.

To make results comparable and counteract issues of pre-collected data, the user study was thus conducted during a single day, on September 03, 2014, using



► **Figure 7.5 —** The image shows a snapshot of the plaintext search tool created for the study. Queries could be entered in the same fashion as in *TreeQueST*. 1000 results were listed per request.

online API access to Twitter. At that time, the 2014 Ukraine crisis[2] and the threat of ISIS terrorists in the middle east[3] were hot topics of global relevance.

7.3.1 Experimental Setup

The study was conducted with **6 participants** - all graduate students at the University of Stuttgart computer science department - who assumed the role of political journalists tasked with investigating these topics. Two broad seed queries, `ukraine` and `barack obama`, which were both frequently mentioned media entities during the day, were initially selected. Similar to the case study (Section 3.6), they were asked to find relevant events connected to the keywords, investigate the communities reactions and opinions, and identify news content that could dominate further media debates. Ultimately, the goal of the participants was to incrementally refine their query vocabulary and come up with requests that would allow ongoing information harvesting for the respective topics. For each participant, one of the two seed queries to be investigated with *TreeQueST* was randomly selected. The other seed query had to be investigated with a plaintext search tool based on the Twitter Search API (Figure 7.5). By this means, a baseline was established to compare the results. For each tool, the participants were given five minutes to get familiar with it, and then they had ten minutes to investigate the topic. To conclude the evaluation, they were asked to rate both tools in an anonymous online survey. The participants were encouraged to document all relevant findings, observations, and problems by taking screenshots and providing think-aloud comments during their investigation.

[2] `http://en.wikipedia.org/wiki/2014_Crimean_crisis`

[3] `http://en.wikipedia.org/wiki/Islamic_State_of_Iraq_and_the_Levant`

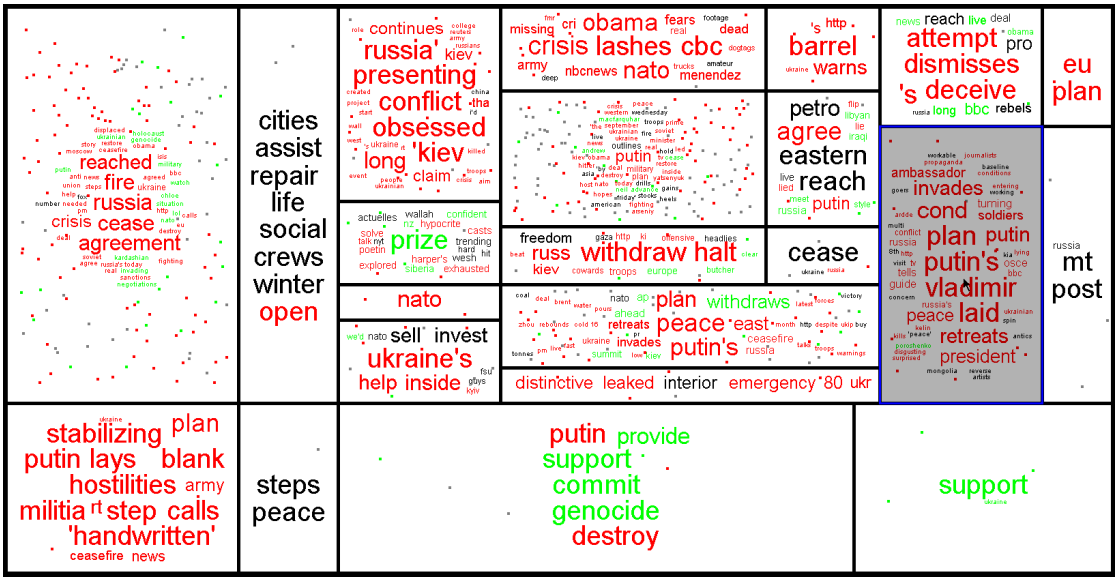


Figure 7.6 — Snapshot taken during the user study. The participant just entered the seed query `ukraine` and activated sentiment highlighting. The overview indicates the cease fire and Putin's alleged peace plans. © 2015 IEEE

7.3.2 Findings and Comments

Major news items surrounding the given topics were usually immediately indicated by *TreeQueST* right after the seed query was entered, as can be seen in Figure 7.6. The participants thus easily discovered that the journalist Steven Sotloff was executed by IS terrorists, that Barack Obama publicly reacted to that in a speech, and that a temporary cease fire between Russia and Ukraine had been signed. Also, most of them discovered a speech that was given by Obama to reassure Estonia and other Baltic states that defenses in eastern Europe will be reinforced. Several participants commented positively on the indication of entities, such as names, places, and institutions, that could be easily connected to the topics using the tag clouds.

Some participants had already learned about few of the major news items from other media before participating in the study on the same day. When using the plaintext search, they commented that they would perform equally well on these topics, since they knew in advance which queries might be successful. However, participants also commented on several occasions that they tend to randomly read messages in the top-ranked documents before they try a new query, and that they thus ignore possibly relevant information on the following pages.

With *TreeQueST* all participants made heavy use of the possibility to drill down on major topics in order to find subtopics, to highlight query words, and to create new queries based on selected topics. By this means, participants discovered additional newsworthy information, e.g., that US stocks on wall street opened higher because of the cease fire, that the announcement of an ongoing truce from Ukraine was initially denied by Russia, which lead to ongoing confusion in the community, and that several Twitter users requested Obama to give back his Nobel peace prize. Participants commented that the plaintext search did not support similar discoveries, as it gave no clue in which direction they should broaden their queries. Two participants made frequent use of the possibility to highlight the sentiment of tweets and tags, which, however, did not lead to significant additional findings in these cases.

To compensate for the drawbacks of missing ground-truth data, major news media was intensively searched on the following days. This investigation revealed that no significant information entities related to the topics had been missed by the participants using *TreeQueST*.

7.3.3 Survey

The online survey was conducted on the same day to collect unbiased impressions and prevent participants from sharing their experiences with each other beforehand. The survey comprised one questionnaire for each tool consisting of Likert [1932] scales and written feedback. First, the participants had to rate five subjective statements about the tools on a 5-point Likert scale, ranging from 1 (*strongly disagree*) to 5 (*strongly agree*). The statements corresponded to five defining elements of information retrieval performance:

- **Overview** - *I could get a good overview of the topic.*
- **Precision** - *I could quickly find relevant news items.*
- **Recall** - *I did not feel to miss any relevant information.*
- **Optimization** - *I was enabled to optimize my queries.*
- **Usability** - *The tool was easy to use.*

The results in Figure 7.7 show that *TreeQueST* enabled participants to get a better overview of the topic and the feeling to achieve better retrieval results. Compared to the values for the plaintext search, the results particularly support the claim that *TreeQueST* optimizes recall of information foraging while at the same time maintaining acceptable precision. This corresponds to the

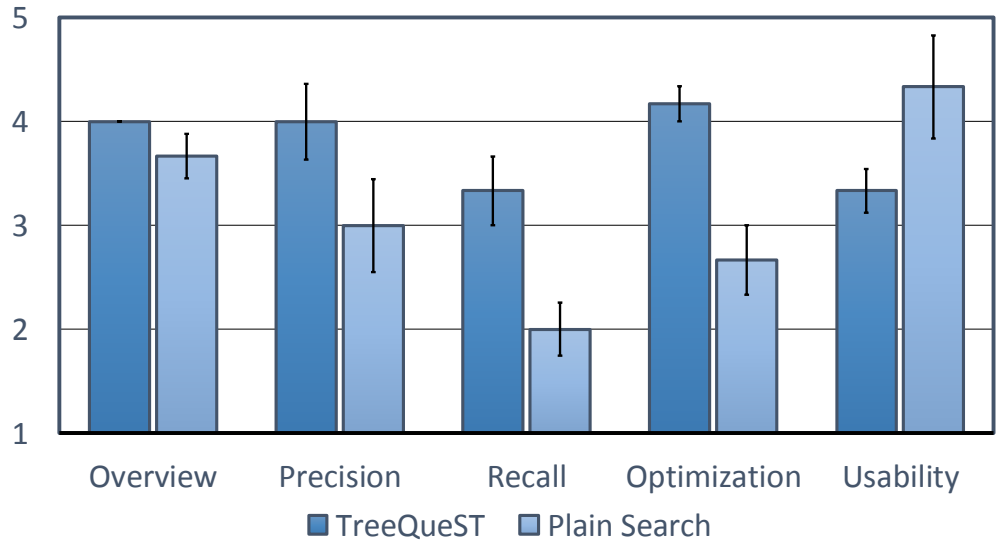


Figure 7.7 — Mean Likert scores and standard errors of the search experience evaluation. *TreeQueST* performed better in all aspects except ease of use. © 2015 IEEE

primary goal of tackling limited data access without ignoring possibly relevant information areas.

The participants also felt better supported in optimizing their queries based on initial retrieval results. Correspondingly, it was observed during the investigations that participants tried to find further query words in retrieval sets of the plaintext search, but they commented that additional support would be helpful. The results furthermore show that *TreeQueST* has a steep learning curve, which reflects the observation that not all features were used by all participants, and that some participants mentioned that they had already forgotten features from the introduction.

On top of the system comparison, participants were also asked to rate individual features of *TreeQueST*. The results in Figure 7.8 show that visualizing information in a tree hierarchy, aggregating topics as tag clouds, and the automated query generation were most favored by participants. Sentiment coloring and the exploration lens received lower average scores, corresponding to the observation that no significant finding was made on this basis. However, it was already mentioned, that only two participants used it more frequently. Results of the earlier Oculus VR case study as well as the performance of the *ContentLens* in the expert study indicated that there are at least some situations where such tools can be of significant benefit.

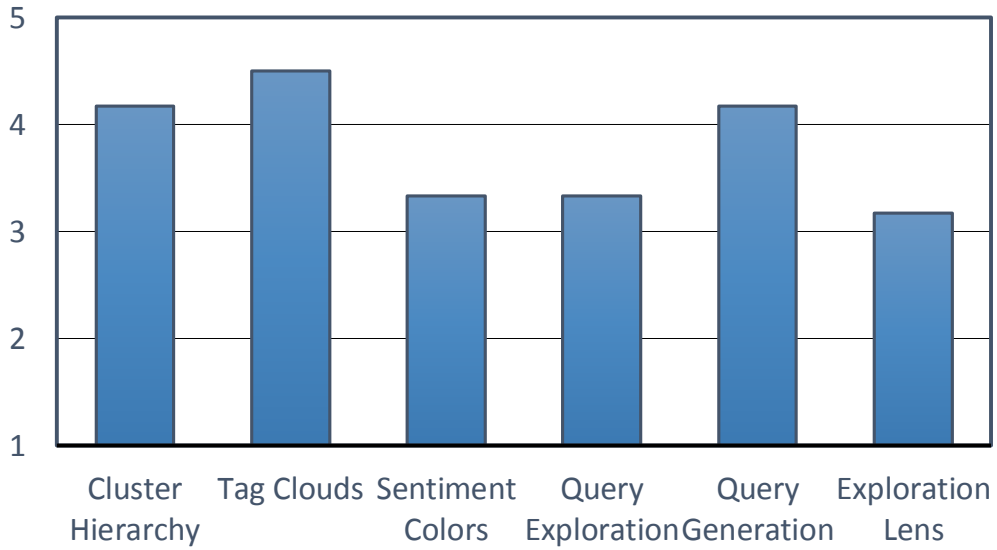


Figure 7.8 — Mean Likert scores of the *TreeQueST* feature evaluation.

In the next step, feedback had to be given based on two questions: First, the questionnaire asked what the participants liked better about each tool compared to the other. They liked about *TreeQueST* that it felt more powerful, provided better overviews, gave them more control over the investigation, and was more enjoyable to use. They also mentioned that the structured overview of *TreeQueST* would motivate them to read tweets from various subtopics, which they might have ignored in the plaintext search, as they rarely looked beyond the top-ranked documents.

About the plaintext search they commented that it was more easy to use and to learn. One participant also mentioned that it was a more stateless approach, meaning that he would not lose relevant views and preliminary analysis results when re-iterating the query. In the second question, the participants were asked, what they would like to change about the tools. Among others, they commented that *TreeQueST* should have a separate text field for query testing, that the query words should be highlighted in the tweet table, and that the tag clouds should be improved in terms of scaling and colors.

To conclude the questionnaires, a German school grade, ranging from 1 (best) to 6 (worst), had to be given to rate the overall usefulness of each tool in supporting the task. On average, plaintext Twitter search scored 2.8, and *TreeQueST* scored 1.8, both with a standard error of 1.6.

CHAPTER 8

Conclusion and Outlook

Just as *ScatterBlogs* is a prototypical implementation of the approaches presented in this thesis, the approaches themselves can be considered as patterns for solving the distinct problems in achieving situation awareness from social media. By providing the templates, and by showing how they must be arranged in an overarching model, it was demonstrated how key ideas of visual analytics can be utilized to solve the general challenges of coping with large and streaming textual data. This chapter summarizes the contributions of this thesis, discusses results and lessons learned, investigates how the methods can be generalized to other domains and problems, and concludes with final remarks about the ongoing challenge.

8.1 Summary of Contributions

In the beginning of this thesis, four central research challenges were identified based on requirements of domain experts. They comprise the questions of how data can be accessed from request- and throughput-limited web APIs; how information from this data can be extracted and presented in context of space, time, and related content; how the semantic complexity of messages can be handled in ongoing monitoring; and how processing and management of the data can be facilitated in a scalable and real-time-enabled fashion. Following these requirements, the thesis presented three corresponding methods, an over-

arching analytics model, and a prototypical implementation which integrates them.

Chapter 3 highlighted that the challenge of data access is caused by limitations of existing tools for information retrieval of microdocuments. Traditional web search is focused on retrieving larger, comprehensive documents. By contrast, the information in social media platforms if often heavily distributed, redundant, and it is hard to automatically assess the relevance of individual entries. With *TreeQueST*, a method was presented to repeatedly retrieve data samples from the APIs and explore them in a comprehensible fashion. To this end, the notion of agglomerative clustering is used to create a hierarchy of message groups. Based on a newly defined similarity metric, the hierarchy resembles the nested structure of topics and subtopics existing in the sample. Messages that belong to the same topic are related with each other, and the corresponding similarity groups are summarized by tag clouds and a spatialization of messages. Actual topics can be easily recognized by looking for coherent and meaningful tag sets. At the same time, less frequent topics and signal noise are organized in deeper levels of the hierarchy. Following the well-known Scatter/Gather exploration scheme, the analyst can zoom-in on the hierarchy, select possibly relevant topics, and re-iterate the clustering based on the results. The basic scheme was then enhanced by a method for automated query creation and validation to cover the remote social media corpus. A real-world media awareness study illustrated how users can enter an analysis loop that continuously advances their query vocabulary, the quality of retrieved samples, and their own understanding of the larger social media corpus. The usefulness and applicability of the approach as well as the validity of the design decisions were demonstrated in a comprehensive user study. Although some of the tools were not easy to apply and might require some training, the users were convinced that *TreeQueST* would significantly enhance their retrieval efforts compared to traditional web search solutions.

The problem of extracting event information, relating it to spatiotemporal context, and visualizing it in comprehensive overviews was addressed in Chapter 4. With the *TagMap*, a technique was presented that finds clusters in streaming social media messages, automatically identifies possible anomalies, and visualizes the results to facilitate event discovery. It allows adaptive zooming by automatically re-calculating the position and size of visible labels and by combining similar ones to optimize the usage of available screen space. The strengths of the approach result from the specifically adapted clustering algorithm, which allows scalable processing to enable real-time aggregation and indication within vast volumes of messages. The inherent drawbacks of this heuristic algorithm, particularly the problem of overfitting, are complemented by the adaptive vi-

sualization. Multiple similar clusters are seamlessly merged with each other in zoomed-out views to indicate the extent and location of events on a global scale. However, as the clustering considers each term individually, small events with just few related messages can still be discovered within large volumes of unrelated messages. Highlighting of such smaller anomalies is further boosted by a significance function that rewards the spatial closeness of related messages to determine the size of the labels. This helps to detect localized incidents such as lootings, which were seen in the London Riots case study, or power outages, as investigated in the Hurricane Irene case study.

However, the case studies also showed that problems can occur with visual clutter and unproportional label sizes. The former is a consequence of random clusters, and the latter is caused by different base frequencies of terms in differently populated areas. Just normalizing label sizes to population densities, however, would not be a solution here, as they do not directly correlate to the actual distribution of social media users or the base popularity of topics. To tackle this challenge, the *idd* measure was presented as an extension to exploit global term and document distribution based on recorded data. Using adaptively pre-aggregated data structures, the described implementation strategy allows to quickly compute location-sensitive term weights. The study results showed that the measure comprehensively estimates the abnormality of terms and topics, and that its performance is suitable to support real-time processing as part of the *TagMap* and *ContentLens*.

Using these tools for exploration provides an entry point for further investigations. By means of interactive classifier orchestration, the analyst's observations and insights can then be externalized to reliably detect relevant entities. To this end, it was shown in Chapter 5 how interactive exploration of past and recent situations can inform the creation of supervised filters for ongoing and future events. In a long-term workflow which exists orthogonal to the overarching analytics cycle, analysts can define basic information need with visual classifier training. These building blocks are then arranged to build comprehensive monitoring configurations or to enable drill-down operations in time-critical environments. With interactive controls, analysts can furthermore configure the trade-off between precision and recall of orchestrated filers. This can be used to incrementally examine dataset sizes that correspond to current temporal constraints.

In Chapter 6, the *ScatterBlogs* visual analytics platform was introduced. It implements all of the presented methods and allows to employ them in an integrated fashion as part of the overarching analytics cycle. The chapter showed how data structures can be defined to allow scalable real-time processing and interaction with spatially, temporally, and textually indexed data. The

ScatterBlogs system architecture fosters a seamless transition from post-analysis of archived data to ongoing monitoring of current events. Furthermore, the user interface provides powerful means for searching, filtering, and exploring social media messages. To allow fast assessment of larger message sets, the system provides additional text aggregation tools based on exploration lenses and LDA topic extraction.

In its three-year development time, the software quality of *ScatterBlogs* was enhanced to a near production-ready level. This allowed comprehensive evaluation of the individual techniques and their interplay. By contacting various larger companies and government authorities with sophisticated experience in situation awareness, the usefulness and applicability of the system were validated together with almost 30 domain experts. The results of task-driven analysis sessions, open review rounds, and a comprehensive questionnaire showed that the approach is suitable to facilitate sophisticated situation assessment. The experts highly appreciated the real-time nature of the techniques. While some analysts first had difficulties to understand the usefulness of more advanced tools, like the *TagMap* or classifier orchestration, the application to tasks in a real-world setting quickly convinced them of their necessity. In their comments, they positively highlighted the means to discover previously unexpected information, gather vital observations in a directed fashion, and spatially relate messages to ongoing events.

8.2 Lessons Learned

In the course of this thesis, visual analytics methodology was employed to enable comprehensive analysis of social media data. However, the experiences made in developing and adapting the methods, comparing them to other approaches, and evaluating them with domain experts could also help to further inform and advance the research. This section highlights two distinct lessons that were learned by employing machine learning and by tackling the challenge of collecting required web data.

8.2.1 Two Sides of the Medal

"Detect the expected and discover the unexpected" is a creed that the research agenda of Thomas and Cook [2005] embedded at the very heart of their visual analytics conception. In their work, the statement particularly relates to the process of enhancing and enabling cognitive reasoning by means of visual interfaces. Although automated means for organizing and transforming data were also discussed [Thomas and Cook, 2005, Chapter 4], they do not seem to

constitute a key element in their definition. More recent conceptions by Keim et al. [2010] emphasized the particular role that automated statistical algorithms should play in the interactive workflow. Here, the statistical model element is introduced as part of an overarching knowledge generation cycle. Keim et al. highlight that the models can be automatically derived from data by unsupervised methods, and that the analyst can be allowed to configure, steer, and orchestrate them.

Building on these conceptions, this thesis demonstrated how the challenges of *detection and discovery* tightly align with *supervised and unsupervised* techniques in machine learning methodology. To enable visual analysis of streaming text in time-critical decision-making, analysts must consider both aspects. While cluster analysis helps them to quickly identify events and entry points for the analysis, classification is suitable to prevent them from missing possibly relevant reports. Beyond previous conceptions, it was demonstrated that these techniques can even complement and inform each other in a loop of discovering what is relevant; use these insights to define what needs detection; and employ discovery again to further explore the important. By opening up and adapting the black box of clustering and classification through visual interfaces, this loop is realized by the sensemaking part of the overarching analytics cycle.

What analysts can learn from ongoing events through exploration with the *TagMap* enhances their reasoning and builds up experience. It can henceforth help to inform short-termed orchestration as well as long-term creation of classifiers. After applying these filters, the unsupervised exploration means can then again be used to investigate the results and adapt classifier behavior. Finally, if additional data is required or if generated hypotheses open new paths of possible inquiries, analysts can move back to the larger cycle and re-enter information foraging with *TreeQueST*.

8.2.2 The Latent Challenge

The *TreeQueST* approach addresses a problem that was frequently underestimated or ignored in past research. Existing data mining solutions for the big data era often assume that their data is readily stored on some local database or some high-throughput distributed system - providing fast availability and almost unlimited access. The approaches often rely on highly scalable architectures and apply sophisticated statistical models to retrieve information. However, the actual situation of analysts sometimes differs from the ideal case, in which they could successfully employ these methods.

Although the number and size of community driven Web 2.0 platforms has tremendously increased over the last decade, so has the monetary value of the

data they possess as well as the limitations they impose on accessing it through their APIs. While well-financed multinational corporations and intelligence agencies can make arrangements for comprehensive and broad social media monitoring, others, including journalists, humanitarian NGOs, researchers, or crisis response agencies - who could heavily benefit from web intelligence as a social sensor - often lack the necessary resources. With the *TreeQueST* approach, a method was found that allows analysts to explore remote data volumes with limited accessibility piece by piece. By this means, they can incrementally build a complete picture of the available information and decide on strategies to establish ongoing, pre-filtered data collection.

In recent years, Scatter/Gather and agglomerative clustering have both lost popularity in the ongoing scientific discourse. In case of Scatter/Gather, the reason can be found in its lower efficiency for web information retrieval compared to traditional search engines. In case of agglomerative clustering, the primary point of criticism is its poor computational complexity, which seems to render it unsuitable for problems with large data. However, with the *TreeQueST* approach, it has been demonstrated that these algorithms are perfectly fit for the challenge of microdocument retrieval. In this domain, traditional search engines fail to deliver satisfactory results because they do not cope well with highly fragmented and redundant information sets. By contrast, Scatter/Gather particularly relies on the possibility to find similarity groups from distributed content. Moreover, by only investigating small samples for query optimization, the limited processing capabilities of agglomerative clustering do not constitute an actual problem in the implementation. In contrast to faster partitional clustering schemes, they automatically generate an information structure that is suitable for interactive exploration. This also allows to overcome drawbacks of automated model fitting.

8.3 Generalization

The approaches presented in this thesis can be extended and generalized in multiple directions. Social media analytics is essentially just one instance of the more general problem of text and streaming data analytics. In this regard, the thesis contributed techniques that can help to cope with retrieval and classification problems of very short informal text documents and the discovery of spatiotemporal events in real-time sensor data. Furthermore, the *idd* language model (Section 4.4), which was derived from Twitter data, could also be used to spatially normalize term frequency in other text analysis settings. Corresponding challenges can be found in analyzing SMS (short message service) data from cell-phones, in organizing business communications,

in harvesting general geolocated web data, in enhancing movement data with semantic annotations, and in managing computer generated text.

An example for the latter, was provided by DB Energie, the power grid branch of Deutsche Bahn (introduced in Section 7.2.1). They employ a system that automatically transforms sensor readings of the power grid into textual snippets. It reports parameters and problems with the infrastructure together with a location of occurrence. Currently, this machine-generated "microblog network" needs to be manually filtered and explored. In this case, techniques like the *TagMap* and classifier orchestration could also help to aggregate information, discover patterns, or to classify messages beyond keyword- and meta-data-based selections.

The challenge of visually relating social media messages to corresponding movement behavior of users was already addressed in the context of this thesis. In [Krüger et al., 2012b, 2014b,a] it was investigated how GPS data from other sources, in this case electronic scooters, can be related to Twitter and Foursquare content at location hotspots. By this means, analysts can investigate the reasons and context of movement to understand underlying patterns. Instead of just observing plain trajectories and stop-points, aggregation techniques like the *ContentLens* are used to summarize historic messages in the surrounding areas. This provides background and indicates to analysts why places are frequently visited and how they relate to other parts of the movement path. In [Jäckle et al., 2013] it was furthermore investigated how trajectories can be directly extracted from the profiles of Twitter users. By this means, one can visually analyze global and local movement patterns at the same time. Such visualizations can help to further advance research in areas like disease control and human migration. In these cases, the necessary data is currently extracted from sparser and less accessible sources, such as air traffic records or census information.

Finally, instead of just analyzing current events, social media can also be used to predict future trends and developments. In [Krüger et al., 2013a; Lu et al., 2014], visual analytics systems for predicting the revenue of box office movies were presented. These systems also feature text aggregation techniques and interactive model steering. Based on extracting information out of Youtube, Twitter, and IMDB[1], analysts are enabled to examine textual comments, sentiments, and the development of a movie's popularity. Measures like re-tweet numbers and Youtube trailer views can be used as features to train various regression models. The analysts' own experience and background knowledge are an integral part of this process. They can decide which features might be relevant for the current prediction and how the current reception in social media has to

[1] `http://www.imdb.com/`

be assessed. They can select features, parameters, and models accordingly and explore prediction results based on different configurations.

8.4 Future Work

The evaluations of *ScatterBlogs* showed that the system and the underlying techniques are well-received by experts, and that they are almost ready to be employed in real-world situation assessment. However, there were also several minor issues as well as significant larger challenges that were noted in the evaluations as well as in the general course of this thesis. Open challenges encountered with *ScatterBlogs* included problems to perceive relevant labels in the *TagMap*, even with activated *idd* highlighting. The system is furthermore missing more sophisticated means to validate information, it does not adhere to conventions and symbols from existing SA environments, and has only limited means to incorporate media directly within the analysis. The study participants of the *TreeQueST* evaluation highlighted additional needs for local query exploration and recommended improvements in the tag cloud layout. While the use of convention and the reduction of visualized items in the *TagMap* can easily be implemented, other requirements pose more significant challenges in future efforts.

To some degree, the trustworthiness of event-related information can be verified by comparing accounts from multiple users in proximity to the location of the event. This kind of assessment is automatically supported by aggregated visualizations like the *TagMap*. However, relevant accounts of single users are not so easily verified. First takes on automated creditability analysis were proposed by various researchers [Castillo et al., 2011; Gupta et al., 2012; Derczynski and Bontcheva, 2014]. Based on the message timeline of users, their topics of interest, their relationships to other users, their reception within the network, as well as their location history, statistical models can be created that compute a trustworthiness score for authors and their messages. However, this kind of research also touches sensitive aspects of user privacy and poses the danger of ignoring information due to false negatives. The system should thus only rank the messages according to possible relevance. The final decision to consider or ignore it should be made by the analyst.

To incorporate attached videos and photos within the automated analysis process, one also has to consider existing research in the areas of computer vision and image similarity. Past events have shown that people often provide similar images of ongoing critical situations. For example, after a thunderstorm that happened in June 2014 in the north-western part of Germany, several users

posted photographs of fallen trees and damaged cars in Twitter.[2] As these images are related in terms of dominating colors, structures, and content, they might provide a different means to detect event-related information based on image recognition. Of course, this kind of media can be of particular value to disaster managers, journalists, and insurance analysts because it facilitates more reliable assessment of the actual severity of remote events. First takes in this direction have been shown by Kisilevich et al. [2010], who use density-based clustering to find event-related images in Flickr.

In addition to these challenges, more research is also needed in information diffusion. In this thesis, information about social network links and bi-directional communication behavior was not considered. The primary reasons were again questions of privacy and the constrained accessibility to such data, which was discussed in Section 2.2. However, related works from Cao et al. [2012] and Wu et al. [2014] made considerable progress regarding these research questions by extracting and visualizing re-tweet information from live Twitter data. They investigate how specific information is distributed in the spatial and temporal dimensions. This is also an important question for disaster managers and other authorities, particularly if they distribute own warnings and public information. Here they could assess how the public reacts to their recommendations, how they are disseminated, and how they need to be adapted to reach more audiences.

Gaining holistic situation awareness from web data was examined by Bosch et al. [2012]. Instead of just addressing social media, this research asks how structured as well as unstructured information from other web sources can be additionally integrated. In future visual analytics systems, analysts might combine information from social media event observers with recent data from ubiquitous weather stations, traffic sensors, or even public web-cams in layered interactive overviews.

[2] See `http://www.spiegel.de/panorama/gesellschaft/unwetter-in-nrw-das-gewitter-auf-twitter-a-974282.html`

Bibliography

F. Abel, C. Hauff, G. Houben, R. Stronkman, and K. Tao. Twitcident: Fighting fire with information from social web streams. In *World Wide Web Conference (WWW)*, pages 305–308. ACM, 2012. 42

S. Ahern, M. Naaman, R. Nair, and J. H. Yang. World Explorer: Visualizing aggregate data from unstructured text in geo-referenced collections. In *ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 1–10. ACM, 2007. 78

W. Aigner, S. Miksch, W. Müller, H. Schumann, and C. Tominski. Visual methods for analyzing time-oriented data. *IEEE Transactions on Visualization and Computer Graphics*, 14(1):47–60, 2008. 76

A. D. Andre, C. D. Wickens, and L. Moorman. Display formatting techniques for improving situation awareness in the aircraft cockpit. *The International Journal of Aviation Psychology*, 1(3):205–218, 1991. 33

G. L. Andrienko, N. V. Andrienko, C. Hurter, S. Rinzivillo, and S. Wrobel. From movement tracks through events to places: Extracting and characterizing significant places from mobility data. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 161–170. IEEE Computer Society, 2011. 75

G. L. Andrienko, N. V. Andrienko, H. Bosch, T. Ertl, G. Fuchs, P. Jankowski, and D. Thom. Thematic patterns in georeferenced tweets through space-time visual analytics. *Computing in Science and Engineering*, 15(3):72–82, 2013. 8

N. V. Andrienko and G. L. Andrienko. Spatial generalization and aggregation of massive movement data. *IEEE Transactions on Visualization and Computer Graphics*, 17(2):205–219, 2011. 75

I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, and C. D. Spyropoulos. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 160–167. ACM, 2000. 110

E. Aramaki, S. Maskawa, and M. Morita. Twitter catches the flu: Detecting influenza epidemics using twitter. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1568–1576. ACL, 2011. 110

S. Asur and B. A. Huberman. Predicting the future with social media. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 492–499. IEEE Computer Society, 2010. 41

B. B. Bederson, B. Shneiderman, and M. Wattenberg. Ordered and quantum treemaps: Making effective use of 2d space to display hierarchies. *ACM Transactions on Graphics*, 21(4):833–854, 2002. 53

M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, C. Sieb, K. Thiel, and B. Wiswedel. KNIME: The konstanz information miner. In *Conference of the Gesellschaft für Klassifikation*, Studies in Classification, Data Analysis, and Knowledge, pages 319–326. Springer, 2007. 111

E. A. Bier, M. C. Stone, K. A. Pier, K. P. Fishkin, T. Baudel, M. Conway, W. Buxton, and T. DeRose. Toolglass and magic lenses: The see-through interface. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 445–446. ACM, 1994. 18

R. Blanch and E. Lecolinet. Browsing zoomable treemaps: Structure-aware multi-scale navigation techniques. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1248–1253, 2007. 12, 53

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003. 133

H. Bosch, D. Thom, and T. Ertl. Das Web als personalisierte Entscheidungsplattform: Die PESCaDO Idee. In *GI Informatik 2011: Informatik schafft Communities*, volume P-192 of *Lecture Notes in Informatics (LNI)*, page 256. Springer, 2011a. 8

H. Bosch, D. Thom, M. Wörner, S. Koch, E. Puttmann, D. Jackle, and T. Ertl. ScatterBlogs: Geo-spatial document analysis. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 309–310. IEEE Computer Society, 2011b. 71, 116, 125

H. Bosch, D. Thom, G.-A. Heinze, S. Wokusch, and T. Ertl. Dynamic ontology supported user interface for personalized decision support. In *Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC)*, pages 101–107. IARIA XPS Press, 2012. 8, 171

H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Krüger, M. Wörner, and T. Ertl. ScatterBlogs2: Real-time monitoring of microblog messages through user-guided filtering. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2022–2031, 2013.

M. Bruls, K. Huizing, and J. van Wijk. Squarified treemaps. In *Joint Eurographics and IEEE TCVG Symposium on Visualization*, pages 33–42. IEEE Computer Society, 2000. 53

C. Brunk, J. Kelly, and R. Kohavi. Mineset: An integrated system for data mining. In *AAAI Conference on Knowledge Discovery and Data Mining (KDD)*, pages 135–138. AAAI Press, 1997. 15

N. Cao, Y. Lin, X. Sun, D. Lazer, S. Liu, and H. Qu. Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2649–2658, 2012. 42, 171

S. K. Card, J. D. Mackinlay, and B. Shneiderman. *Readings in information visualization - using vision to think*. Morgan Kaufmann, Burlington, Mass., USA, 1999. 11, 17

D. A. Carr. Guidelines for designing information visualization applications. In *Ericsson Conference on Usability Engineering ECUE*, pages 1–7, 1999. 18

G. Casella and E. I. George. Explaining the Gibbs sampler. *The American Statistician*, 46(3):167–174, 1992. 134

C. Castillo, M. Mendoza, and B. Poblete. Information credibility on Twitter. In *World Wide Web Conference (WWW)*, pages 675–684. ACM, 2011. 170

J. Chae, D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D. S. Ebert, and T. Ertl. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 143–152. IEEE Computer Society, 2012. 8, 134

J. Chae, D. Thom, Y. Jang, S. Y. Kim, T. Ertl, and D. S. Ebert. Visual analytics of microblog data for public behavior analysis in disaster events. In *EuroVis workshop on visual analytics (EuroVA)*, pages 67–71. Eurographics Association, 2013. 8

J. Chae, D. Thom, Y. Jang, S. Kim, T. Ertl, and D. S. Ebert. Public behavior response analysis in disaster events utilizing visual analytics of microblog data. *Computers & Graphics*, 38:51–60, 2014. 8

C. Chew and G. Eysenbach. Pandemics in the age of twitter: Content analysis of tweets during the 2009 H1N1 outbreak. *PLoS ONE*, 5(11):e14118, 11 2010. 34

D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey. Scatter/Gather: A cluster-based approach to browsing large document collections. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 318–329. ACM, 1992. 47, 48, 51

D. R. Cutting, D. R. Karger, and J. O. Pedersen. Constant interaction-time Scatter/Gather browsing of very large document collections. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 126–134. ACM, 1993. 51

L. Derczynski and K. Bontcheva. PHEME: Veracity in digital social networks. In *Conference on User Modeling, Adaptation, and Personalization*, volume 1181 of *CEUR Workshop Proceedings*, pages 1–4. CEUR-WS.org, 2014. 170

DHS. Using social media for enhanced situational awareness and decision support. Technical report, DHS Virtual Social Media Working Group and DHS First Responders Group, June 2014a. 34, 35, 36

DHS. Lessons learned: Social media and hurricane Sandy. Technical report, DHS Virtual Social Media Working Group and DHS First Responders Group, June 2014b. 35, 36

I. Dilrukshi, K. De Zoysa, and A. Caldera. Twitter news classification using SVM. In *International Conference on Computer Science Education (ICCSE)*, pages 287–291. IEEE Press, April 2013. 110

M. Dörk, M. S. T. Carpendale, C. Collins, and C. Williamson. VisGets: Coordinated visualizations for web-based information exploration and discovery. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1205–1212, 2008. 111

W. Dou, X. Wang, D. Skau, W. Ribarsky, and M. X. Zhou. LeadLine: Interactive visual analysis of text data through event identification and exploration. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 93–102. IEEE Computer Society, 2012. 42, 77, 134

W. Dou, L. Yu, X. Wang, Z. Ma, and W. Ribarsky. HierarchicalTopics: Visually exploring large text collections using topic hierarchies. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2002–2011, 2013. 42, 54, 134

R. Eccles, T. Kapler, R. Harper, and W. Wright. Stories in GeoTime. *Information Visualization*, 7(1):3–17, 2008. 75

N. Elmqvist, J. T. Stasko, and P. Tsigas. DataMeadow: A visual canvas for analysis of large-scale multivariate data. *Information Visualization*, 7(1):18–33, 2008. 111

A. Endert, M. S. Hossain, N. Ramakrishnan, C. North, P. Fiaux, and C. Andrews. The human is the loop: New directions for visual analytics. *Journal of Intelligent Information Systems*, 43(3):411–435, 2014. 17

M. R. Endsley. Design and evaluation for situation awareness enhancement. In *Human Factors and Ergonomics Society Annual Meeting*, volume 32, pages 97–101. SAGE, 1988. 31

M. R. Endsley. Toward a theory of situation awareness in dynamic systems. *SAGE Journal of the Human Factors and Ergonomics Society*, 37(1):32–64, 1995. 31

R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. 113

R. A. Finkel and J. L. Bentley. Quad Trees: A data structure for retrieval on composite keys. *Acta Informatica*, 4:1–9, 1974. 99

K. Florek, J. Łukaszewicz, J. Perkal, H. Steinhaus, and S. Zubrzycki. Sur la liaison et la division des points d'un ensemble fini. In *Colloquium Mathematicae*, volume 2, pages 282–285. Institute of Mathematics, Polish Academy of Sciences, 1951. 54

E. Frank, M. A. Hall, G. Holmes, R. Kirkby, B. Pfahringer, I. H. Witten, and L. Trigg. Weka: A machine learning workbench for data mining. In O. Maimon and L. Rokach, editors, *Data Mining and Knowledge Discovery Handbook, 2nd ed.*, pages 1269–1277. Springer, Berlin, Heidelberg, Germany, 2010. 55

G. W. Furnas. Generalized fisheye views. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 16–23. ACM, 1986. 18

J. Ghosh and A. Strehl. Similarity-based text clustering: A comparative study. In J. Kogan, C. K. Nicholas, and M. Teboulle, editors, *Grouping Multidimensional Data - Recent Advances in Clustering*, pages 73–97. Springer, Berlin, Heidelberg, Germany, 2006. 55

A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. Technical report, Stanford Digital Library, 2009. 41, 110

X. Gong, W. Ke, and R. Khare. Studying Scatter/Gather browsing for web search. *Proceedings of the American Society for Information Science and Technology*, 49(1): 1–4, 2012. 52

G. Grinstein, K. Cook, P. Havig, K. Liggett, B. Nebesh, M. Whiting, K. Whitley, and S. Knoecni. Vast 2011 challenge: cyber security and epidemic. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 299–301. IEEE Computer Society, 2011. 71

M. Gupta, P. Zhao, and J. Han. Evaluating event credibility on twitter. In *SIAM International Conference on Data Mining*, pages 153–164. SIAM / Omnipress, 2012. 170

V. Harinarayan, A. Rajaraman, and J. D. Ullman. Implementing data cubes efficiently. In *ACM SIGMOD Conference on Management of Data*, pages 205–216. ACM, 1996. 127

M. Hassenzahl, M. Burmester, and F. Koller. AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität. In J. Ziegler and G. Szwillus, editors, *Mensch & Computer 2003: Interatkion in Bewegung*, pages 187–196. Springer, Berlin, Heidelberg, Germany, 2003. 154

M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 76–84. ACM, 1996. 52

J. Heer and D. Boyd. Vizster: Visualizing online social networks. In *IEEE Information Visualization Conference (InfoVis)*, page 5. IEEE Computer Society, 2005. 12

P. Heim, D. Thom, and T. Ertl. Semsor: Combining social and semantic web to support the analysis of emergency situations. In *International Workshop on Semantic Models for Adaptive Interactive Systems (SEMAIS) (co-located with IUI 2011)*, pages 1–5, 2011, Online Proceedings. 8

F. Heimerl, S. Koch, H. Bosch, and T. Ertl. Visual classifier training for text document retrieval. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2839–2848, 2012. 112, 114

T. Heverin and L. Zach. Microblogging for crisis communication: Examination of Twitter use in response to a 2009 violent crisis in Seattle-Tacoma, Washington Area. In *Information Systems for Crisis Response and Management Conference (ISCRAM)*, pages 1–5. ISCRAM Association, 2010. 34

B. Höferlin, R. Netzel, M. Höferlin, D. Weiskopf, and G. Heidemann. Inter-active learning of ad-hoc classifiers for video visual analytics. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 23–32. IEEE Computer Society, 2012. 111

A. Hughes and L. Palen. Twitter adoption and use in mass convergence and emergency events. *International Journal of Emergency Management*, 6(3):246–260, 2009. 34

C. Hurter, A. Telea, and O. Ersoy. MoleView: An attribute and structure-based semantic lens for large element-based plots. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2600–2609, 2011. 18

IBM, P. Zikopoulos, and C. Eaton. *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data*. McGraw-Hill Osborne Media, New York, NY, USA, 1st edition, 2011. 4

Instagram. Press release. `http://instagram.com/press/`, 2014, [Online; accessed 30-October-2014]. 26

D. Jäckle, H. Bosch, D. Thom, R. Krüger, D. A. Keim, and T. Ertl. Visual analysis of social media data in emergency situations by aggregating annotated user movements (Poster). In *Information Systems for Crisis Response and Management Conference (ISCRAM)*, 2013. 8, 169

A. Jaffe, M. Naaman, T. Tassa, and M. Davis. Generating summaries and visualization for large collections of geo-referenced photographs. In *ACM SIGMM Workshop on Multimedia Information Retrieval (MIR)*, pages 89–98. ACM, 2006. 76

A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999. 20

B. Johnson and B. Shneiderman. Tree maps: A space-filling approach to the visualization of hierarchical information structures. In *IEEE Conference on Visualization (Vis)*, pages 284–291. IEEE Computer Society, 1991. 12, 53

D. G. Jones and M. R. Endsley. Sources of situation awareness errors in aviation. *Aviation, Space, and Environmental Medicine*, 67(6):507–512, 1996. 33

K. S. Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972. 94

D. Jonker, W. Wright, D. Schroh, P. Proulx, B. Cort, et al. Information triage with TRIST. In *2005 Intelligence Analysis Conference*, pages 1–6, 2005. 17

M. Joshi, D. Das, K. Gimpel, and N. A. Smith. Movie reviews and revenues: An experiment in text regression. In *Human Language Technologies Conference of the ACL North American Chapter (HLT)*, pages 293–296. ACL, 2010. 19

W. Ke, C. R. Sugimoto, and J. Mostafa. Dynamicity vs. effectiveness: Studying online clustering for Scatter/Gather. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 19–26. ACM, 2009. 52

D. Keim, G. L. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, and G. Melançon. Visual analytics: Definition, process, and challenges. In A. Kerren, J. T. Stasko, J.-D. Fekete, and C. North, editors, *Information Visualization*, pages 154–175. Springer, Berlin, Heidelberg, Germany, 2008. 13, 14

D. A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002. 9, 15

D. A. Keim, J. Kohlhammer, G. P. Ellis, and F. Mansmann. *Mastering the Information Age - Solving Problems with Visual Analytics*. Eurographics Association, Goslar, 2010. 13, 14, 75, 167

A. Kerren and F. Schreiber. Toward the role of interaction in visual analytics. In *Winter Simulation Conference (WSC)*, pages 1–13. WSC, 2012. 16

S. Kisilevich, M. Krstajic, D. A. Keim, N. V. Andrienko, and G. L. Andrienko. Event-based analysis of people's activities and behavior using Flickr and Panoramio geotagged photo collections. In *IEEE Information Visualization Conference (InfoVis)*, pages 289–296. IEEE Computer Society, 2010. 171

G. Klein, B. M. Moon, and R. R. Hoffman. Making sense of sensemaking 1: Alternative perspectives. *IEEE Intelligent Systems*, 21(4):70–73, 2006. 32

S. Klenk, D. Thom, and G. Heidemann. The normalized compression distance as a distance measure in entity identification. In *Advances in Data Mining. Applications and Theoretical Aspects*, pages 325–337. Springer, Berlin, Heidelberg, 2009. 8

S. Koch, H. Bosch, M. Giereth, and T. Ertl. Iterative integration of visual insights during scalable patent search and analysis. *IEEE Transactions on Visualization and Computer Graphics*, 17(5):557–569, 2011. 17, 116

S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica (Slovenia)*, 31(3):249–268, 2007. 19

M. Kreuseler and H. Schumann. A flexible approach for visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):39–51, 2002. 9

R. Krikorian. New tweets per second record, and how! https://blog.twitter.com/2013/new-tweets-per-second-record-and-how, August 2013, [Online; accessed 30-October-2014]. 29

R. Krüger, H. Bosch, S. Koch, C. Müller, G. Reina, D. Thom, and T. Ertl. HIVEBEAT - A highly interactive visualization environment for broad-scale exploratory analysis and tracing. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, VAST Challenge, pages 277–278. IEEE Computer Society, 2012a. 8

R. Krüger, S. Lohmann, D. Thom, H. Bosch, and T. Ertl. Using social media content in the visual analysis of movement data. In *Workshop on Interactive Visual Text Analytics (co-located with IEEE VisWeek 2012)*, VisWeek USB Proceedings, pages 1–3, 2012b. 8, 169

R. Krüger, H. Bosch, D. Thom, E. Püttmann, Q. Han, S. Koch, F. Heimerl, and T. Ertl. Prolix - Visual prediction analysis for box office success. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, VAST Challenge, pages 1–2. IEEE VAST USB Proceedings, 2013a. 8, 169

R. Krüger, D. Thom, M. Wörner, H. Bosch, and T. Ertl. Trajectorylenses - A set-based filtering and exploration technique for long-term trajectory data. *Computer Graphics Forum*, 32(3):451–460, 2013b. 8

R. Krüger, D. Thom, and T. Ertl. Visual analysis of movement behavior using web data for context enrichment. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 193–200. IEEE Computer Society, 2014a. 8, 169

R. Krüger, D. Thom, and T. Ertl. Semantic enrichment of movement behavior with foursquare - A visual analytics approach. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–14, 2014b, to appear. 8, 169

S. Laskowski and C. Plaisant. Evaluation methodologies for visual analytics. In J. Thomas and K. Cook, editors, *Illuminating the Path, the Research and Development Agenda for Visual Analytics*, pages 150–157. IEEE Computer Society, Hoboken, NJ, USA, 2005. 141

R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 22(140):1–55, 1932. 35, 159

L. D. Lins, J. T. Klosowski, and C. E. Scheidegger. Nanocubes for real-time exploration of spatiotemporal datasets. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2456–2465, 2013. 127

Z. Liu, B. Jiang, and J. Heer. imMens: Real-time visual querying of big data. *Computer Graphics Forum*, 32(3):421–430, 2013. 127

S. P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–136, 1982. 20

S. Lohmann, J. Ziegler, and L. Tetzlaff. Comparison of tag cloud layouts: Task-related performance and visual exploration. In *International Conference on Human-Computer Interaction (INTERACT)*, volume 5726 of *Lecture Notes in Computer Science (LNCS)*, pages 392–404. Springer, 2009. 87, 132

Y. Lu, R. Krüger, D. Thom, F. Wang, S. Koch, T. Ertl, and R. Maciejewski. Integrating predictive analytics and social media. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, USB proceedings paper, pages 1–10. IEEE Computer Society, 2014. 8, 169

M. Luboschik, H. Schumann, and H. Cords. Particle-based labeling: Fast point-feature labeling without obscuring other visual features. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1237–1244, 2008. 87

A. M. MacEachren, A. R. Jaiswal, A. C. Robinson, S. Pezanowski, A. Savelyev, P. Mitra, X. Zhang, and J. Blanford. SensePlace2: Geotwitter analytics support for situational awareness. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 181–190. IEEE Computer Society, 2011. 34, 36, 42

R. Maciejewski, S. Rudolph, R. Hafen, A. M. Abusalah, M. Yakout, M. Ouzzani, W. S. Cleveland, S. J. Grannis, and D. S. Ebert. A visual analytics approach to understanding spatiotemporal hotspots. *IEEE Transactions on Visualization and Computer Graphics*, 16(2):205–220, 2010. 96

C. D. Manning, P. Raghavan, and H. Schuetze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008. 94, 104

A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller. Twitinfo: Aggregating and visualizing microblogs for event exploration. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 227–236. ACM, 2011. 42, 77

C. Matheus, M. Kokar, and K. Baclawski. A core ontology for situation awareness. In *International Conference on Information Fusion (FUSION)*, volume 1, pages 545–552. IEEE Computer Society, 2003. 69

E. J. McCluskey. Minimization of Boolean functions. *Bell System Technical Journal*, 35(6):1417–1444, 1956. 61

P. C. McGeer, J. V. Sanghavi, R. K. Brayton, and A. L. Sangiovanni-Vincentelli. ESPRESSO-SIGNATURE: a new exact minimizer for logic functions. *IEEE Transactions on Very Large Scale Integration Systems*, 1(4):432–440, 1993. 61

M. Mendoza, B. Poblete, and C. Castillo. Twitter under crisis: Can we trust what we rt? In *Workshop on Social Media Analytics (SOMA)*, pages 71–79. ACM, 2010. 34

Z. Ming, C. Luo, W. Gao, R. Han, Q. Yang, L. Wang, and J. Zhan. BDGS: A scalable big data generator suite in big data benchmarking. In *Workshop Series on Big Data Benchmarking (WBDB)*, volume 8585 of *Lecture Notes in Computer Science (LNCS)*, pages 138–154. Springer, 2013. 4

S. Mittelstädt, D. Spretke, D. Thom, D. Jäckle, A. Karsten, and D. Keim. Situational awareness for critical infrastructures and decision support. In *IST-116 Symposium on Visual Analytics*. NATO Science and Technology Organization (STO), 2013. 8

S. Mittelstädt, X. Wang, T. Eaglin, D. Thom, D. Keim, W. Tolone, and W. Ribarsky. An integrated in-situ approach to impacts from natural disasters on critical infrastructures. In *Hawaii International Conference on System Sciences (HICSS)*. IEEE Computer Society, 2015, to appear. 8

J. Moehrmann and G. Heidemann. Efficient annotation of image data sets for computer vision applications. In *International Workshop on Visual Interfaces for Ground Truth Collection in Computer Vision Applications (VIGTA)*, pages 2:1–2:6. ACM, 2012. 111

A. B. M. Moniruzzaman and S. A. Hossain. NoSQL database: New era of databases for big data analytics - classification, characteristics and comparison. *International Journal of Database Theory and Application*, 6(4):1–14, 2013. 26

F. Morstatter, J. Pfeffer, H. Liu, and K. M. Carley. Is the sample good enough? comparing data from Twitter's streaming api with Twitter's firehose. In *AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 400–408. AAAI, 2013. 30

C. North. Toward measuring visualization insight. *IEEE Computer Graphics and Applications*, 26(3):6–9, 2006. 140

O. Okolloh. Ushahidi, or 'testimony': Web 2.0 tools for crowdsourcing crisis information. *Participatory Learning and Action*, 59(1):65–70, 2009. 34

T. O'Reilly. *What is Web 2.0*. O'Reilly Media, Newton, MAS, USA, 2009. 21

S. O. Oyeyemi, E. Gabarron, and R. Wynn. Ebola, Twitter, and misinformation: A dangerous combination? *BMJ*, 349:1–2, 2014. 27

A. Panagiotidis, H. Bosch, S. Koch, and T. Ertl. EdgeAnalyzer: Exploratory analysis through advanced edge interaction. In *Hawaii International International Conference on Systems Science (HICSS)*, pages 1–10. IEEE Computer Society, 2011. 18

E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962. 95

D. Pelleg and A. W. Moore. X-Means: Extending K-Means with efficient estimation of the number of clusters. In *International Conference on Machine Learning (ICML)*, pages 727–734. Morgan Kaufmann, 2000. 20, 81

P. Pirolli and S. Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *2005 Intelligence Analysis Conference*, volume 5, pages 2–4. Mitre McLean, VA, 2005. 9, 17

P. Pirolli, P. K. Schank, M. A. Hearst, and C. Diehl. Scatter/Gather browsing communicates the topic structure of a very large text collection. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 213–220. ACM, 1996. 52

C. Plaisant. The challenge of information visualization evaluation. In *International Working Conference on Advanced Visual Interfaces (AVI)*, pages 109–116. ACM, 2004. 140, 142

C. Plaisant, J. Fekete, and G. G. Grinstein. Promoting insight-based evaluation of visualizations: From contest to benchmark repository. *IEEE Transactions on Visualization and Computer Graphics*, 14(1):120–134, 2008. 141

A. Pozdnoukhov and C. Kaiser. Space-time dynamics of topics in streaming text. In *International Workshop on Location Based Social Networks (LBSN)*, pages 1–8. ACM, 2011. 134

Y. Qu, C. Huang, P. Zhang, and J. Zhang. Microblogging after a major disaster in china: A case study of the 2010 Yushu Earthquake. In *ACM Conference on Computer Supported Cooperative Work (CSCW)*, pages 25–34. ACM, 2011. 34

D. Quercia, H. Askham, and J. Crowcroft. TweetLDA: Supervised topic classification and link prediction in Twitter. In *Web Science Conference (WebSci)*, pages 247–250. ACM, 2012. 134

W. V. Quine. The problem of simplifying truth functions. *The American Mathematical Monthly*, 59(8):521–531, 1952. 61

D. Ramage, S. T. Dumais, and D. J. Liebling. Characterizing microblogs with topic models. In *AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 130–137. AAAI Press, 2010. 134

Rapid Earthquake Viewer. Virginia Earthquake. `http://rev.seis.sc.edu/earthquakes/2011/08/23/17/51/03`, 2011, [Online; accessed 30-October-2014]. 27

M. D. Rodgers, R. H. Mogford, and B. Strauch. Post hoc assessment of situation awareness in air traffic control incidents and major aircraft accidents. In M. R. Endsley and D. J. Garland, editors, *Situation Awareness Analysis and Measurement*, pages 64–100. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 2000. 33

S. Roller, M. Speriosu, S. Rallapalli, B. Wing, and J. Baldridge. Supervised text-based geolocation using language models on an adaptive grid. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1500–1510. ACL, 2012. 78

M. Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, 1956. 95

H. a. Rosling. Visual technology unveils the beauty of statistics and swaps policy from dissemination to access. *Statistical Journal of the IAOS: Journal of the International Association for Official Statistics*, 24(1):103–104, 2007. 12

P. Runeson. A survey of unit testing practices. *IEEE Software*, 23(4):22–29, 2006. 102

D. Sacha, A. Stoffel, F. Stoffel, B. C. Kwon, G. P. Ellis, and D. A. Keim. Knowledge generation model for visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1604–1613, 2014. 15

T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: Real-time event detection by social sensors. In *World Wide Web Conference (WWW)*, pages 851–860. ACM, 2010. 25, 34, 76

S. Salvador and P. Chan. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In *IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 576–584. IEEE Computer Society, 2004. 57

M. Sarkar and M. H. Brown. Graphical fisheye views of graphs. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 83–91. ACM, 1992. 18

J. Scholtz, M. A. Whiting, C. Plaisant, and G. Grinstein. A reflection on seven years of the VAST challenge. In *BELIV Workshop: Beyond Time and Errors - Novel Evaluation Methods for Visualization*, pages 13:1–13:8. ACM, 2012. 70

W. J. Schroeder, L. S. Avila, and W. Hoffman. Visualizing with VTK: A tutorial. *IEEE Computer Graphics and Applications*, 20(5):20–27, 2000. 111

H. Schuerig. Social Media statt Web 2.0. `http://www.henningschuerig.de/2010/social-media-statt-web-20/`, 2014, [Online; accessed 30-October-2014]. 21

A. Schulz, P. Ristoski, and H. Paulheim. I see a car crash: Real-time detection of small scale incidents in microblogs. In *ESWC 2013 Satellite Events - The Semantic Web*, volume 7955 of *Lecture Notes in Computer Science (LNCS)*, pages 22–33. Springer, 2013. 76

C. Seifert, V. Sabol, and M. Granitzer. Classifier hypothesis generation using visual analysis methods. In *International Conference on Networked Digital Technologies*, volume 87 of *Communications in Computer and Information Science*, pages 98–111. Springer, 2010. 111

B. Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. 19, 112

B. Shneiderman. Dynamic queries for visual information seeking. *IEEE Software*, 11(6):70–77, 1994. 111

B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *IEEE Symposium on Visual Languages*, pages 336–343. IEEE Computer Society, 1996. 12

B. Shneiderman. Inventing discovery tools: Combining information visualization with data mining. In *International Conference on Discovery Science*, volume 2226 of *Lecture Notes in Computer Science (LNCS)*, pages 17–28. Springer, 2001. 9

Y. B. Shrinivasan and J. J. van Wijk. Supporting the analytical reasoning process in information visualization. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1237–1246. ACM, 2008. 17

R. W. Sinnott. Virtues of the haversine. *Sky and Telescope*, 68(2):158–159, 1984. 97

SKK. *Empfehlungen für taktische Zeichen im Bevölkerungsschutz*. Ständige Konferenz für Katastrophenvorsorge und Bevölkerungsschutz, Bonn, Germany, 2010. 153

A. Slingsby, J. Dykes, J. Wood, and K. Clarke. Interactive tag maps and tag clouds for the multiscale exploration of large spatio-temporal datasets. In *IEEE Information Visualization Conference (InfoVis)*, pages 497–504. IEEE Computer Society, 2007. 76

M. Smuc, E. Mayr, T. Lammarsch, W. Aigner, S. Miksch, and J. Gärtner. To score or not to score? Tripling insights for participatory design. *IEEE Computer Graphics and Applications*, 29(3):29–38, 2009. 141

B. Solis and JESS3. The conversation prism. `http://www.conversationprism.com/`, 2014, [Online; accessed 30-October-2014]. 21, 22

K. Starbird and L. Palen. (How) will the revolution be retweeted?: Information diffusion and the 2011 Egyptian uprising. In *ACM Conference on Computer Supported Cooperative Work (CSCW)*, pages 7–16. ACM, 2012. 34

K. Starbird, J. Maddock, M. Orand, P. Achterman, and R. M. Mason. Rumors, false flags, and digital vigilantes: Misinformation on Twitter after the 2013 Boston Marathon Bombing. In *iConference*, pages 654–662. iSchools, 2014. 27

J. T. Stasko, C. Görg, and Z. Liu. Jigsaw: Supporting investigative analysis through interactive visualization. *Information Visualization*, 7(2):118–132, 2008. 18

J. Steele and N. Iliinsky. *Beautiful Visualization*. O'Reilly Media, Inc., Newton, Mass., USA, 2010. 59

M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas. Sentiment in short strength detection informal text. *Journal of the Association for Information Science and Technology (JASIST)*, 61(12):2544–2558, 2010. 60, 126

D. Thom and T. Ertl. TreeQueST: A treemap-based query sandbox for microdocument retrieval. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1714–1723. IEEE Computer Society, 2015.

D. Thom, H. Bosch, and T. Ertl. Inverse document density: A smooth measure for location-dependent term irregularities. In *International Conference on Computational Linguistics COLING*, pages 2603–2618. Indian Institute of Technology Bombay, 2012a.

D. Thom, H. Bosch, S. Koch, M. Wörner, and T. Ertl. Spatiotemporal anomaly detection through visual analysis of geolocated Twitter messages. In *IEEE Pacific Visualization Symposium (PacificVis)*, pages 41–48. IEEE Computer Society, 2012b.

D. Thom, H. Bosch, R. Krüger, and T. Ertl. Using large scale aggregated knowledge for social media location discovery. In *Hawaii International Conference on System Sciences (HICSS)*, pages 1464–1473. IEEE Computer Society, 2014a. 130

D. Thom, M. Wörner, and S. Koch. Scatterscopes: Understanding events in real-time through spatiotemporal indication and hierarchical drilldown. In *IEEE Conference on Visual Analytics Science and Technology (VAST)*, VAST Challenge, pages 1–2. IEEE VAST USB Proceedings, 2014b. 135

D. Thom, R. Krüger, T. Ertl, U. Bechstedt, A. Platz, J. Zisgen, and B. Volland. Can Twitter really save your life? A broad-scale expert study of visual social media analytics for situation awareness. In *IEEE Pacific Visualization Symposium (PacificVis)*. IEEE Computer Society, 2015, to appear.

J. J. Thomas and K. A. Cook. *Illuminating the path: The research and development agenda for visual analytics*. IEEE Computer Society, Hoboken, NJ, USA, 2005. 9, 12, 75, 166

S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66, 2001. 113

M. Trutschl, G. G. Grinstein, and U. Cvek. Intelligently resolving point occlusion. In *IEEE Information Visualization Conference (InfoVis)*, pages 131–136. IEEE Computer Society, 2003. 15

O. Turetken and R. Sharda. Development of a fisheye-based information search processing aid (FISPA) for managing information overload in the web environment. *Decision Support Systems*, 37(3):415–434, 2004. 53

Twitter. Form s-1 financial statement. `http://www.sec.gov/Archives/edgar/data/1418091/000119312513390321/d564001ds1.htm`, 2014, [Online; accessed 30-October-2014]. 26, 27, 29

J. van Wijk. The value of visualization. In *IEEE Conference on Visualization (Vis)*, pages 79–86. IEEE Computer Society, 2005. 110

S. Vieweg, L. Palen, S. B. Liu, A. L. Hughes, and J. Sutton. Collective intelligence in disaster: An examination of the phenomenon in the aftermath of the 2007 Virginia Tech shootings. In *Information Systems for Crisis Response and Management Conference (ISCRAM)*, pages 44–54. ISCRAM, 2008. 34

S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen. Microblogging during two natural hazards events: What Twitter may contribute to situational awareness. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1079–1088. ACM, 2010. 33

J. Wagner. Two great social data platforms: How Datasift and Gnip stack up. `http://www.programmableweb.com/news/two-great-social-data-platforms-how-datasift-and-gnip-stack/brief/2014/02/10`, 2014, [Online; accessed 30-October-2014]. 38

L. Wang, J. Zhan, C. Luo, Y. Zhu, Q. Yang, Y. He, W. Gao, Z. Jia, Y. Shi, S. Zhang, C. Zheng, G. Lu, K. Zhan, X. Li, and B. Qiu. Bigdatabench: A big data benchmark suite from internet services. *CoRR*, abs/1401.1406:1–12, 2014. 4

F. Wanner, A. Stoffel, D. Jäckle, B. C. Kwon, A. Weiler, and D. A. Keim. State-of-the-art report of visual analysis for event detection in text data streams. In *Eurographics Conference on Visualization (EuroVis)*, EuroVis - STARs, pages 125–139. Eurographics Association, 2014. 77

J. H. Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58(301):236–244, 1963. 54

We Are Social. Global digital statistics. `http://wearesocial.net/blog/2014/01/social-digital-mobile-worldwide-2014/`, 2014, [Online; accessed 30-October-2014]. 27

J. Weng and B.-S. Lee. Event detection in twitter. In *AAAI Conference on Weblogs and Social Media (ICWSM)*, volume 11, pages 401–408. AAAI Press, 2011. 76

J. Weng, E. Lim, J. Jiang, and Q. He. Twitterrank: Finding topic-sensitive influential twitterers. In *ACM Conference on Web Search and Web Data Mining (WSDM)*, pages 261–270. ACM, 2010. 41

L. A. Westover. *Splatting: A parallel, feed-forward volume rendering algorithm*. PhD thesis, University of North Carolina at Chapel Hill, 1991, UMI Order No. GAX92-08005. 100

J. J. D. White and R. E. Roth. Twitterhitter: Geovisual analytics for harvesting insight from volunteered geographic information. In *International Conference on Geographic Information Science (GIScience)*, 2010. 42

G. J. Wills. An interactive view for hierarchical clustering. In *IEEE Information Visualization Conference (InfoVis)*, pages 26–31. IEEE Computer Society, 1998. 53

B. P. Wing and J. Baldridge. Simple supervised document geolocation with geodesic grids. In *Human Language Technologies Conference of the ACL North American Chapter (HLT)*, volume 1, pages 955–964. ACL, 2011. 78

M. G. Wing, A. Eklund, and L. D. Kellogg. Consumer-grade global positioning system (GPS) accuracy and reliability. *Journal of Forestry*, 103(4):169–173, 2005. 24

L. Wittgenstein. *Philosophical Investigations*. Blackwell Publishers, Oxford, England, UK, 2001. 15

J. Wood, J. Dykes, A. Slingsby, and K. Clarke. Interactive visual exploration of a large spatio-temporal dataset: Reflections on a geovisualization mashup. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1176–1183, 2007. 76

B. Wright, J. Payne, M. Steckman, and S. Stevson. Palantir: A visualization platform for real-world analysis. In *IEEE Symposium on Visual Analytics Science and Technology*, pages 249–250. IEEE Computer Society, 2009. 18

S. Wu, W. Lai, T. Daly, and W. Pentney. Systems and methods for providing a microdocument framework for storage, retrieval, and aggregation, 2011. [Online]. Available: `http://www.google.de/patents/US20110258177`, US Patent App. 12/762,807. 24

Y. Wu, S. Liu, K. Yan, M. Liu, and F. Wu. OpinionFlow: Visual analysis of opinion diffusion on social media. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1763–1772, 2014. 171

Youtube. Platform statistics. `https://www.youtube.com/yt/press/statistics.html`, 2014, [Online; accessed 30-October-2014]. 26

D. Zhang, B. Ooi, and A. Tung. Locating mapped resources in Web 2.0. In *IEEE Conference on Data Engineering (ICDE)*, pages 521–532. IEEE Press, 2010. 78

J. Zhao, N. Cao, Z. Wen, Y. Song, Y.-R. Lin, and C. Collins. #FluxFlow: Visual analysis of anomalous information spreading on social media. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1773–1782, Dec 2014. 42

W. X. Zhao, J. Jiang, J. Weng, J. He, E. Lim, H. Yan, and X. Li. Comparing Twitter and traditional media using topic models. In *European Conference on Advances in Information Retrieval Research (ECIR)*, volume 6611 of *Lecture Notes in Computer Science (LNCS)*, pages 338–349. Springer, 2011. 41, 134

J. Zisgen, J. Kern, D. Thom, and T. Ertl. #Hochwasser - Using visual analytics of social media in civil protection. *i-com: Zeitschrift für interaktive und kooperative Medien*, 13(1):37–44, 2014. 8