Enriching Unstructured Media Content About Events to Enable Semi-Automated Summaries, Compilations, and Improved Search by Leveraging Social Networks

Thomas Steiner

Departament de Llenguatges i Sistemes Informàtics

Universitat Politècnica de Catalunya

A thesis submitted for the degree of

Philosophiæ Doctor (PhD)

February 2014

1^{st} Advisor: Joaquim Gabarró Vallés

Universitat Politècnica de Catalunya, Barcelona, Spain

2^{nd} Advisor: Michael Hausenblas

Digital Enterprise Research Institute, Galway, Ireland

MapR Technologies, San Jose, CA, USA

Abstract

(i) Mobile devices and social networks are omnipresent

Mobile devices such as smartphones, tablets, or digital cameras together with social networks enable people to create, share, and consume enormous amounts of media items like videos or photos both on the road or at home. Such mobile devices—by pure definition—accompany their owners almost wherever they may go. In consequence, mobile devices are omnipresent at all sorts of events to capture noteworthy moments. Exemplary events can be keynote speeches at conferences, music concerts in stadiums, or even natural catastrophes like earthquakes that affect whole areas or countries. At such events—given a stable network connection—part of the event-related media items are published on social networks both as the event happens or afterwards, once a stable network connection has been established again.

(ii) Finding representative media items for an event is hard

Common media item search operations, for example, searching for *the* official video clip for a certain hit record on an online video platform can in the simplest case be achieved based on potentially shallow human-generated metadata or based on more profound content analysis techniques like optical character recognition, automatic speech recognition, or acoustic fingerprinting. More advanced scenarios, however, like retrieving all (or just the most representative) media items that were created at a given event with the objective of creating *event summaries* or *media item compilations* covering the event in question are hard, if not impossible, to fulfill at large scale. The central research question of this thesis can be formulated as follows.

(iii) Central research question

"Can user-customizable media galleries that summarize given events be created solely based on textual and multimedia data from social networks?"

(iv) Core contributions

In the context of this thesis, we have developed and evaluated a novel interactive application and related methods for media item enrichment, leveraging social networks, utilizing the Web of Data, techniques known from Content-based Image Retrieval (CBIR) and Content-based Video Retrieval (CBVR), and fine-grained media item addressing schemes like Media Fragments URIs to provide a scalable and near realtime solution to realize the abovementioned scenario of event summarization and media item compilation.

(v) Methodology

For any event with given event title(s), (potentially vague) event location(s), and (arbitrarily fine-grained) event date(s), our approach can be divided in the following six steps.

1. Via the textual search APIs (Application Programming Interfaces) of different social networks, we retrieve a list of potentially event-relevant microposts that either contain media items directly, or that provide links to media items on external media item hosting platforms.
2. Using third-party Natural Language Processing (NLP) tools, we recognize and disambiguate named entities in microposts to predetermine their relevance.
3. We extract the binary media item data from social networks or media item hosting platforms and relate it to the originating microposts.
4. Using CBIR and CBVR techniques, we first deduplicate exact-duplicate and near-duplicate media items and then cluster similar media items.
5. We rank the deduplicated and clustered list of media items and their related microposts according to well-defined ranking criteria.
6. In order to generate interactive and user-customizable media galleries that visually and audially summarize the event in question, we compile the top-n ranked media items and microposts in aesthetically pleasing and functional ways.

To Laura, Lena, Emma, and Nil.

Acknowledgements

Personal Acknowledgements:

First and foremost, I would like to thank my wife Laura for her support, understanding, patience, and energy during my time as a PhD student, and simply for being at my side. Without you, I would not be where I am today.

I wholeheartedly thank my two advisors Joaquim Gabarró Vallés and Michael Hausenblas for their guidance, helpful comments, informative pointers, and especially for their constructive criticisms. The areas of research that I have tackled in this thesis are still young and sometimes uncharted territory. I am very thankful that the two of you have ventured on the undertaking of leading me through this thesis.

I deeply appreciate all the review comments, thoughts, challenging questions, and, last not least, the LaTeX help of my dear friend and research colleague Ruben Verborgh. It was, is, and will be an honor to work with you. My warm thanks also go to Raphaël Troncy and his team at EURECOM Sophia Antipolis in France who have helped shape some of the ideas presented in this thesis.

I would like to thank my former and current managers at Google, namely N. Kryvossidis, R. Ashley, C. Bouchère, and I. Sassarini for their support for my thesis. A lot of valuable input for my thesis came in via social networks. My thanks go out to everyone I have interacted with around the hashtag `#TomsPhD` on Twitter, Google+, and Facebook.

Finally, I sincerely thank my parents and my brother who have made me the person I am today. My parents have taught me not to go for the easy choices, even if at times they may seem tempting, but instead to try harder and never give up. This thesis also is for you.

Formal Acknowledgements:

The research presented in this doctoral thesis was partially supported by the European Commission under Grant No. 248296 with the European Union (FP7 ICT STReP) project *I-SEARCH*.

To cite this document:

```
@phdthesis{steiner2013thesis,
  author = {Thomas Steiner},
  title  = {Enriching Unstructured Media Content About Events to
            Enable Semi-Automated Summaries, Compilations, and
            Improved Search by Leveraging Social Networks},
  year   = {2013},
  school = {Universitat Polit\`{e}cnica de Catalunya}
}
```

Copyright and License:

Contents

List of Figures

List of Listings

List of Tables

Glossary

API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BBC	British Broadcasting Corporation
BPEL	Business Process Execution Language
BPEL4WS	Business Process Execution Language for Web Services
CBIR	Content-based Image Retrieval
CBVR	Content-based Video Retrieval
CEO	Chief Executive Officer
CERN	European Organization for Nuclear Research
CES	Consumer Electronics Show
CPU	Central Processing Unit
CSS	Cascading Style Sheets
CURIE	Compact URI
DAWG	Data Access Working Group
DOM	Document Object Model
ERT WG	Evaluation and Repair Tools Working Group
Exif	Exchangeable image file format
FOAF	Friend of a friend
FP7	7th Framework Programme for Research and Technological Development
HD	High Definition
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ICT	Information and Communications Technology
IP	Internet Protocol
IRC	Internet Relay Chat
ISO	International Organization for Standardization
IT	Information Technology
JSON	JavaScript Object Notation
LOD	Linking Open Data
LOVS	Loose Order, Varying Sizes
MOS	Mean Opinion Score
NEE	Named Entity Extraction
NER	Named Entity Recognition
NERD	Named Entity Recognition and Disambiguation
NLP	Natural Language Processing
NPT	Normal Play Time
OCR	Optical Character Recognition
OWL	Web Ontology Language
PC	Personal Computer
PCRM	Pixel Change Ratio Map
PIPA	Preventing Real Online Threats to Economic Creativity and Theft of Intellectual Property Act

GLOSSARY

POS	Part-of-Speech (tagging)
RAM	Random Access Memory
RDF	Resource Description Framework
RDFa	Resource Description Framework in Attributes
REST	Representational State Transfer
RFC	Request for comment
RSS	Really Simple Syndication
RT	ReTweet
SD	Standard Definition
SERP	Search Engine Results Page
SHA	Secure hash algorithm
SIFT	Scale-Invariant Feature Transform
SNS	Social Network Site
SNS	Social Network(ing) Site
SOAP	Simple Object Access Protocol
SOES	Strict Order, Equal Sizes
SOPA	Stop Online Piracy Act
SPARQL	SPARQL Protocol and RDF Query Language
STReP	Specific Targeted Research Projects
SURF	Speeded Up Robust Features
TED	Technology, Entertainment, Design
tf-idf	Term frequency-inverse document frequency
Turtle	Terse RDF Triple Language
UDDI	Universal Description, Discovery and Integration
URI	Unique Resource Identifier
URL	Unique Resource Locator
US-ASCII	American Standard Code for Information Interchange
UTC	Universal Time Coordinated
UTF-8	8-bit Unicode Transformation Format
W3	World Wide Web
W3C	World Wide Web Consortium
WAI	Web Accessibility Initiative
WSDL	Web Services Description Language
WWW	World Wide Web
XFN	XHTML Friends Network
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language
XOXO	eXtensible Open XHTML Outlines

1

Event Summarization Challenge

1.1 Motivation and Problem Statement

A very open definition of the word *event* given by WordNet [8, 17] is *"something that happens at a given place and time."* Following this definition, we are indeed surrounded by events, most of which are of little to no interest for us. A concert somewhere in the world of a band that we do not even know may be a good example. For some events, however, we may care more, for example, a concert of a band that we know and like, even if it takes place at a location far away from us. Finally, for very few events, we may care a lot, maybe even enough to physically attend the event, like a concert of our favorite band if it takes place in our city, is not sold out, and not too expensive.

All this motivates the need for *event summarization.* If there is an event that we could not attend for any given reason, but that we are interested in, a good event summarization can help us get a feeling for the event's atmosphere. Similarly, if there is an event that we attended, we can revive the event's most fascinating moments based on the event summarization.

A *media gallery* in the context of our event summarization task is a *best-of* compilation of photos, videos, and microposts retrieved from social networks that are related to a given event. Event summarization covers textual as well as multimedia content. We say a media gallery is of high quality, if it fulfills the following properties.

1. *Conciseness:* it conveys a lot of information clearly and in few media items.

2. *Comprehensiveness:* it is complete and covers all representative elements or aspects of an event.

3. *Authenticity:* it is of undisputed origin and genuine.

4. *Diversity:* it shows a great deal of variety.

5. *Interestingness:* it catches and holds the attention of the viewer.

1.2 Research Question and Hypothesis

The main research question for this thesis can be formulated as follows.

"Can user-customizable media galleries that summarize given events be created solely based on textual and multimedia data from social networks?"

The hypothesis that we test in this thesis can be formulated as follows.

We argue that through media galleries that leverage content that was shared on social networks, a more *authentic*, more *concise*, more *comprehensive*, more *diverse*, and also more *interesting* view on events gets possible than by limiting oneself to officially produced media content; and that further such media galleries can be generated more *efficiently* and *in shorter time* than manually produced media galleries.

We validate these subjective and objective criteria with experiments for events of different categories such as sports, politics, culture, leisure, music, conferences, *etc.*

1.3 Approach

The objective of this thesis is the development of methods for the automated summarization of events based on media items shared on social networks. A schematic overview of the approach can be seen in Figure 1.1. As an event takes places and shortly thereafter (symbolized by the timeline marked with *2h Event*), people share media items related to the event on multiple social networks (symbolized by the photo and video pictograms above the event timeline). Via the textual search APIs (Application Programming Interfaces) of these different social networks, we retrieve a list of potentially event-relevant microposts that either contain media items directly, or that provide links to media items on external media item hosting platforms. Using third-party NLP tools, we recognize and disambiguate named entities in the microposts to predetermine their relevance. We extract the binary media item data from social networks or media item hosting platforms and relate it to the originating microposts (symbolized by the central cloud). Using CBIR and CBVR techniques, we first deduplicate exact-duplicate and near-duplicate media items, and then cluster similar media items (symbolized by the green, red, and orange markers). We rank the deduplicated and clustered list of media items and their related microposts according to well-defined ranking criteria. In order to generate interactive and user-customizable media galleries that visually and audially summarize the event in question, we compile the top-n ranked media items and microposts in an aesthetic way (symbolized by the timeline marked with *5min Summary*).

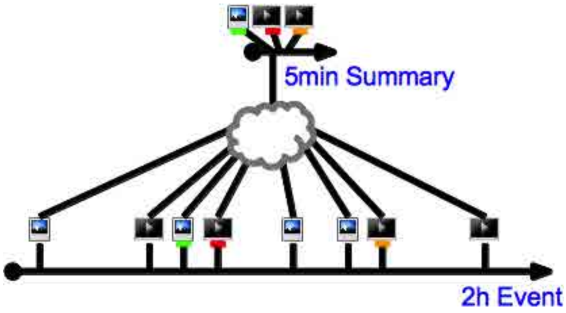


Figure 1.1: Schematic depiction of event summary generation based on deduplicated, clustered, and ranked media items for an exemplary event

1.4 Related Work

In this section, we provide a brief non-exhaustive meta overview of related work for the task of summarizing events based on social network multimedia data, namely videos and photos. More detailed studies of the particular states-of-the-art for each relevant subtask can be found in the upcoming chapters.

In recent years, social media has made rapid strides from a smiled-at phenomenon toward becoming a source of breaking news that is to be taken seriously and that probably has gone mainstream for the first time with Jānis Krūms' tweet on the US Airways Flight 1549 plane crash. At the very moment of the crash, Krūms was a regular passenger on a ferry and happened to witness the crash and posted a widely shared photo on the media hosting platform Twitpic, distributed via the social network Twitter that can be seen in Figure 1.2. As a result of the growing importance of social media, common news media like TV stations and (online) newspapers, but also news agencies themselves, use social networks as a regular source of content. Hashtags[1] are more

[1] People use the hashtag symbol # before a relevant keyword or phrase in social network posts to categorize them and help them show more easily in search

and more frequently displayed and propagated around events to facilitate gathering social media. An example around the event of the elections for the German Bundestag is the hashtag #btw2013, which stands for “Bundestagswahl 2013”. News media then create and publish hand-curated social media galleries like in the example in Figure 1.3 that feature prominent or widely shared social network contributions. This can help convey the feeling of the social network community about an event (which sometimes is interpolated to represent the feeling of the whole population).

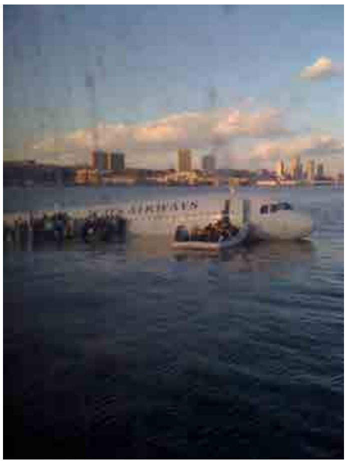


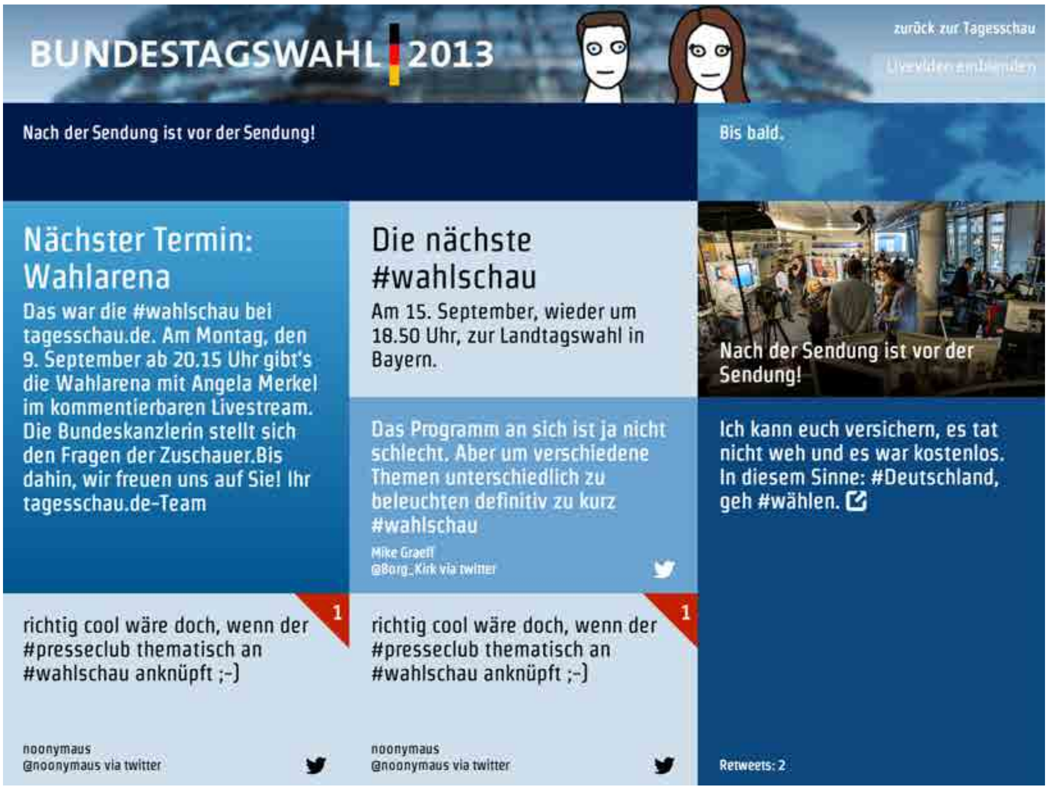
Figure 1.2: Tweet by Jānis Krūms (@jkrums, https://twitter.com/jkrums/status/1121915133): “http://twitpic.com/135xa – There’s a plane in the Hudson. I’m on the ferry going to pick up the people. Crazy.”

Examples of such manual social media curation tools are FlypSite,[1] a tool that facilitates the creation of embeddable second screen applications or TV social media widgets, Storify [2, 9],[2] a social network service that lets users create stories or timelines using social media, and Storyful,[3] a news agency focused on verifying and distributing user-generated content relating to news events from social networks. More automated approaches for content identification exist, for example, [14] and [15] by Liu *et al.* who combine semantic inference and visual analysis to automatically find media items that illustrate events. Further, there are [5] and [4] by Becker *et al.* who focus on identifying media items related to events by learning similarity metrics and identifying search terms. These approaches do not help with the tasks of ranking [13], deduplicating [58],

[1] http://www.flyp.tv/

[2] http://storify.com/

[3] http://storyful.com/



Nach der Sendung ist vor der Sendung!

Bis bald.

Nächster Termin: Wahlarena

Das war die #wahlschau bei tagesschau.de. Am Montag, den 9. September ab 20.15 Uhr gibt's die Wahlarena mit Angela Merkel im kommentierbaren Livestream. Die Bundeskanzlerin stellt sich den Fragen der Zuschauer.Bis dahin, wir freuen uns auf Sie! Ihr tagesschau.de-Team

Die nächste #wahlschau

Am 15. September, wieder um 18.50 Uhr, zur Landtagswahl in Bayern.

Nach der Sendung ist vor der Sendung!

Das Programm an sich ist ja nicht schlecht. Aber um verschiedene Themen unterschiedlich zu beleuchten definitiv zu kurz #wahlschau

Mike Graeff
@Borg_Kirk via twitter

Ich kann euch versichern, es tat nicht weh und es war kostenlos. In diesem Sinne: #Deutschland, geh #wählen.

richtig cool wäre doch, wenn der #presseclub thematisch an #wahlschau anknüpft ;-)

noonymaus
@noonymaus via twitter

richtig cool wäre doch, wenn der #presseclub thematisch an #wahlschau anknüpft ;-)

noonymaus
@noonymaus via twitter

Retweets: 2

Figure 1.3: Social media visualization for the elections to the German Bundestag (`http://wahlschau.tagesschau.flyp.tv/`)

and representing the event-related content aesthetically [18, 20]. Event archiving services such as Eventifier[1] do a great job at storing all the social media content around entire events, however, do not rank the information. Closest to our approach is Seen[2] an engine that aggregates, organizes, and ranks media and collects information on topics trending in social media. Seen does not create interactive visualizations, whereas with our approach we create speech-enabled media item compilations. Finally, there is MediaFinder [45],[3] which uses a fork of our media item collector and which specializes on clustering media items based on named entities.

As we will motivate throughout the thesis, there is definitely a need for tools that make sense of events, that organize them into clusters, that summarize them, and that let users quickly grasp what happened.

[1] `http://eventifier.co/`

[2] `http://seen.co/`

[3] `http://mediafinder.eurecom.fr/`

1.5 Contributions

In this thesis, we report on methods for the automated generation of event summaries. This particular field of research touches on many related areas of research and research communities, amongst which social network research, multimedia content analysis, Semantic Web and Natural Language Processing (NLP), human factors in computing systems, and Web services. Early on in the process of this thesis, we have sought and incorporated expert feedback based on a Doctoral Consortium paper [22]. We have broken our contributions down into the following topics.

1.5.1 Social Network Multimedia and Data Analysis

We have worked on methods for the aggregation, extraction, deduplication, clustering, and compilation of social media contents from multiple social networks [19]. These methods were applied and evaluated for the enhancement of conference experiences [11, 12] and events in general [16, 21, 26, 29, 39, 40].

1.5.2 Application of Semantic Web and NLP Techniques

In order to make sense out of social network microposts, we have worked on methods to consolidate and rank the results of multiple named entity recognition and disambiguation APIs and to track their data provenance [37, 38]. We have applied and evaluated those methods for the consumer-oriented detection of trending microposts on a major commercial social network [25] and in the cultural heritage domain [10].

1.5.3 Video Content and Metadata Analysis

We have worked on methods for named entity extraction and disambiguation for online videos based on closed captions and other textual metadata, which make online video more accessible, searchable, and interconnected [23, 27]. Further, we have combined those textual methods with video content analysis methods for the on-the-fly detection of shot boundaries for online videos [36]. We have defined aesthetic principles for the automated generation of media gallery layouts for visual and audial event summarization based on social network multimedia data [41].

1.5.4 Event Detection

We have done research on online new event detection based on Wikipedia edit spikes. We have developed an application called *Wikipedia Live Monitor* that monitors Wikipedia article edits on different language versions of Wikipedia—as they happen in realtime [28].

1.5.5 Standardization and Specifications

We have helped to shape a W3C specification on media fragment addressing schemes for audio and video items [44]. Further, we have worked on the definition of a unified framework for the description of multimedia content objects [3, 6]. Finally, we have contributed to a white paper on the Future Media Internet Architecture [1].

1.5.6 Crowdsourcing

The video content analysis methods mentioned before were combined with methods for the crowdsourced detection of events in online videos [42]. We have further worked on crowdsourcing methods for the extraction of knowledge items from arbitrary Web pages at scale [31, 32].

1.5.7 Studies

We have contributed an examination of Linked Data usage and visualization techniques of a major commercial search engine [34]. In addition to that, we have studied the usefulness and relevance of social network updates which were added to search engine results pages (SERP) of a major commercial search engine [43].

1.5.8 Multimodal Search Engines

We have worked on an examination of context-aware querying for multimodal search engines [7, 33]. Further, we have studied user interface constraints on mobile and desktop devices for a multimodal search engine and demonstrated that those constraints can be overcome both effectively and efficiently [30].

1.5.9 Web Service Description

We have worked on methods for the semantic description of Web APIs, their discoverability, their automated consumption, their semantic interlinking, and their social aspects [46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57]. We have studied the feasibility of truly RESTful behavior for Web APIs in the sense of Dr. Roy Fielding [24].

1.5.10 Others

We have developed methods for unobtrusively fixing common annoyances and typographic issues on arbitrary Web pages [35].

1.6 Thesis Structure

Each chapter is closed by a final section called *Chapter Notes*, which contains references to the publications that the chapter is based upon and in some cases pointers to related material for further reading. The remainder of this thesis is structured as follows.

Chapter 2: This chapter introduces the Semantic Web and its technologies. Starting from the non-semantic Web, we show how structured data can be added to Web pages and briefly present DBpedia as a knowledge base founded on structured data extracted from Wikipedia. We then continue with the Resource Description Framework (RDF) and explain how it represents facts with triples. We provide examples of RDF's different serialization formats. Afterwards, we outline the Semantic Web vision of a global giant database and present the Semantic Web query language SPARQL. We close the chapter with an introduction of Sir Tim Berners-Lee's Linked Data principles and show how data publisher that publish datasets according to those principles are visualized in the Linking Open Data cloud.

Chapter 3: This chapter provides the necessary definitions and terms that we will use throughout this thesis. It introduces social networks and media platforms as concepts *per se* and then lists the most popular social networks together with their core features and multimedia data support. We briefly look at decentralized social networks and explain why we do not consider them in this thesis. Finally, we propose a classification scheme for social networks that classifies them by their level of media item support.

Chapter 4: A micropost is defined as a textual status message on a social network. In this chapter, we describe how microposts can be semantically annotated in order to make sense of their contents. We show how we have developed two browser extensions to obtain access to real-world micropost data of actual micropost consumers. In continuation, we show how Natural Language Processing (NLP) Web services, machine translation, and part-of-speech tagging (POS) are combined by our annotation workflow and how the results of multiple NLP services are consolidated and reconciled. We show how provenance information can be automatically added to the generated output, so that the contribution of each Web service to the combined result is traceable, which is desirable to acknowledge and credit each service's work and also for debugging purposes.

Chapter 5: This chapter is about breaking news event detection based on concurrent Wikipedia edits. We have developed an application that automatically reports breaking news event candidates by clustering articles from multi-language editing activity streams and checking if well-defined breaking news conditions are fulfilled. The application uses social networks for plausibility checks in order to avoid false-positive alerts, *i.e.*, a breaking news event has to be reflected on Wikipedia *and* on social networks. We evaluate the event detection system with various global and local news events for its timeliness and accuracy.

Chapter 6: A media item is defined as a photo or video file that is publicly shared or published on at least one social network. In this chapter, we describe how media items can be extracted from different social networks. We introduce an alignment scheme that acts as an abstraction layer to overcome the underlying differences in data structure of the supported social networks. We evaluate the media item extractors with nine different events and motivate the need for media item deduplication and ranking.

Chapter 7: In video production and filmmaking, a *camera shot* is a series of frames that runs for an uninterrupted period of time. In this chapter, we present an algorithm and an accompanying application for the task of detecting camera shot boundaries on-the-fly in streaming Web video, which is a required step for the deduplication of videos and photos contained in videos. We evaluate the approach with videos from a popular video hosting platform in form of a browser extension.

Chapter 8: This chapter is about the on-the-fly deduplication and clustering of media items extracted from social networks. We analyze reasons for the occurrence of exact-duplicate and near-duplicate media items and introduce an algorithm tailored to this task, incorporating matching conditions that are based on the findings of the analysis. We evaluate the algorithm with two events and show its effectiveness. A media fragment is a part of a media file of the same media content type as its parent resource, *i.e.*, photo or video, that can be identified using a URI. We show a novel approach to debugging algorithms by combining media fragments URIs and speech synthesis, so that non-expert human raters can understand why or why not media items are clustered.

Chapter 9: In this chapter, we focus on ranking media item clusters that contain visually similar media items. We show how social interactions from different social networks can be merged in order to obtain a network-agnostic view on the performance of the clustered media items. Further, we show an algorithm for the selection of one representative media item per cluster that represents all media items contained in the same cluster. We then propose a ranking formula that is based on both social interactions and other features. We evaluate our ranking formula by comparing the ranked results for a given event with event highlight summaries regarding the same event that were created by different social networks.

Chapter 10: This chapter is about the compilation of ranked media item clusters to interactive media galleries. We define aesthetic principles that media galleries should fulfill based on high-level and low-level features. We show different media gallery styles and analyze their advantages and disadvantages. Examples of media gallery styles are interactive style, aspect-ratio-preserving style, or order-preserving style, among others. Further, we present an approach to make media galleries interactive by using a speech synthesis system combined with media item animations. At the end of the chapter, we describe the application *Social Media Illustrator* that is the main software outcome of this doctoral thesis.

Chapter 11: In the final chapter of this thesis, we provide conclusions and give an outlook on future work. We focus on media item verification and authenticity, the evaluation of subjective data with multi-armed bandit experiments, further application domains were our application could find use, and close with a comparison of commercial and academic event summarization and archiving Web applications.

References

[1] Maria Alduan, Federico Álvarez, Jan Bouwen, Gonzalo Camarillo, Pablo Cesar, Pedros Daras, et al. *Future Media Internet Architecture Reference Model (v1. 0).* 2011. URL: http://www.coast-fp7.eu/public/FMIA_Reference_Architecture.pdf.

[2] Berke Atasoy and Jean-Bernard Martens. "STORIFY: A Tool to Assist Design Teams in Envisioning and Discussing User Experience". In: *CHI '11 Extended Abstracts on Human Factors in Computing Systems.* CHI EA '11. Vancouver, BC, Canada: ACM, 2011, pp. 2263–2268. ISBN: 978-1-4503-0268-5. DOI: 10.1145/1979742.1979905. URL: http://doi.acm.org/10.1145/1979742.1979905.

[3] Apostolos Axenopoulos, Petros Daras, Sotiris Malassiotis, Vincenzo Croce, Marilena Lazzaro, Jonas Etzold, et al. "I-SEARCH: A Unified Framework for Multimodal Search and Retrieval". In: *The Future Internet.* Ed. by Federico Álvarez, Frances Cleary, Petros Daras, John Domingue, Alex Galis, Ana Garcia, et al. Vol. 7281. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 130–141. ISBN: 978-3-642-30240-4. URL: http://www.lsi.upc.edu/~tsteiner/papers/2012/isearch-a-unified-framework-for-multimodal-search-and-retrieval.pdf.

[4] Hila Becker, Dan Iter, Mor Naaman, and Luis Gravano. "Identifying Content for Planned Events Across Social Media Sites". In: *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining.* WSDM '12. ACM, 2012, pp. 533–542.

[5] Hila Becker, Mor Naaman, and Luis Gravano. "Learning Similarity Metrics for Event Identification in Social Media". In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining.* WSDM '10. ACM, 2010, pp. 291–300.

[6] Petros Daras, Apostolos Axenopoulos, Vasileios Darlagiannis, Dimitrios Tzovaras, Xavier Le Bourdon, Laurent Joyeux, et al. "Introducing a Unified Framework for Content Object Description". In: *Multimedia Intelligence and Security* 2.3 (2011), pp. 351–375. ISSN: 2042–3470. URL: http://www.iti.gr/iti/files/document/work/IJMIS0203-0409%20DARAS.pdf.

[7] Jonas Etzold, Arnaud Brousseau, Paul Grimm, and Thomas Steiner. "Context-Aware Querying for Multimodal Search Engines". In: *Advances in Multimedia Modeling.* Ed. by Klaus Schoeffmann, Bernard Merialdo, Alexander G. Hauptmann, et al. Vol. 7131. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 728–739. ISBN: 978-3-642-27354-4. URL: `http://research.google.com/pubs/archive/37423.pdf`.

[8] Christiane Fellbaum. *WordNet: An Electronic Lexical Database.* Language, Speech and Communication Series. Cambridge, MA: MIT Press, 1998.

[9] Kelly Fincham. "Review: Storify (2011)". In: *Journal of Media Literacy Education* 3.1 (2011).

[10] Seth van Hooland, Max De Wilde, Ruben Verborgh, Thomas Steiner, and Rik Van de Walle. "Named-Entity Recognition: A Gateway Drug for Cultural Heritage Collections to the Linked Data Cloud?" In: *Literary and Linguistic Computing* (2013). URL: `http://freeyourmetadata.org/publications/named-entity-recognition.pdf`.

[11] Houda Khrouf, Ghislain Atemezing, Giuseppe Rizzo, Raphaël Troncy, and Thomas Steiner. "Aggregating Social Media for Enhancing Conference Experience". In: *Real-Time Analysis And Mining of Social Streams, Papers from the 2012 ICWSM Workshop.* Ed. by Arkaitz Zubiaga, Maarten de Rijke, Markus Strohmaier, and Mor Naaman. AAAI Technical Report WS-12–02. June 2012. URL: `http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/download/4779/5086`.

[12] Houda Khrouf, Ghislain Atemezing, Thomas Steiner, Giuseppe Rizzo, and Raphaël Troncy. *Confomaton: A Conference Enhancer with Social Media from the Cloud.* 2012. URL: `http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_343.pdf`.

[13] Tie-Yan Liu. "Learning to Rank for Information Retrieval". In: *Found. Trends Inf. Retr.* 3.3 (Mar. 2009), pp. 225–331. ISSN: 1554-0669.

[14] Xueliang Liu, Raphaël Troncy, and Benoit Huet. "Finding Media Illustrating Events". In: *Proceedings of the 1st ACM International Conference on Multimedia Retrieval.* ICMR '11. ACM, 2011, pp. 1–8.

[15] Xueliang Liu, Raphaël Troncy, and Benoit Huet. "Using Social Media to Identify Events". In: *Proceedings of the 3rd ACM SIGMM International Workshop on Social Media.* WSM '11. 2011, pp. 3–8.

[16] Vuk Milicic, Giuseppe Rizzo, José Luis Redondo Garcia, Raphaël Troncy, and Thomas Steiner. “Live topic generation from event streams”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 285–288. ISBN: 978-1-4503-2038-2. URL: http://dl.acm.org/citation.cfm?id=2487788.2487924.

[17] George A. Miller. “WordNet: a Lexical Database for English”. In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[18] Pere Obrador, Michele Saad, Poonam Suryanarayan, and Nuria Oliver. “Towards Category-Based Aesthetic Models of Photographs”. In: *Proceedings of the 18th International Conference on Advances in Multimedia Modeling – Volume Part I (MMM 2012).* 2012, pp. 63–76.

[19] Giuseppe Rizzo, Thomas Steiner, Raphaël Troncy, Ruben Verborgh, José Luis Redondo García, and Rik Van de Walle. “What Fresh Media Are You Looking For?: Retrieving Media Items From Multiple Social Networks”. In: *Proceedings of the 2012 International Workshop on Socially-aware Multimedia.* SAM ’12. Nara, Japan: ACM, 2012, pp. 15–20. ISBN: 978-1-4503-1586-9. URL: http://www.eurecom.fr/~troncy/Publications/Troncy-saw12.pdf.

[20] Philipp Sandhaus, Mohammad Rabbath, and Susanne Boll. “Employing Aesthetic Principles for Automatic Photo Book Layout”. In: *Proceedings of the 17th International Conference on Advances in Multimedia Modeling – Volume Part I (MMM 2011).* 2011, pp. 84–95.

[21] Thomas Steiner. “A meteoroid on steroids: ranking media items stemming from multiple social networks”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 31–34. ISBN: 978-1-4503-2038-2. URL: http://dl.acm.org/citation.cfm?id=2487788.2487798.

[22] Thomas Steiner. “DC Proposal: Enriching Unstructured Media Content About Events to Enable Semi-Automated Summaries, Compilations, and Improved Search by Leveraging Social Networks”. In: *Proceedings of the 10th International Conference on The Semantic Web – Volume Part II.* ISWC’ 11. Bonn, Germany: Springer-Verlag, 2011, pp. 365–372. ISBN: 978-3-642-25092-7. URL: http://iswc2011.semanticweb.org/fileadmin/iswc/Papers/DC_Proposals/70320369.pdf.

[23] Thomas Steiner. “SemWebVid – Making Video a First Class Semantic Web Citizen and a First Class Web Bourgeois”. In: *Proceedings of the ISWC 2010 Posters & Demonstrations Track: Collected Abstracts, Shanghai, China, November 9, 2010.* Ed. by Axel Polleres and Huajun Chen. Vol. 658. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2010, pp. 97–100. URL: `http://ceur-ws.org/Vol-658/paper469.pdf`.

[24] Thomas Steiner and Jan Algermissen. “Fulfilling the Hypermedia Constraint via HTTP OPTIONS, the HTTP Vocabulary in RDF, and Link Headers”. In: *Proceedings of the Second International Workshop on RESTful Design.* WS-REST ’11. Hyderabad, India: ACM, 2011, pp. 11–14. ISBN: 978-1-4503-0623-2. URL: `http://ws-rest.org/2011/proc/a3-steiner.pdf`.

[25] Thomas Steiner, Arnaud Brousseau, and Raphaël Troncy. *A Tweet Consumers’ Look At Twitter Trends.* May 2011. URL: `http://research.hypios.com/msm2011/posters/steiner.pdf`.

[26] Thomas Steiner and Christopher Chedeau. “To crop, or not to crop: compiling online media galleries”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 201–202. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2487890`.

[27] Thomas Steiner and Michael Hausenblas. *SemWebVid – Making Video a First Class Semantic Web Citizen and a First Class Web Bourgeois, Submission to the Open Track of the Semantic Web Challenge 2010.* Nov. 2010. URL: `http://challenge.semanticweb.org/submissions/swc2010_submission_12.pdf`.

[28] Thomas Steiner, Seth van Hooland, and Ed Summers. “MJ no more: using concurrent wikipedia edit spikes with social network plausibility checks for breaking news detection”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 791–794. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2488049`.

[29] Thomas Steiner, Seth van Hooland, Ruben Verborgh, Joseph Tennis, and Rik Van de Walle. “Identifying VHS Recording Artifacts in the Age of Online Video Platforms”. In: *Proceedings of the 1st international Workshop on Search and Exploration of X-rated Information.* Feb. 2013. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2013/identifying-vhs-recording-sexi2013.pdf`.

[30] Thomas Steiner, Marilena Lazzaro, Francesco Nucci, Vincenzo Croce, Lorenzo Sutton, Alberto Massari, et al. "One Size Does Not Fit All: Multimodal Search on Mobile and Desktop Devices with the I-SEARCH Search Engine". In: *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval.* ICMR '12. Hong Kong, China: ACM, 2012, 58:1–58:2. ISBN: 978-1-4503-1329-2. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/one-size-does-not-fit-all-icmr2012.pdf`.

[31] Thomas Steiner and Stefan Mirea. *SEKI@home, a Generic Approach for Crowdsourcing Knowledge Extraction from Arbitrary Web Pages.* Nov. 2012. URL: `http://challenge.semanticweb.org/2012/submissions/swc2012_submission_28.pdf`.

[32] Thomas Steiner and Stefan Mirea. "SEKI@home, or Crowdsourcing an Open Knowledge Graph". In: *Proceedings of the 1st International Workshop on Knowledge Extraction & Consolidation from Social Media, in conjunction with the 11th International Semantic Web Conference (ISWC 2012), Boston, USA, November 12, 2012.* Ed. by Diana Maynard, Stefan Dietze, Wim Peters, and Jonathon Hare. Vol. 895. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2012. URL: `http://ceur-ws.org/Vol-895/paper2.pdf`.

[33] Thomas Steiner, Lorenzo Sutton, Sabine Spiller, Marilena Lazzaro, Francesco Nucci, Vincenzo Croce, et al. "I-SEARCH: A Multimodal Search Engine based on Rich Unified Content Description (RUCoD)". In: *Proceedings of the 21st International Conference Companion on World Wide Web.* WWW '12 Companion. Lyon, France: ACM, 2012, pp. 291–294. ISBN: 978-1-4503-1230-1. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/isearch-multimodal-search-www2012.pdf`.

[34] Thomas Steiner, Raphaël Troncy, and Michael Hausenblas. "How Google is using Linked Data Today and Vision For Tomorrow". In: *Proceedings of the Workshop on Linked Data in the Future Internet at the Future Internet Assembly, Ghent 16–17 Dec 2010.* Ed. by Sören Auer, Stefan Decker, and Manfred Hauswirth. Vol. 700. CEUR Workshop Proceedings ISSN 1613-0073. Dec. 2010. URL: `http://CEUR-WS.org/Vol-700/Paper5.pdf`.

[35] Thomas Steiner and Ruben Verborgh. *Fixing the Web One Page at a Time, or Actually Implementing xkcd #37.* Apr. 2012. URL: `http://www2012.org/proceedings/nocompanion/DevTrack_032.pdf`.

[36] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, Michael Hausenblas, Raphaël Troncy, and Rik Van de Walle. *Enabling on-the-fly Video Shot Detection on YouTube*. Apr. 2012.

[37] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Facebook Microposts via a Mash-up API and Tracking its Data Provenance". In: *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on.* Oct. 2011, pp. 342–345. URL: `http://research.google.com/pubs/archive/37426.pdf`.

[38] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Social Network Microposts via Multiple Named Entity Disambiguation APIs and Tracking Their Data Provenance". In: *International Journal of Computer Information Systems and Industrial Management* 5 (2013), pp. 69–78. URL: `http://www.mirlabs.org/ijcisim/regular_papers_2013/Paper82.pdf`.

[39] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Near-duplicate Photo Deduplication in Event Media Shared on Social Networks". In: *Proceedings of the International Conference on Advanced IT, Engineering and Management*. Feb. 2013, pp. 187–188.

[40] Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, Erik Mannens, and Rik Van de Walle. "Clustering Media Items Stemming from Multiple Social Networks". In: *The Computer Journal* (2013). DOI: `10.1093/comjnl/bxt147`. eprint: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.full.pdf+html`. URL: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.abstract`.

[41] Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, and Rik Van de Walle. "Defining Aesthetic Principles for Automatic Media Gallery Layout for Visual and Audial Event Summarization based on Social Networks". In: *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on.* July 2012, pp. 27–28. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/defining-aesthetic-principles-for-automatic-media-gallery-layout-qomex2012.pdf`.

[42] Thomas Steiner, Ruben Verborgh, and Michael Hausenblas. "Crowdsourcing Event Detection in YouTube Videos". In: *Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011), Workshop in conjunction with the 10th International Semantic Web Conference 2011 (ISWC 2011), Bonn, Germany, October 23, 2011.* Ed. by Marieke van Erp, Willem Robert van Hage, Laura Hollink, Anthony Jameson, and Raphaël Troncy. Vol. 779. CEUR Workshop Proceedings ISSN 1613-0073. Oct. 2011, pp. 58–67. URL: `http://ceur-ws.org/Vol-779/derive2011_submission_8.pdf`.

[43] Thomas Steiner, Ruben Verborgh, Raphael Troncy, Joaquim Gabarro, and Rik Van de Walle. "Adding Realtime Coverage to the Google Knowledge Graph". In: *Proceedings of the ISWC 2012 Posters & Demonstrations Track, Boston, USA, November 11–15, 2012.* Ed. by Birte Glimm and David Huynh. Vol. 914. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2012. URL: `http://ceur-ws.org/Vol-914/paper_2.pdf`.

[44] R. Troncy, E. Mannens, S. Pfeiffer, D. Van Deursen, M. Hausenblas, P. Jägenstedt, et al. *Media Fragments URI 1.0 (basic).* Recommendation. `http://www.w3.org/TR/media-frags/`, accessed July 15, 2013. W3C, 2012.

[45] Raphaël Troncy, Vuk Milicic, Giuseppe Rizzo, and José Luis Redondo García. "MediaFinder: Collect, Enrich and Visualize Media Memes Shared by the Crowd". In: *Proceedings of the 22Nd International Conference on World Wide Web Companion.* WWW '13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 789–790. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2488048`.

[46] Ruben Verborgh, Vincent Haerinck, Thomas Steiner, Davy Van Deursen, Sofie Van Hoecke, Jos De Roo, et al. "Functional Composition of Sensor Web APIs ". In: *Proceedings of the 5th International Workshop on Semantic Sensor Networks, A Workshop of the 11th International Semantic Web Conference 2012 (ISWC 2012), Boston, Massachusetts, USA, November 12, 2012.* Ed. by Cory Henson, Kerry Taylor, and Oscar Corcho. Vol. 904. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2012, pp. 65–80. URL: `http://ceur-ws.org/Vol-904/paper6.pdf`.

[47] Ruben Verborgh, Andreas Harth, Maria Maleshkova, Steffen Stadtmüller, Thomas Steiner, Mohsen Taheriyan, et al. "Semantic Description of REST APIs". In: *rest: Advanced Research Topics and Practical Applications* (2013).

[48] Ruben Verborgh, Michael Hausenblas, Thomas Steiner, Erik Mannens, and Rik Van de Walle. "Distributed Affordance: An Open-World Assumption for Hypermedia". In: *Proceedings of the Fourth International Workshop on RESTful Design.* May 2013. URL: `http://distributedaffordance.org/publications/ws-rest2013.pdf`.

[49] Ruben Verborgh, Thomas Steiner, Davy Deursen, Jos Roo, RikVan de Walle, and Joaquim Gabarró Vallés. "Capturing the functionality of Web services with functional descriptions". In: *Multimedia Tools and Applications* (2012), pp. 1–23. ISSN: 1380-7501. URL: `http://rd.springer.com/content/pdf/10.1007/s11042-012-1004-5`.

[50] Ruben Verborgh, Thomas Steiner, Joaquim Gabarró Vallés, Erik Mannens, and Rik Van de Walle. "A Social Description Revolution—Describing Web APIs' Social Parameters with RESTdesc". In: *Proceedings of the AAAI 2012 Spring Symposia.* Mar. 2012. URL: `http://www.aaai.org/ocs/index.php/SSS/SSS12/paper/viewFile/4283/4665`.

[51] Ruben Verborgh, Thomas Steiner, Erik Mannens, Rik Van de Walle, and Joaquim Gabarró Vallés. "Proof-based Automated Web API Composition and Integration". In: *Proceedings of the International Conference on Advanced IT, Engineering and Management* (2013), pp. 181–182.

[52] Ruben Verborgh, Thomas Steiner, Rik Van de Walle, and Joaquim Gabarró Vallés. "The Missing Links – How the Description Format RESTdesc Applies the Linked Data Vision to Connect Hypermedia APIs". In: *Proceedings of the First Linked APIs Workshop at the Ninth Extended Semantic Web Conference.* May 2012. URL: `http://lapis2012.linkedservices.org/papers/3.pdf`.

[53] Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Sam Coppens, Joaquim Gabarró Vallés, and Rik Van de Walle. "Functional Descriptions as the Bridge between Hypermedia APIs and the Semantic Web". In: *Proceedings of the Third International Workshop on RESTful Design.* WS-REST '12. Lyon, France: ACM, 2012, pp. 33–40. ISBN: 978-1-4503-1190-8. URL: `http://ws-rest.org/2012/proc/a5-9-verborgh.pdf`.

[54] Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Sam Coppens, Erik Mannens, Rik Van de Walle, et al. "Integrating Data and Services through Functional Semantic Service Descriptions". In: *Proceedings of the W3C Workshop on Data and Services Integration.* Oct. 2011. URL: `http://www.w3.org/2011/10/integration-workshop/p/integration-ws-mmlab.pdf`.

[55] Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Sam Coppens, Erik Mannens, Rik Van de Walle, et al. *RESTdesc—A Functionality-Centered Approach to Semantic Service Description and Composition.* 2012. URL: `http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_302.pdf`.

[56] Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Jos De Roo, Rik Van de Walle, and Joaquim Gabarró Vallés. “Description and Interaction of RESTful Services for Automatic Discovery and Execution”. In: *Proceedings of the FTRA 2011 International Workshop on Advanced Future Multimedia Services.* Dec. 2011. URL: `https://biblio.ugent.be/publication/2003291/file/2003308.pdf`.

[57] Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Rik Van de Walle, and Joaquim Gabarró Vallés. “Efficient Runtime Service Discovery and Consumption with Hyperlinked RESTdesc”. In: *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on.* Oct. 2011, pp. 373–379. URL: `http://research.google.com/pubs/archive/37427.pdf`.

[58] Xin Yang, Qiang Zhu, and Kwang-Ting Cheng. “Near-duplicate Detection for Images and Videos”. In: *1st ACM Workshop on Large-Scale Multimedia Retrieval and Mining.* LS–MMRM ’09. 2009, pp. 73–80.

2

Semantic Web Technologies

The main contributions of this thesis are methods for the automated generation of user-customizable media galleries for the visual and audial summarization of events. To provide context for the proposed approaches in the later parts of the thesis, we start with two introductory chapters related to Semantic Web technologies and social networks. The current Chapter 2 covers the Semantic Web, Linked Data, and the Resource Description Framework (RDF). The following Chapter 3 will cover social networks by first providing a definition and classification of social networks, and then introducing popular social networks and some of their core features.

2.1 The World Wide Web and Semantics

Tim Berners Lee, inventor of the World Wide Web (W3, WWW), or simply, the *Web*, writes in [5]: *"The World Wide Web was developed to be a pool of human knowledge, which would allow collaborators in remote sites to share their ideas and all aspects of a common project."* Since its earliest days at CERN, the European Organization for Nuclear Research in Geneva, Switzerland, the Web has scaled to a truly global system of interlinked hypertext documents accessed via the Internet.

Semantics is the study of meaning. It focuses on the relation between words, phrases, signs, and symbols, and what they stand for, *i.e.*, the actual object referred to by a linguistic expression. Michel Bréal can be counted as the founder of modern semantics with his 1897 *Essai de sémantique* [14]. The *Semantic Web* brings these two worlds—the World Wide Web and semantics—together.

2.2 The Semantic Web

The lexical database WordNet [23, 32] by the Cognitive Science Laboratory of Princeton University defines the term *semantic* as *"of or relating to meaning or the study of meaning."* The same source defines the term *Web*, which is a common form for the complete term *World Wide Web* as *"computer network consisting of a collection of internet sites that offer text and graphics and sound and animation resources through the hypertext transfer protocol."* Finally, WordNet defines the term *meaning* as *"the message that is intended or expressed or signified,"* or *"the idea that is intended."*

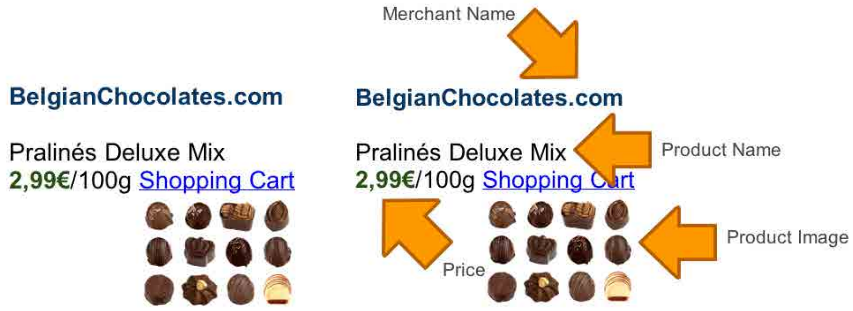
The combined term *Semantic Web* was coined by Sir Tim Berners-Lee, in a May 2001 article co-published with James Hendler and Ora Lassila in the Scientific American [8].

> *"The Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users. [. . .] The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation. The first steps in weaving the Semantic Web into the structure of the existing Web are already under way. In the near future, these developments will usher in significant new functionality as machines become much better able to process and* understand *the data that they merely display at present."*

We are currently experiencing a fundamental shift from the World Wide Web to the Semantic Web, a shift from moving bits to moving bits with a meaning. This can have a huge impact, which might not be as drastic as Tim Berners-Lee describes in his Scientific American article, but which might introduce many improvements, like more accurate search results, more intelligent price comparison services, *etc.* Figure 2.1 illustrates this idea.

2.2.1 The Non-Semantic Web

To differentiate the Semantic Web from the non-semantic Web, it helps to step back one step and see why the non-semantic Web is not semantic. The Web is a system of interlinked hypertext documents accessed through the Internet. These documents



(a) Bits without meaning.

(b) Bits with a meaning.

Figure 2.1: Fundamental shift from moving bits to moving bits with a meaning

are typically marked up in the Hypertext Markup Language (HTML), a language that defines a syntax understandable to user agents like Web browsers, however, not one that provides meaning beyond the level of text layout. This means that an HTML snippet like the one below

```
<h1>The Catcher in the Rye</h1>
<h2>J. D. Salinger</h2>
```

reveals that *The Catcher in the Rye* is a level one header element and that *J. D. Salinger* a level two header element, but to a machine it is not evident that the prior is the title of a book, and that the latter is (i) a book author, and (ii) the author of *The Catcher in the Rye*.

2.2.2 Structured Data on the Web

A very first step towards adding semantics to the Web is using tabular data. Table 2.1 shows an example for such tabular data. For human beings (interested in sports), the meaning of the columns in Table 2.1 is clear:

P = matches **P**layed
W = **W**ins
D = **D**raws
L = **L**osses
F = Goals **F**or
A = Goals **A**gainst
Pts = **P**oin**ts**

Team	P	W	D	L	F	A	Pts
Manchester United	6	4	0	2	10	5	12
Celtic	6	3	0	3	8	9	9
Benfica	6	2	1	3	7	8	7
FC Copenhagen	6	2	1	2	5	8	7

Table 2.1: Sample table with structured data for sports results

The problem, however, is for machines to understand the structure of the table. Let us imagine one wanted to automate the task of retrieving sports results from a Web page with tabular data. While it is a straightforward job to implement a scraper bot that searches for column titles like "P", "W", "D", *etc.*, it would require the same work over and over again for a different language, for example, German, where the terms would be: "Sp." (Spiele), "g." (gewonnen), "u." (unentschieden), "v." (verloren), "Tore" (Tore), "Pkte." (Punkte). A German-speaking reader might have noticed that the exemplary German system listed here does not differentiate between *goals for* and *goals against*, but only has a list of *Goals*. Tiny differences like this make the scraping approach brittle. If data providers were to use unique column identifiers like Unique Resource Identifiers (URIs), the problem would be easier. In the concrete example for English and German, rather than using "D" (*Draws*) and "u." (*unentschieden*), which both mean that the result was a tie, the machine-readable column name could instead be identified by a *Unique* Resource Identifier (URI) like `http://dbpedia.org/page/Tie_(draw)`, or even a fictive URI like `http://example.org/VGllXyhkcmF3KQ==`. In the next section, we therefore introduce the structured knowledge base and interlinking hub in the Web of Data, DBpedia [2].

2.2.3 The Structured Knowledge Base DBpedia

An often reoccurring pattern in the Semantic Web world is the use of DBpedia [2] as a hub for identifying concepts by URIs. DBpedia is a Semantic Web knowledge base with the objective of automatically extracting pre-structured tabular data from the human-generated info-boxes from the online encyclopedia Wikipedia.[1] This pre-structured information is then made available on the World Wide Web in many formats, for example, in JSON [18] and many RDF [29] serializations. DBpedia al-

[1] `http://en.wikipedia.org/wiki/Main_Page`, accessed July 15, 2013

lows for querying relationships and properties associated with Wikipedia resources, including links to other related datasets. As outlined before, the concept of a tie draw in the sense of sports could thus be uniquely identified by the DBpedia URI `http://dbpedia.org/page/Tie_(draw)`, free of all ambiguity. Similar knowledge bases are among others Freebase [12, 31], YAGO [42], and CYC [30].

2.2.4 Semantics in HTML Versions 4.01 and 5

As outlined in subsection 2.2.1, HTML versions 4.01 [37] and 5 [3] contain a basic level of semantics. The main focus, however, is on the separation of the markup of the textual structure from the actual presentation. For example the `<b>` and the `<strong>` tags both have the same visual effect: they make the node value appear in a bold face **like so**. Visually, there is no way to differentiate between the two, however, semantically the difference exists and is well-defined: `<strong>` should be used when one wants to give special emphasis on something. Screen readers will typically read out such text with a more emphasized voice. In contrast, `<b>` should be used if only visually one wants to create a bold face look. In the following, we present a list of semantic HTML tags and attributes and their meaning.

Semantic HTML 4.01 Tags:

- `<abbr>` specifies an abbreviation, `<acronym>` specifies an acronym.
- `<h1>-<h6>` specify level 1–6 headers, `<caption>` specifies a caption for a table.
- `<blockquote>` specifies a block-level quotation (a source in form of a URI may be specified via the `cite` attribute), `<cite>` specifies a citation.
- `<dl>` specifies a definition list, `<dt>` specifies a definition term in a definition list, `<dd>` specifies the definition of a term in a definition list.
- `<em>` specifies an emphasis, `<strong>` specifies a strong emphasis.
- `<code>` specifies a code snippet, `<dfn>` specifies an inline definition of a single term, `<address>` specifies contact information for the document author, `<legend>` specifies a legend for `<fieldset>` containers for adding structure to forms, `<samp>` specifies sample output from a script or program.

Semantic HTML5 Tags:

- `<article>` specifies an independent item section of content, `<aside>` specifies a section of a page that consists of content that is tangentially related to the content around the `<aside>` element, and which could be considered separate from that content, `<header>` specifies a group of introductory or navigational aids, `<footer>` specifies a footer for its nearest ancestor sectioning content or sectioning root element, `<nav>` specifies a section with navigation links.
- `<figure>` specifies some flow content, `<mark>` specifies a run of text in one document marked or highlighted for reference purposes due to its relevance in another context, `<meter>` specifies a scalar measurement within a known range, or a fractional value.
- `<audio>` specifies a sound or an audio stream, `<video>` specifies a video or movie.
- `<progress>` specifies the completion progress of a task, `<time>` specifies either a time on a 24 hour clock, or a precise date in the calendar (optionally with a time and a time-zone offset), `<command>` provides an abstraction layer between user interface and commands, so that multiple user interface elements can refer to the same command.
- `<details>` specifies a disclosure widget from which the user can obtain additional information or controls, `<datalist>` specifies the list that represents predefined options for input elements.
- `<keygen>` specifies a key pair generator control, `<output>` specifies the result of a calculation, `<ruby>` allows one or more spans of phrasing content to be marked with ruby annotations.

HTML5 Input Type Attributes:

- `datetime` specifies a control for setting the element's value to a string representing a global date and time (with timezone information).
- `datetime-local` specifies a control for setting the element's value to a string representing a local date and time (with no timezone information).

- `date` specifies a control for setting the element's value to a string representing a date, `month` specifies a control for setting the element's value to a string representing a month, `week` specifies a control for setting the element's value to a string representing a week.
- `time` specifies a control for setting the element's value to a string representing a time (with no timezone information).
- `number` specifies a control for setting the element's value to a string representing a number.
- `range` represents an imprecise control for setting the element's value to a string representing a number.
- `email` specifies a control for editing a list of email addresses given in the element's value. A regular expression can be used to validate the email.
- `url` specifies a control for editing an absolute URL given in the element's value. A regular expression can be used to validate the URL.
- `search` specifies a one-line plain-text edit control for entering one or more textual search terms.
- `color` specifies a color-well control for setting the element's value to a string representing a simple color.

2.2.5 Structured Data Beyond Pure HTML

In this subsection, we describe how structured data can be included in HTML documents by either overloading existing HTML attributes or by adding new ones.

Microformats

Microformats [17] are a set of open data mark-up formats developed and defined by the Microformats community.[1] Microformats are not an official standard, but rather a widely adopted grass-roots-driven movement with origins in the blogging scene. It is to be noted that Microformats do not require a new language, but reuse building blocks

[1] `http://microformats.org/discuss`, accessed July 15, 2013

from widely adopted standards such as the `class`, `rel`, and `title` attributes in HTML. Their main design goal is to focus first on humans, then on machines. A concrete example of Microformat mark-up in HTML can be seen in Listing 2.1. There are currently nine stable Microformats,[1] as listed below:

- `hCalendar` is a distributed calendaring and events format, using a 1:1 representation of the standard `iCalendar` format (RFC 2445, [22]).
- `hCard` is a format for representing people, companies, organizations, and places, using a 1:1 representation of the standard `vCard` format (RFC 2426, [21]).
- `rel-license` is a format for indicating content licenses, which is embeddable in HTML [37] or XHTML [34], Atom [33], RSS [16], and arbitrary XML [13].
- `rel-nofollow` is a format for hyperlinks indicating that the destination of that hyperlink should not be afforded any additional weight or ranking by user agents such as search engines, which perform link analysis upon Web pages.
- `rel-tag` is a format for hyperlinks indicating that the destination of that hyperlink is an author-designated keyword for the current page.
- `VoteLinks` is a format for adding the idea of agreement, abstention or indifference, and disagreement to hyperlinks.
- `XFN` is a format for representing human relationships (XHTML Friends Network) using hyperlinks, which enables Web authors to indicate their relationships to other people.
- `XMDP` is a format for defining metadata profile documents (XHTML Meta Data Profile), which enables Web authors to well-define custom meta tags.
- `XOXO` is a format for defining a new XHTML [34] document type for subsetting and extending XHTML, which serves as the basis for XHTML-friendly outlines (eXtensible Open XHTML Outlines) for processing by XML engines and for easy interactive rendering by browsers.

[1] `http://microformats.org/wiki/Main_Page#Specifications`, accessed July 15, 2013

```
<div class="vcard">
  <a class="fn org url" href="http://www.commerce.net/">CommerceNet</a>
  <div class="adr">
    <span class="type">Work</span>:
    <div class="street-address">169 University Avenue</div>
    <span class="locality">Palo Alto</span>,
    <abbr class="region" title="California">CA</abbr>
    <span class="postal-code">94301</span>
    <div class="country-name">USA</div>
  </div>
  <div class="tel">
   <span class="type">Work</span> +1-650-289-4040
  </div>
  <div class="tel">
    <span class="type">Fax</span> +1-650-289-4041
  </div>
  <div>Email:
   <span class="email">info@commerce.net</span>
  </div>
</div>
```

Listing 2.1: Sample code snippet with embedded `hCard` Microformat mark-up (`http://microformats.org/wiki/hcard`)

Microdata

Microdata [28] defines a way to annotate content (or items) with specific machine-readable labels, for example, to allow scripts to provide services that are customized to a website. Microdata allows for nested groups of name-value pairs to be added to documents, in parallel with the existing content. The Microdata specification introduces a set of new attributes to HTML:

- `itemscope` creates an item (or thing) and indicates that descendants of this element contain information about it. This attribute precedes the `itemtype` attribute in the HTML element's tag.
- `itemtype` a valid URL of a vocabulary that describes the item in question and its properties context.
- `itemid` indicates a unique identifier of the item in the vocabulary.
- `itemprop` indicates that its containing tag holds the value of the specified item property. The properties name and value context are described by the items

vocabulary. Properties values usually consist of string values, but can also use URLs using the `<a>` tag and its `href` attribute, the `<img>` tag and its `src` attribute, or other tags that link to or embed external resources.

- `itemref` properties that are not descendants of the element with the `itemscope` attribute can be associated with the item using this attribute. It provides a list of elements to Web crawlers to find additional property values of the item elsewhere in the document.

An example of Microdata in HTML can be seen in Listing 2.2.

```
<div itemscope>
  <p>My name is <span itemprop="name">Neil</span>.</p>
  <p>
    My band is called
    <span itemprop="band">Four Parts Water</span>.
  </p>
  <p>I am <span itemprop="nationality">British</span>.</p>
</div>
```

Listing 2.2: Sample code snippet with embedded Microdata mark-up (`http://www.w3.org/TR/microdata/`)

2.3 Resource Description Framework (RDF)

The Resource Description Framework (RDF, [29]) defines a set of W3C standards for the formal description of resources that are identified by URIs. RDF is a core component of the Semantic Web. Initially, it was designed to describe metadata on the World Wide Web (WWW) such as authors, copyrights, *etc.* of documents, however, applying a definition of the term *resource* beyond the WWW context, RDF is now also used to describe metadata of any URI-identifiable entity like cities, genes, *etc.*

2.3.1 Triples as a Data Structure

As outlined before, one of the main purposes of the Semantic Web is to give information a well-defined meaning. Using an example from Tim Berners-Lee's article [8], meaning can be to differentiate between the concepts of a shipping and a billing address, or the concept of an address in the sense of delivering a formal spoken communication to an audience. In order to assure the differences in meaning, things are identified by a Unique Resource Identifier (URI). The majority of the data processed by machines can be described by elementary sentences like *A cat is a mammal*, *Thomas Steiner is the author of this document*, or *Prince William is married to Kate Middleton*. Each of these sentences has a subject (*A cat*), a predicate (*is a*), and an object (*mammal*). Every subject, predicate, and object can be identified by a URI. This idea is very powerful, as it allows to express the same concept represented by a URI (for example, mammal by `http://dbpedia.org/resource/Mammal`) with different terms in different languages (like, for example, Säugetier, mammal, or nisäkkäät). Everyone can extend the set of concepts simply by creating a URI on the Web, which is exploited by RDF.

2.3.2 Important RDF Serialization Syntaxes

Knowledge or facts represented in the RDF triple data structure need to be serialized in order to be stored or transmitted over the Internet. Several serialization formats exist, each of which with its particular advantages and disadvantages, mostly around readability for human beings and parsability for machines. According to our experience, most people prefer the Turtle [35] format for its readability, whereas for machines, oftentimes RDF/XML [20] is the easiest to work with.

RDF Sample Graph

In the following, we will illustrate the various RDF serialization formats with an RDF sample graph inspired by a default example of the Apache Anything To Triples project (Any23, `http://any23.org/`, accessed July 15, 2013). It contains data about a fictive FOAF (Friend of a friend, [15]) person named *John X. Foobar* with an email address with the SHA1 (secure hash algorithm) checksum of *cef817456278b70cee8e5a1611539-ef9d928810e*. The actual email address is obscured to avoid spam emails. Figure 2.2 shows the graphical representation of this sample graph.

The RDF/XML Syntax

RDF/XML [20] was introduced by the W3C as the first RDF serialization syntax. In order to encode an RDF graph in XML, the nodes and predicates have to be represented in XML terms—element names, attribute names, element contents, and attribute values.

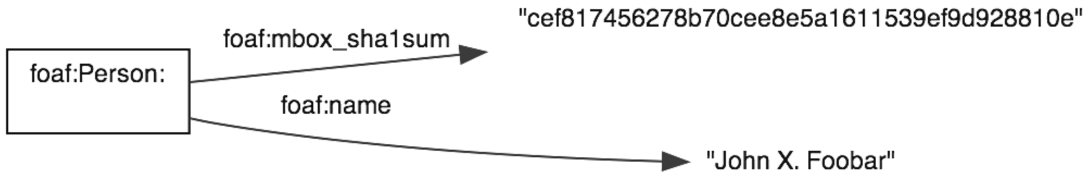


Figure 2.2: Sample RDF graph visualized

Albeit more human-friendly serialization formats such as Turtle [35] gain more and more traction, RDF/XML is still very wide-spread. Its media type is `application/rdf+xml`, the recommended file extension is `.rdf`, the encoding is UTF-8. Listing 2.3 shows the previously introduced sample graph serialized in RDF/XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description rdf:nodeID="node15urahancx74224">
    <rdf:type rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
    <foaf:name>John X. Foobar</foaf:name>
    <foaf:mbox_sha1sum>
      cef817456278b70cee8e5a1611539ef9d928810e
    </foaf:mbox_sha1sum>
  </rdf:Description>
</rdf:RDF>
```

Listing 2.3: Sample graph in RDF/XML syntax

The N-Triples Syntax

The N-Triples [25] syntax was primarily developed by Dave Beckett and Art Barstow. N-Triples is a subset of Turtle (see section 2.3.2), which in turn is a subset of Notation3 (see section 2.3.2). There are very few variations to express a graph in N-Triples, which makes it an ideal syntax for testing purposes, however, as it is missing some shortcuts of Turtle, it is quite verbose. Its media type is `text/plain`, the recommended file extension is `.nt`, and the encoding is 7-bit US-ASCII (and explicitly *not* UTF-8). Listing 2.4 shows the previously introduced sample graph serialized as N-Triples.

```
_:1 <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
    <http://xmlns.com/foaf/0.1/Person> .
_:1 <http://xmlns.com/foaf/0.1/name>
    "John X. Foobar" .
_:1 <http://xmlns.com/foaf/0.1/mbox_sha1sum>
    "cef817456278b70cee8e5a1611539ef9d928810e" .
```

Listing 2.4: Sample graph in N-Triples syntax

The Turtle Syntax

Turtle [35], or the Terse RDF Triple Language, was defined by Dave Beckett. It is a superset of N-Triples (see section 2.3.2) and a subset of Notation3 (see section 2.3.2). It has reached a *de facto* standard status, with the RDF Working Group publishing the new Turtle specification as a W3C Candidate Recommendation on February 19, 2013. Its media type is `text/turtle` (the sometimes still observable media type `application/x-turtle` is deprecated), the recommended file extension is `.ttl`, the encoding is UTF-8. Listing 2.5 shows the previously introduced sample graph serialized in Turtle syntax.

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

_:node15urahancx74223 a foaf:Person ;
  foaf:name "John X. Foobar" ;
  foaf:mbox_sha1sum "cef817456278b70cee8e5a1611539ef9d928810e" .
```

Listing 2.5: Sample graph in Turtle syntax, the syntax is equivalent to Listing 2.6

The Notation3 Syntax

Notation3 [6] was introduced by Tim Berners-Lee. Notation3 has some features that go beyond the pure expressiveness of RDF like rules, support for variables, and quantification. Its media type is `text/n3`, the recommended file extension is `.n3`, the encoding is always UTF-8. Listing 2.6 shows the previously introduced sample graph serialized in Notation3, which, given the present trivial example, is syntactically equal to Turtle.

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

_:node15urahancx74223 a foaf:Person ;
  foaf:name "John X. Foobar" ;
  foaf:mbox_sha1sum "cef817456278b70cee8e5a1611539ef9d928810e" .
```

Listing 2.6: Sample graph in Notation3 syntax

The RDFa Syntax

RDFa [1] has a special role in that it is a specification for attributes to express structured data in XHTML [34], but also in HTML4 and HTML5 [39]. It uses the rendered hypertext content of (X)HTML for the RDFa markup, so that data publishers can use the same document for human- and machine-readable content. The contained RDF triples can be extracted with distillers. In consequence, RDFa can be considered another serialization syntax for RDF, with the same expressive power as RDF/XML [20], Turtle [35], *etc.* Its media type is `application/xhtml+xml`, the recommended file extension is `.html`. RDFa shares some design goals with Microformats [17] and Microdata [28]. Where Microformats specify both a syntax for embedding structured data into HTML *and* a vocabulary of specific terms for each Microformat, RDFa in contrast *only* specifies a syntax, since the vocabularies it relies on are externally and independently specified. The essence of RDFa is a set of attributes that contain metadata about things, and that can be embedded in mark-up languages, for example in XHTML or HTML. The concrete attributes are as follows.

- `about` and `src` a URI or CURIE (compact URI) [9] that specifies the resource the metadata is about.
- `rel` specifies a relationship with another resource.
- `href` and `resource` specify the partner resource.
- `property` specifies a property for the content of an element.
- `content` optional attribute that overrides the content of the element when using the property attribute.

- `datatype` optional attribute that specifies the datatype of text specified for use with the property attribute.
- `typeof` optional attribute that specifies the RDF type(s) of the subject (the resource that the metadata is about).

An additional simplified subset of RDFa is RDFa Lite [38], which is aligned with Microdata. Listing 2.7 shows the previously introduced sample graph in RDFa.

```
<div about="_:1" typeof="http://xmlns.com/foaf/0.1/Person">
  <span property="http://xmlns.com/foaf/0.1/mbox_sha1sum">
    cef817456278b70cee8e5a1611539ef9d928810e
  </span>
  <span property="http://xmlns.com/foaf/0.1/name">
    John X. Foobar
  </span>
</div>
```

Listing 2.7: Sample graph in RDFa syntax

2.4 SPARQL: Semantic Web Query Language

SPARQL is a recursive acronym that stands for SPARQL Protocol and RDF Query Language. The SPARQL specification [36] defines the syntax and semantics of the SPARQL query language for RDF. SPARQL can be used to express queries across diverse data sources, whether the data is stored natively as RDF, or viewed as RDF via middleware. SPARQL allows for querying required and optional graph patterns along with their conjunctions and disjunctions. SPARQL also supports extensible value testing and constraining queries by source RDF graph. The results of SPARQL queries can be result sets or RDF graphs. SPARQL became an official W3C Recommendation in 2008. It was standardized by the RDF Data Access Working Group (DAWG).

2.4.1 The Vision of the Web as a Giant Single Database

The Web as we know it today is a *network of documents*, interconnected by hyperlinks that everyone can participate in by placing links to existing documents. The vision of the Semantic Web, however, is a *network of facts* about entities, interconnected by means of graphs of data. Where the Web of today is a graph of documents, the Semantic Web is envisioned to be a huge global graph, formed by many individual graphs. If one party publishes facts about an entity and a different party publishes different facts about the same entity, then the overall knowledge about that entity is represented in a decentralized way, accessible to all, and open for everyone to enrich. This requires strong globally unique identifiers, or at least ways to map one identifier to another.

Given the (visionary) huge global graph, a fictive SPARQL query like the one in Listing 2.8 could be used to get results from the graph, like the email addresses of every person in the world. SPARQL queries, unlike traditional databases, are not necessarily guaranteed to return *all* existing results (completeness).

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?email
WHERE {
  ?person a foaf:Person.
  ?person foaf:name ?name.
  ?person foaf:mbox ?email.
}
```

Listing 2.8: Fictive SPARQL query returning the names and email addresses of every person in the world (`http://en.wikipedia.org/wiki/SPARQL#Benefits`, accessed July 15, 2013)

This query selects the names and email addresses from all persons who have facts about them in the global graph. The query starts with a prefix definition, and then constrains the results to be of type `foaf:Person` [15], whose name and email address are the values of the triples with the predicates `foaf:name` and `foaf:mbox` respectively. However, in practice SPARQL endpoints like the DBpedia SPARQL endpoint[1] typically only allow for querying a local graph for performance reasons.

2.4.2 Different SPARQL Query Variations

The SPARQL Query Language currently specifies four different query variants, which we will list in the following. Each query variant is accompanied by a basic example query with the particular result.

[1] `http://dbpedia.org/sparql`, accessed July 15, 2013

SELECT

The SELECT query variant is used to extract raw values from a SPARQL endpoint. The results are returned in a tabular format. A sample query was given in Listing 2.8.

DESCRIBE

The DESCRIBE query variant is used to extract an RDF graph from the SPARQL endpoint, the contents of which is left to the endpoint to decide based on what the maintainer deems as useful information. An example query is given below:

```
DESCRIBE <http://example.org/sparql>
```

ASK

The ASK query variant is used to provide a simple true or false result for a query on a SPARQL endpoint. No information is returned about possible solutions, just whether or not a solution exists. An example query with a sample response is given below:

Given the RDF graph in Figure 2.2 and the following SPARQL ASK query:

```
PREFIX foaf:  <http://xmlns.com/foaf/0.1/>
ASK { ?x foaf:name "Alice" }
```

This query creates a negative response, as there is no person named Alice in the graph:

```
no
```

CONSTRUCT

The CONSTRUCT query variant is used to extract information from the SPARQL endpoint and to transform the results into valid RDF specified by a graph template. The result is an RDF graph formed by taking each query solution in the solution sequence, substituting for the variables in the graph template, and combining the resulting triples into a single RDF graph by set union. If any such instantiation produces a triple containing an unbound variable or an illegal RDF construct, then that triple is not included in the output RDF graph. An example query is given below.

Given the following RDF graph, serialized in Turtle syntax:

```
@prefix foaf:  <http://xmlns.com/foaf/0.1/> .
_:a foaf:name "Alice" .
_:a foaf:mbox <mailto:alice@example.org> .
```

Given the following SPARQL `CONSTRUCT` query:

```
PREFIX foaf:  <http://xmlns.com/foaf/0.1/>
PREFIX vcard:  <http://www.w3.org/2001/vcard-rdf/3.0#>
CONSTRUCT { <http://example.org/person#Alice> vcard:FN ?name }
WHERE  ?x foaf:name ?name
```

This query creates the following `vcard` [21] properties from the FOAF information:

```
@prefix vcard:  <http://www.w3.org/2001/vcard-rdf/3.0#> .
<http://example.org/person#Alice> vcard:FN "Alice" .
```

2.5 Linked Data

Linked Data [4] defines a set of agreed-on best practices and principles for interconnecting and publishing structured data on the Web. It uses Web technologies like the Hypertext Transfer Protocol (HTTP , [24]) and Unique Resource Identifiers (URIs [7]) to create typed links between different sources. The portal `http://linkeddata.org/` (accessed July 15, 2013) defines Linked Data as being *"about using the Web to connect related data that wasn't previously linked, or using the Web to lower the barriers to linking data currently linked using other methods."*

2.5.1 The Linked Data Principles

Tim Berners-Lee defined the four rules for Linked Data in a W3C Design Issue [4] published in 2006 as follows:

1. Use URIs as names for things.

2. Use HTTP URIs so that people can look up those names.

3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).

4. Include links to other URIs, so that they can discover more things.

Linked Data uses RDF [29] to create typed links between things in the world. The result is oftentimes referred to as the *Web of Data*. As outlined before, RDF encodes statements about things in the form of `(subject, predicate, object)` triples. If subject and object have URIs from different namespaces, Bizer *et al.* speak of *RDF links* in [27]. An exemplary RDF link adapted from [11] stating that a description of the movie Pulp Fiction from the Linked Movie Database [26] and a description from DBpedia [2] are indeed talking about the same movie can be seen in Listing 2.9.

```
<http://data.linkedmdb.org/resource/film/77> ↩
  <http://www.w3.org/2002/07/owl#sameAs> ↩
  <http://dbpedia.org/page/Pulp_Fiction> .
```

Listing 2.9: Exemplary RDF link stating that a description of the movie Pulp Fiction from the Linked Movie Database [26] and a description from DBpedia are indeed talking about the same movie

2.5.2 The Linking Open Data Cloud Diagram

The Linking Open Data (LOD) cloud diagram [19] is a visualization effort that shows datasets that have been published in Linked Data [4] format by contributors to the Linking Open Data community project and other individuals and organizations. The objective is to identify existing datasets with open licenses, convert them to RDF whilst obeying the Linked Data principles, and finally publish them on the Web. Due to its open structure, everyone can contribute to the project by publishing a dataset and interlinking it to existing datasets. Today, the project includes datasets of major organizations such as the BBC, Thomson Reuters, or the Library of Congress to name just a few. The state of the LOD cloud has been examined in [10]. The latest LOD cloud diagram as of September 2011 can be seen in Figure 2.3.

2.6 Conclusions

In this chapter, we have first introduced the Semantic Web and compared it to the non-semantic Web. We have shown how structured data in the form of tables is a first step towards richer semantics. An example of converting structured data from Wikipedia into machine-readable data is the knowledge base DBpedia. Further, we have looked at the intrinsic semantics of HTML in versions 4 and 5, and how through additional attributes even richer semantics can be added by the annotation formats Microdata and Microformats. We have introduced the Resource Description Format (RDF) and its different serializations. On top of RDF, we have detailed how the Semantic Web query language SPARQL can be used to express queries across data sources. Finally, we have shown how data on the Web can be exposed as so-called Linked Data, an effort which is visualized in the Linking Open Data cloud. By introducing these Semantic Web technologies, we have set the foundations for the coming chapters that build upon those basic pillars.

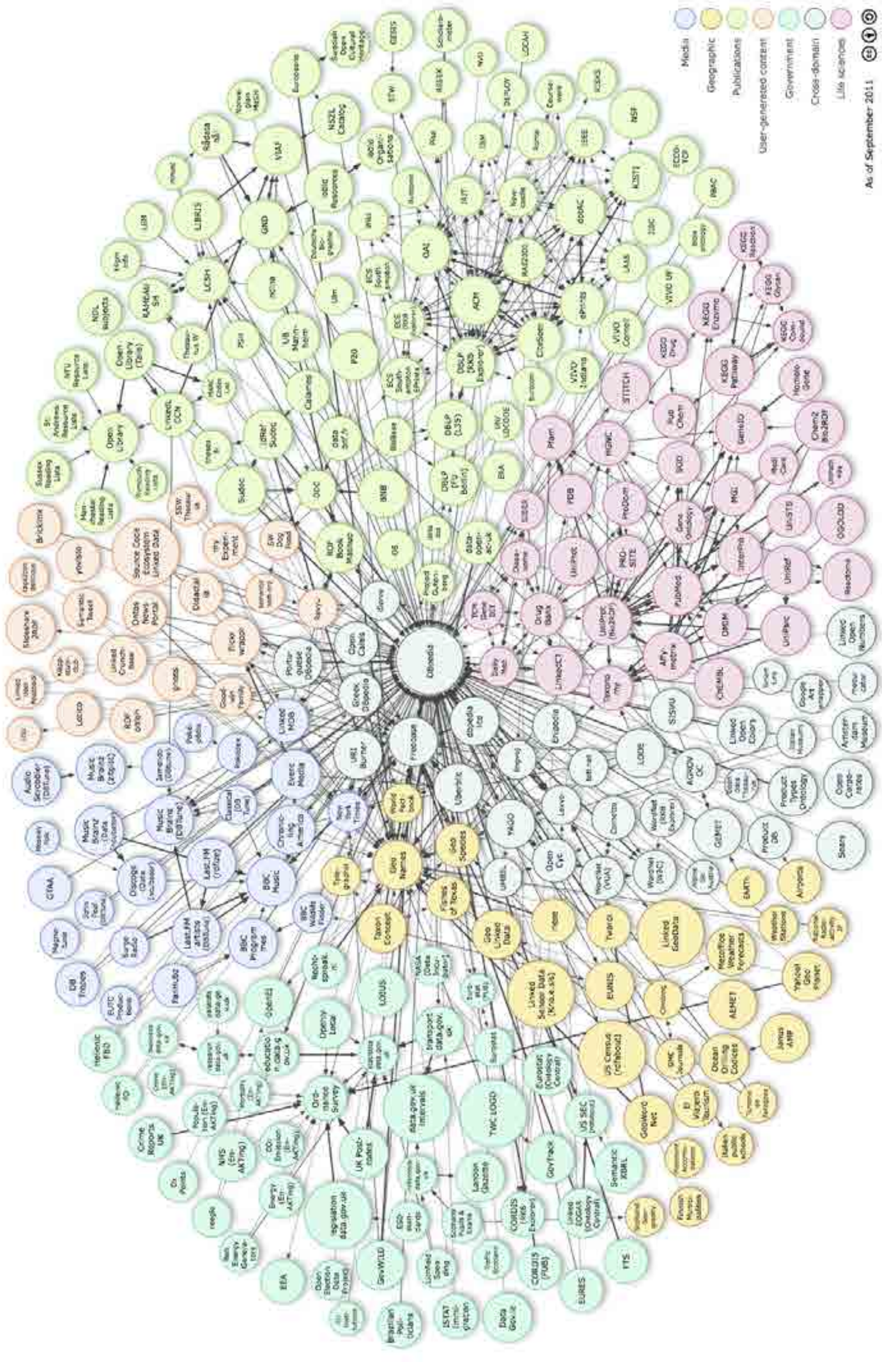


Figure 2.3: Linking Open Data cloud diagram as of September 2011, by Richard Cyganiak and Anja Jentzsch `http://lod-cloud.net/` (accessed July 15, 2013)

Chapter Notes

This chapter is partly based on the following publications.

- Thomas Steiner, Raphaël Troncy, and Michael Hausenblas. "How Google is using Linked Data Today and Vision For Tomorrow". In: *Proceedings of the Workshop on Linked Data in the Future Internet at the Future Internet Assembly, Ghent 16–17 Dec 2010.* Ed. by Sören Auer, Stefan Decker, and Manfred Hauswirth. Vol. 700. CEUR Workshop Proceedings ISSN 1613-0073. Dec. 2010. URL: http://CEUR-WS.org/Vol-700/Paper5.pdf.

- Thomas Steiner. "DC Proposal: Enriching Unstructured Media Content About Events to Enable Semi-Automated Summaries, Compilations, and Improved Search by Leveraging Social Networks". In: *Proceedings of the 10th International Conference on The Semantic Web – Volume Part II.* ISWC' 11. Bonn, Germany: Springer-Verlag, 2011, pp. 365–372. ISBN: 978-3-642-25092-7. URL: http://iswc2011.semanticweb.org/fileadmin/iswc/Papers/DC_Proposals/70320369.pdf.

References

[1] Ben Adida, Mark Birbeck, Shane McCarron, and Ivan Herman. *RDFa Core 1.1: Syntax and processing rules for embedding RDF through attributes.* Recommendation. W3C, 2012.

[2] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. "DBpedia: a Nucleus for a Web of Open Data". In: *The Semantic Web: 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference.* ISWC 2007 + ASWC 2007. Springer, 2007, pp. 722–735.

[3] Robin Berjon, Travis Leithead, Erika Doyle Navara, Edward O'Connor, and Silvia Pfeiffer. *HTML5: a vocabulary and associated APIs for HTML and XHTML.* Working Draft. W3C, 2012.

[4] Tim Berners-Lee. *Linked Data.* `http://www.w3.org/DesignIssues/LinkedData.html`, accessed July 15, 2013. 2006.

[5] Tim Berners-Lee, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen, and Arthur Secret. "The World-Wide Web". In: *Communications of the ACM* 37.8 (1994), pp. 76–82.

[6] Tim Berners-Lee and Dan Connolly. *Notation3 (N3): a readable RDF syntax.* Team Submission. W3C, 2011.

[7] Tim Berners-Lee, Roy T. Fielding, and Larry Masinter. *Uniform Resource Identifier (URI): Generic Syntax.* RFC 3986. IETF, 2005.

[8] Tim Berners-Lee, James Hendler, and Ora Lassila. "The Semantic Web". In: *Scientific American* (2001). `http://www.sciam.com/article.cfm?id=the-semantic-web&print=true`, accessed July 15, 2013.

[9] Mark Birbeck and Shane McCarron. *CURIE Syntax 1.0, a Syntax for expressing Compact URIs.* Working Draft. W3C, 2007.

[10] Chris Bizer, Anja Jentzsch, and Richard Cyganiak. *State of the LOD Cloud.* `http://wifo5-03.informatik.uni-mannheim.de/lodcloud/state/`, accessed July 15, 2013. 2011.

[11] Christian Bizer, Tom Heath, and Tim Berners-Lee. "Linked Data – The Story So Far". In: *International Journal On Semantic Web and Information Systems* 5.3 (2009), pp. 1–22.

[12] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. "Freebase: a collaboratively created graph database for structuring human knowledge". In: *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. SIGMOD '08. Vancouver, Canada: ACM, 2008, pp. 1247–1250. ISBN: 978-1-60558-102-6. DOI: 10.1145/1376616.1376746. URL: http://doi.acm.org/10.1145/1376616.1376746.

[13] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, and François Yergeau. *Extensible Markup Language (XML) 1.0 (Fifth Edition)*. Recommendation. W3C, 2008.

[14] Michel Bréal. *Essai de Sémantique : science des significations*. Paris : Hachette, 1897.

[15] Dan Brickley and Libby Miller. *FOAF Vocabulary Specification 0.98*. Namespace Document. http://xmlns.com/foaf/spec/, accessed July 15, 2013. 2010.

[16] Rogers Cadenhead, Sterling Camden, Simone Carletti, James Holderness, Jenny Levine, Eric Lunt, et al. *RSS 2.0 Specification*. http://www.rssboard.org/rss-specification, accessed July 15, 2013. 2006.

[17] Tantek Çelik, Kevin Marks, Eric Meyer, Matthew Mullenweg, Mark Pilgrim, and Morten W. Petersen. *Microformats*. http://microformats.org, accessed July 15, 2013. 2006.

[18] Douglas Crockford. *Introducing JSON*. http://json.org/, accessed July 15, 2013. 2006.

[19] Richard Cyganiak and Anja Jentzsch. *The Linking Open Data cloud diagram*. http://lod-cloud.net/, accessed July 15, 2013. 2011.

[20] Brian McBride Dave Beckett. *RDF/XML Syntax Specification (Revised)*. Recommendation. W3C, 2004.

[21] F. Dawson and T. Howes. *vCard MIME Directory Profile*. RFC 2426. IETF, 1998.

[22] F. Dawson and D. Stenerson. *Internet Calendaring and Scheduling Core Object Specification (iCalendar)*. RFC 2445. IETF, 1998.

[23] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Language, Speech and Communication Series. Cambridge, MA: MIT Press, 1998.

[24] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, et al. *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616. IETF, 1999.

[25] Jan Grant, Dave Beckett, and Brian McBride. *RDF Test Cases*. Recommendation. W3C, 2004.

[26] O. Hassanzadeh and M. P. Consens. "Linked Movie Data Base". In: *Proceedings of the Linked Data on the Web Workshop (LDOW2009)*. Madrid, Spain, 2009.

[27] T. Heath and C. Bizer. *Linked Data: Evolving the Web into a Global Data Space*. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool, 2011.

[28] Ian Hickson. *HTML Microdata*. Working Draft. W3C, 2012.

[29] Graham Klyne and Jeremy J. Carroll. *Resource Description Framework (RDF): Concepts and Abstract Syntax*. Recommendation. W3C, 2004.

[30] Douglas B. Lenat. "CYC: a Large-scale Investment In Knowledge Infrastructure". In: *Communications of the ACM* 38.11 (Nov. 1995), pp. 33–38.

[31] John Markoff. *Start-Up Aims for Database to Automate Web Searching*. `http://www.nytimes.com/2007/03/09/technology/09data.html`, accessed July 15, 2013. 2007.

[32] George A. Miller. "WordNet: a Lexical Database for English". In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[33] M. Nottingham and R. Sayre. *The Atom Syndication Format*. RFC 4287. IETF, 2005.

[34] Steven Pemberton, Daniel Austin, Jonny Axelsson, Tantek Çelik, et al. *XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition), a Reformulation of HTML 4 in XML 1.0*. Recommendation. W3C, 2000.

[35] E. Prud'hommeaux, G. Carothers, D. Beckett, and T. Berners-Lee. *Turtle – Terse RDF Triple Language*. Candidate Recommendation. `http://www.w3.org/TR/turtle/`, accessed July 15, 2013. W3C, 2013.

[36] Eric Prud'hommeaux and Andy Seaborne. *SPARQL Query Language for RDF*. Recommendation. W3C, 2008.

[37] Dave Raggett, Arnaud Le Hors, and Ian Jacobs. *HTML 4.01 Specification*. Recommendation. W3C, 1999.

[38] Manu Sporny. *RDFa Lite 1.1*. Recommendation. W3C, June 2012.

[39] Manu Sporny, Shane McCarron, Ben Adida, Mark Birbeck, and Steven Pemberton. *HTML+RDFa 1.1: Support for RDFa in HTML4 and HTML5*. Working Draft. W3C, 2012.

[40] Thomas Steiner. “DC Proposal: Enriching Unstructured Media Content About Events to Enable Semi-Automated Summaries, Compilations, and Improved Search by Leveraging Social Networks”. In: *Proceedings of the 10th International Conference on The Semantic Web – Volume Part II.* ISWC’ 11. Bonn, Germany: Springer-Verlag, 2011, pp. 365–372. ISBN: 978-3-642-25092-7. URL: `http://iswc2011.semanticweb.org/fileadmin/iswc/Papers/DC_Proposals/70320369.pdf`.

[41] Thomas Steiner, Raphaël Troncy, and Michael Hausenblas. “How Google is using Linked Data Today and Vision For Tomorrow”. In: *Proceedings of the Workshop on Linked Data in the Future Internet at the Future Internet Assembly, Ghent 16–17 Dec 2010.* Ed. by Sören Auer, Stefan Decker, and Manfred Hauswirth. Vol. 700. CEUR Workshop Proceedings ISSN 1613-0073. Dec. 2010. URL: `http://CEUR-WS.org/Vol-700/Paper5.pdf`.

[42] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. “YAGO: a Core of Semantic Knowledge”. In: *Proceedings of the 16th International Conference on World Wide Web.* WWW ’07. New York, NY, USA: ACM, 2007, pp. 697–706.

3

Social Networks

From the first ever email to video calls on the go, the Internet has always been about communication. Historically, communities formed around Usenet mailing lists or Bulletin Board Systems. Starting from the early eighties, often around all sorts of topics like fine arts, literature, and philosophy (*e.g.*, `humanities.classics` or `humanities.-design.misc`). Then, starting from the late eighties, Internet Relay Chats (IRC) allowed people to communicate interactively and in realtime, organized in channels (*e.g.*, `#linux`). Starting from the nineties, blogs began to spread, reaching mainstream popularity somewhere in mid-2000. While the early social communities where created entirely *ad hoc* whenever someone logged in to a system, the first social networks, among them `http://sixdegrees.com/` in 1997, allowed people to maintain a public profile with a list of connections (*friends*) that others could browse. In [1], boyd (*sic*[1]) and Ellison define the term *social network site (SNS)* as follows.

> *"We define social network sites as web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site.*
>
> *While we use the term 'social network site´ to describe this phenomenon, the term 'social networking sites´ also appears in public discourse, and the two terms are often used interchangeably."*

[1] `http://www.danah.org/name.html`, accessed July 15, 2013

Literature on social networks typically uses the term SNS. However, in order to differentiate ourselves from the therein defined for our purposes overly strict idea of social network, we decided to avoid the term altogether in favor of a more open definition of social network, which we detail in the following.

3.1 Definition of Terms Used in this Thesis

In this section, we define the terms that we will use throughout this thesis in order to avoid any ambiguity. In particular, we highlight that social networks have different levels of support for media items.

Social Network: A social network is an online service or media platform that focuses on building and reflecting relationships among people who share common interests and/or activities.

Media Item: A media item is defined as a photo[1] or video file that is publicly shared or published on at least one social network.

Micropost: A micropost is defined as a textual status message on a social network that can optionally be accompanied by a media item.

Hashtag The `#` symbol, called a hashtag, is used to mark keywords or topics in a micropost. It was created organically by Twitter users as a way to categorize messages. People use the hashtag symbol `#` before a relevant keyword or phrase (no spaces) in microposts to categorize them and help them show more easily in search.[2]

The boundary between *social networks* and *media platforms* is blurred. Several media sharing platforms, *e.g.*, YouTube (`http://youtube.com/`) enable people to upload content and optionally allow other people to react to this content in the form of comments, likes, or dislikes. On other social networks, *e.g.*, Facebook (`http://facebook.com/`) users can update their status, post links to stories, upload media content, and also give readers the option to react. Finally, there are hybrid clients, *e.g.*, the application TweetDeck (`http://www.tweetdeck.com/`) released by Twitter together with the media hosting platform Twitpic (`http://twitpic.com/`), where social networks integrate with media platforms, typically via third-party applications.

[1] We choose the term *photo* over the term *image* as Facebook, Twitter, and Google+ use it.

[2] Definition adapted from `https://support.twitter.com/articles/49309`, accessed July 15, 2013

3.2 Description of Popular Social Networks

In this section, we introduce several social networks and some of their key features that are relevant for our research. As we treat all networks the same—independent from their not always publicly known user population—they are listed in alphabetic order. For active participation, all social networks require users to be logged in. In the description below, we thus assume a logged in user.

3.2.1 Facebook

Facebook (`http://www.facebook.com/`) is a social networking service launched in February 2004, operated and owned by the American multinational Internet corporation Facebook, Inc. At time of writing, Facebook is the most popular social network with one billion monthly active users[1] as of October 2012. Facebook has native photo and video support, allowing people to upload an unlimited amount of media items. Photos and videos can also be recorded *ad hoc* via webcam. People can *Like* content via a designated Like button that can also be embedded on other websites. Initially, the button was called the *Awesome* button, but eventually[2] got rebranded to its current form. Individual microposts can also be shared. Facebook has a bidirectional relationship model (friend model) with an optional unidirectional relationship model (follow model), typically for following celebrities, remote friends, *etc.*

3.2.2 Flickr

Flickr (`http://www.flickr.com/`) is a photo and video hosting online community created by Ludicorp in 2004 and acquired by Yahoo! in 2005. All users can upload up to one Terabyte of photos or videos to the service. As of May 2013, the former account types (Free or Pro) are no longer available.[3] People can *Favorite* photos they like via a designated Favorite button. Flickr has a unidirectional relationship model (follow model), however, also allows people to mark other users as friends or family *without* the other party having to confirm. Following an urgent plea from Flickr users[4] that went

[1] `http://newsroom.fb.com/Key-Facts`, accessed July 15, 2013

[2] `http://www.quora.com/Facebook-Inc-company/Whats-the-history-of-the-Awesome-Button-that-eventually-became-the-Like-button-on-Facebook`, accessed July 15, 2013

[3] `http://blog.flickr.net/en/2013/05/20/a-better-brighter-flickr/`, accessed July 15, 2013

[4] `http://dearmarissamayer.com/`, accessed July 15, 2013

viral under the hashtag #dearmarissamayer where users complained that Yahoo! had semi-abandoned the service for too long, Flickr has now been revived under the new Yahoo! CEO Marissa Mayer.[1] The urgent plea website has since been updated with a "thank you" notice.

3.2.3 Google+

Google+ (http://google.com/+), sometimes transcribed as Google Plus and abbreviated as G+, is Google's social network. It was opened to the general public on September 20, 2011. Google+ has native photo support. Photos can either be manually uploaded when authoring a new micropost, or be automatically uploaded via the Google+ mobile application. External videos, for example, from the also Google-owned online video platform YouTube, but also from other services, get displayed in an inline view so that they can be viewed directly on the website. However, the network also allows for videos to be uploaded directly, or to be recorded *ad hoc* via webcam. People can *+1* (pronounced like a verb "to plus-one") content they like via a designated +1 button that can also be embedded on other websites. Individual microposts can also be shared. Google+ has a unidirectional relationship model (follow model).

3.2.4 Img.ly

Img.ly (http://img.ly/) is a photo hosting service operated by 9elements GmbH that was founded in 2009. It integrates deeply with Twitter, however, can also be used independently. Img.ly integrates with Twitter's *Tweet* button. The service has no own relationship model, but uses a user's social graph on Twitter.

3.2.5 Imgur

Imgur (http://imgur.com/) is a photo hosting service founded by Alan Schaaf in February 2009. While the service is deeply integrated with Twitter and Facebook, it can be used independently as well. Imgur integrates with all major social networks, and also has designated *Like* and *Dislike* buttons. The service has no own relationship model, but uses a user's social graph on Facebook.

[1] http://www.flickr.com/dearinternet, accessed July 15, 2013

3.2.6 Instagram

Instagram (`http://instagram.com/`) is a mobile photo and (since June 2013) video sharing application that was acquired by Facebook in April 2012. The application allows users to apply filters to photos. These photos can then be shared on external social networks like Facebook, Twitter, or Google+, and are also visible on Instagram's own social network. The service launched in October 2010. Instagram has native photo and video support, where its level of video support is comparable to Vine's. People can *Like* content via a designated Like button from within the Instagram application. Instagram has a unidirectional relationship model (follow model).

3.2.7 Lockerz

Lockerz (`http://lockerz.com/`) is an international social commerce website based in Seattle, WA. In 2011, Lockerz acquired the photo sharing service Plixi, which was formerly known as TweetPhoto. Lockerz keeps Plixi's service as a media platform running under the new Lockerz branding. While the service is deeply integrated with Twitter, it can be used independently as well. People can *Love* content they like via a designated Love button, but the service is also integrated with all major social networks. Since April 2012, the service no longer offers or supports photo-sharing services for developers and third-party applications.

3.2.8 MobyPicture

MobyPicture (`http://www.mobypicture.com/`) is a mobile messaging service owned by entrepreneur Mathys van Abbe. Users of the service can upload an unlimited number of photos and videos to the service. MobyPicture integrates with a number of third-party social networks. The service natively supports videos and photos, which can either be uploaded, or be recorded *ad hoc* via webcam. People can *Favorite* content they like via a designated Favorite button, however, the service also integrates with Google's *+1* button and Twitter's *Tweet* button. MobyPicture has a unidirectional relationship model (follower model).

3.2.9 Myspace

Myspace (`http://www.myspace.com/`), formerly MySpace and My_ _ _ _ _ (*sic*), is a social networking service owned by Specific Media LLC and pop star Justin Timberlake. The social network launched in August 2003. Once the most visited website in the United States in June 2006, the network's importance has steadily declined since. Instead of as a social networking website, Myspace has attempted to redefine itself as a social entertainment website, putting more focus on music, movies, celebrities, and TV. As such, Myspace has native photo, video, and, via special musician profiles, audio support. Videos can either be uploaded, or be recorded *ad hoc* via webcam. In January 2012, a rebranding strategy to Myspace TV in collaboration with Panasonic was unveiled with an exclusive focus on social TV that would allow people to watch and comment on videos. The latest reinvention of the service was launched on June 12, 2013.[1] People can *Like* certain content via a designated Like link. Myspace has a bidirectional relationship model (friend model) with an optional unidirectional relationship model (follow model), typically meant for following celebrities.

3.2.10 Photobucket

Photobucket (`http://photobucket.com/`) is a photo and video hosting service founded in 2003 by Alex Welch and Darren Crystal. It was acquired by Fox Interactive Media in 2007. In June 2011, Twitter announced an exclusive partnership with Photobucket that made the service the default photo sharing platform for Twitter, used for its native media item support. Since then, in December 2012, Twitter has rolled out its own photo storage solution.[2]

3.2.11 Twitpic

Twitpic (`http://twitpic.com/`) is a service that allows users to upload photos and videos. It optionally integrates with Twitter. Twitpic was launched in 2008 by Noah Everett. While Twitpic can be used independently from Twitter, the integration is

[1] `http://www.cbc.ca/news/yourcommunity/2013/06/myspaces-20m-relaunch-deletes-its-remaining-users-blogs.html`, accessed July 15, 2013

[2] `https://blog.twitter.com/2012/blobstore-twitter's-house-photo-storage-system`, accessed July 15, 2013

made easy with Twitpic usernames and passwords being the same as the ones on Twitter. Twitpic integrates with Twitter via the *Tweet* button. The service has no own relationship model, but uses a user's social graph on Twitter.

3.2.12 Twitter

Twitter (`http://twitter.com/`) is an online social networking service and microblogging service that enables its users to send and read microposts of up to 140 characters. These microposts are referred to as *tweets*. Twitter was founded in March 2006 by Jack Dorsey and launched to the public in July 2006. The website is ranked among the top-10 websites globally by the Web information company Alexa.[1] As of August 2011, Twitter has native photo support, which allows users to upload photos to the service. However, at time of writing, it is not possible to record photos or videos *ad hoc* via webcam. Videos are not supported natively, however, likewise the situation with photos before (and also in part still today), an ecosystem of media platforms takes care of hosting media items on behalf of Twitter users. These third-party-hosted media items can be linked to from within tweets. In October 2012, Twitter acquired Vine, a mobile app that enables its users to create and post six seconds long video clips. People can *ReTweet* content they like either via a designated ReTweet button, or—following the prior, but still widely popular manual ReTweet convention—by quoting a Twitter user by prepending "RT @username:" in front of the original tweet. In addition to that, Twitter offers a *Tweet* button that can be embedded on other websites. Twitter has a unidirectional relationship model (follow model).

3.2.13 Yfrog

Yfrog (`http://yfrog.com/`) is a photo and video hosting service run by ImageShack that was launched in February 2009. While the service is deeply integrated with Twitter, it can be used independently as well. Yfrog integrates with Twitter's *Tweet* button. The service has no own relationship model, but uses a user's social graph on Twitter.

[1] `http://www.alexa.com/topsites`, accessed July 15, 2013

3.2.14 YouTube

YouTube (`http://www.youtube.com/`) is a video sharing website founded in February 2005. In November 2006, YouTube was acquired by Google and now operates as a subsidiary of the company. It allows people to upload, view, and share an unlimited number of videos. YouTube has native video support, but does not support photos. Videos can be uploaded, or be recorded *ad hoc* via webcam. People can *Like* or *Dislike* content via designated Like or Dislike buttons. YouTube has a unidirectional relationship model (follow model).

3.3 Decentralized Social Networks

All social networks presented up to now are centralized networks. In contrast, *distributed*, or also referred to as *decentralized social networks*, are social network services that are decentralized and distributed across different providers, with a special focus on portability, interoperability, and federation capability, *i.e.*, an agreement upon standards of operation in a collective fashion. Decentralized, protocol-based systems offer users a choice of providers, which means that if one provider should terminate their service, the user is free to take out her data and start where she left off with a different provider. As a final aspect, governments cannot effectively censor decentralized social networks, as this would be impracticable due to the distributedness of user data. None of the decentralized social networks could reach a critical mass of users and/or network activity as of yet. We will therefore not consider them for this thesis.

In the following, we will list representative efforts in the direction of truly decentralized social networks. This list is not meant to be complete, but covers the efforts that received the most media attention in the years 2011 to 2013.

StatusNet: A first example of decentralized social network software providers is StatusNet (`http://status.net/`), which provides an open-source implementation of the OStatus[1] open standard, most prominently deployed at `http://identi.ca/`.

[1] `http://gitorious.org/projects/ostatus/`, accessed July 15, 2013

The DIASPORA* Project: A second example is the DIASPORA* project (`http://diasporaproject.org/`), which provides a free and open-source personal Web server component referred to as *pod* that allows participants in the project to form nodes that span the distributed Diaspora social network.

Tent: Third, there is Tent™ (`https://tent.io/`). Tent is an open-source protocol for distributed social networking and personal data storage. Anyone can run a Tent server or write an app or alternative server implementation that uses the Tent protocol. Users can take their content and relationships with them when they change or move servers. Tent supports extensible data types, so developers can create new kinds of interactions. Rather than running an own server, users can also rely on Tent.is (`https://tent.is/`), a service which hosts Tent servers and basic applications for users. At time of writing, the global site feed[1], suggests that the service is not very actively used.

3.4 Classification of Social Networks

As motivated in section 3.1, different social networks have varying support for media items, ranging from native support in media-centric social networks to optional support in micropost-centric social networks. In order to differentiate social networks by their media item support level, we introduce a classification of social networks as follows.

- *First-order support*: The social network is centered around media items and posting requires the inclusion of a media item (*e.g.*, YouTube, Flickr).
- *Second-order support*: The social network lets users upload media items, but it is also possible to post purely textual messages (*e.g.*, Facebook).
- *Third-order support*: The social network has no direct support for media items, but relies on third-party media platforms to host media items, which are linked to the status update (*e.g.*, Twitter relying on third-party video hosting via Twitpic).

In this chapter, we consider 11 different social networks that represent all together most of the market share of the Western world. The criteria for inclusion follow a study [4] performed by the company Sysomos, specialized in social media monitoring

[1] `https://app.tent.is/global`, accessed July 15, 2013

and analytics. Table 3.1 lists the considered social networks according to the categorization defined above. Due to language barriers, we had to omit popular Chinese social networks such as Sina Weibo (`http://www.weibo.com/`) with more than 500 million registered users,[1] Tencent Weibo (`http://t.qq.com/`), Renren (`http://www.renren.com/`) with 31 million active users,[2] and Kaixin001 (`http://www.kaixin001.com/`).

3.5 Conclusions

Alongside the Semantic Web technologies that were introduced in the previous chapter, social networking sites form the backbone of this thesis. In this chapter, we have thus first defined the terms of *social network*, *micropost*, *media platform*, and *media item*. Subsequently, we have introduced and described in detail the most popular social networking sites and media platforms. Different social networking sites have a different level of support for media items. We have therefore classified the social networking site landscape accordingly. In the upcoming chapters, we will get to the heart of micropost annotation, breaking news event detection, media item extraction from microposts, followed by media item deduplication, clustering, and ranking. Finally, we will close the core part of the thesis with media item compilation.

[1] `http://thenextweb.com/asia/2013/02/21/chinas-sina-weibo-grew-73-in-2012-passing-500-million-registered-accounts/`, accessed July 15, 2013

[2] `http://online.wsj.com/article/SB10001424052748704729304576286903217555660.html#ixzz1KqsoJPb8`, accessed July 15, 2013

Social Network	URL	Category	Comment
Facebook	`http://facebook.com`	second-order	Media item links are returned via the Facebook API.
Google+	`http://google.com/+`	second-order	Media item links are returned via the Google+ API.
Myspace	`http://myspace.com`	second-order	Media item links are returned via the Myspace API.
Twitter	`http://twitter.com`	second-/third-order	In second order mode, media item links are returned via the Twitter API. In third order mode, Web scraping or additional media platform API usage are necessary to retrieve media item links. Many people still use Twitter in third order mode.
Flickr	`http://flickr.com`	first-order	Media item links are returned via the Flickr API.
Img.ly	`http://img.ly`	first-order	Media platform for Twitter. Media item link must be retrieved via Web scraping.
Instagram	`http://instagram.com`	first-order	Media item links are returned via the Instagram API.
MobyPicture	`http://mobypicture.com`	first-order	Media platform for Twitter. Media item links are returned via the MobyPicture API.
Twitpic	`http://twitpic.com`	first-order	Media platform for Twitter. Media item links are returned via the Twitpic API.
Yfrog	`http://yfrog.com`	first-order	Media platform for Twitter. Media item links must be retrieved via Web scraping.
YouTube	`http://youtube.com`	first-order	Media item links are returned via the YouTube API.

Table 3.1: 11 social networks with different level of support for media items and techniques needed to retrieve them

Chapter Notes

This chapter is partly based on the following publications.

- Giuseppe Rizzo, Thomas Steiner, Raphaël Troncy, Ruben Verborgh, José Luis Redondo García, and Rik Van de Walle. "What Fresh Media Are You Looking For?: Retrieving Media Items From Multiple Social Networks". In: *Proceedings of the 2012 International Workshop on Socially-aware Multimedia*. SAM '12. Nara, Japan: ACM, 2012, pp. 15–20. ISBN: 978-1-4503-1586-9. URL: `http://www.eurecom.fr/~troncy/Publications/Troncy-saw12.pdf`.

- Houda Khrouf, Ghislain Atemezing, Giuseppe Rizzo, Raphaël Troncy, and Thomas Steiner. "Aggregating Social Media for Enhancing Conference Experience". In: *Real-Time Analysis And Mining of Social Streams, Papers from the 2012 ICWSM Workshop*. Ed. by Arkaitz Zubiaga, Maarten de Rijke, Markus Strohmaier, and Mor Naaman. AAAI Technical Report WS-12–02. June 2012. URL: `http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/download/4779/5086`.

- Houda Khrouf, Ghislain Atemezing, Thomas Steiner, Giuseppe Rizzo, and Raphaël Troncy. *Confomaton: A Conference Enhancer with Social Media from the Cloud.* 2012. URL: `http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_343.pdf`.

References

[1] danah m. boyd and Nicole B. Ellison. “Social Network Sites: Definition, History, and Scholarship”. In: *Journal of Computer-Mediated Communication* 13.1 (2007), pp. 210–230.

[2] Houda Khrouf, Ghislain Atemezing, Giuseppe Rizzo, Raphaël Troncy, and Thomas Steiner. “Aggregating Social Media for Enhancing Conference Experience”. In: *Real-Time Analysis And Mining of Social Streams, Papers from the 2012 ICWSM Workshop*. Ed. by Arkaitz Zubiaga, Maarten de Rijke, Markus Strohmaier, and Mor Naaman. AAAI Technical Report WS-12–02. June 2012. URL: `http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/download/4779/5086`.

[3] Houda Khrouf, Ghislain Atemezing, Thomas Steiner, Giuseppe Rizzo, and Raphaël Troncy. *Confomaton: A Conference Enhancer with Social Media from the Cloud.* 2012. URL: `http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_343.pdf`.

[4] Sheldon Levine. *How People Currently Share Pictures On Twitter.* `http://blog.sysomos.com/2011/06/02/`, accessed July 15, 2013. 2011.

[5] Giuseppe Rizzo, Thomas Steiner, Raphaël Troncy, Ruben Verborgh, José Luis Redondo García, and Rik Van de Walle. “What Fresh Media Are You Looking For?: Retrieving Media Items From Multiple Social Networks”. In: *Proceedings of the 2012 International Workshop on Socially-aware Multimedia*. SAM ’12. Nara, Japan: ACM, 2012, pp. 15–20. ISBN: 978-1-4503-1586-9. URL: `http://www.eurecom.fr/~troncy/Publications/Troncy-saw12.pdf`.

4

Micropost Annotation

4.1 Introduction

Microposts are the textual metadata that accompany media items. *Per se*, these microposts are nothing but strings. For the task of making sense out of social network microposts, our contributions are methods to consolidate and rank the results of multiple named entity recognition and disambiguation Web services that we have unified in form of a wrapper Web service that (i) takes care of both consolidation and ranking, and that (ii) transparently tracks the underlying Web services' data provenance.

The impact of social networks is ever-growing. According to official statistics, Facebook is the biggest social network with one billion monthly active users[1] as of October 2012. Official user statistics from Twitter[2] stemming from March 2012 suggest that currently more than 140 million active users share 340 million tweets a day. Altogether, the users of social networks produce an incredible amount of public and private data. In this chapter, we thus report on methods to access and make sense out of *public* status updates, or, our preferred term, microposts.

4.1.1 Direct Access to Micropost Raw Data

Social networks in general offer so-called Application Programming Interfaces (APIs) in order to allow for developers to access part of the networks' data programmatically. Similar to the microblogging site Twitter with its search API,[3] Facebook offers both

[1] `http://newsroom.fb.com/Key-Facts`, accessed July 15, 2013

[2] `http://blog.twitter.com/2012/03/twitter-turns-six.html`, accessed July 15, 2013

[3] `https://dev.twitter.com/docs/api/1.1/get/search/tweets`, accessed July 15, 2013

a search function on the website and a search API,[1] and so does Google+.[2] In order to perform data mining, a statistically significant amount of microposts is necessary. Having access to *all* microposts of a service is referred to as having access to the *fire hose*. Typically, developers are only granted access to a smaller random sample of microposts (colloquially referred to as *garden hose* access). While Twitter grants all developers *garden hose* access to its Streaming APIs,[3] for Facebook and Google+ there are no such documented options.

4.1.2 Browser Extensions to Access Microposts Indirectly

To address this shortage, we have developed browser extensions for the two major social networks Facebook and Twitter called Facebook Swarm NLP[4] and Twitter Swarm NLP[5] that can be added to a popular Web browser. These extensions inject JavaScript code into Facebook and Twitter to perform data analysis on the encountered set of *public* microposts by sending extracted data to a central data processing unit. Users need to be logged in to Facebook or Twitter for the extensions to work and must have given their *explicit agreement* during the extension installation process for part of their data to be shared in an anonymized way. While this is far inferior and not comparable with direct *fire hose* access, given a critical amount of participants, it still provides access to a random sample of microposts from different social networks.

4.1.3 Data Analysis Flow

The extensions first retrieve all status updates from the contacts that are displayed on the current user's timeline. Second, the extensions perform named entity extraction (NEE) and disambiguation via Natural Language Processing (NLP) using a remote NLP API on each of the microposts in order to add semantic meaning to them. The extracted named entities are then displayed along each micropost, as illustrated in Figure 4.1. Finally the extracted named entities are sent to a central Web analytics framework [26] to compute basic or advanced trends, for example, by ranking the most discussed named

[1] `https://developers.facebook.com/docs/reference/api/`, accessed July 15, 2013

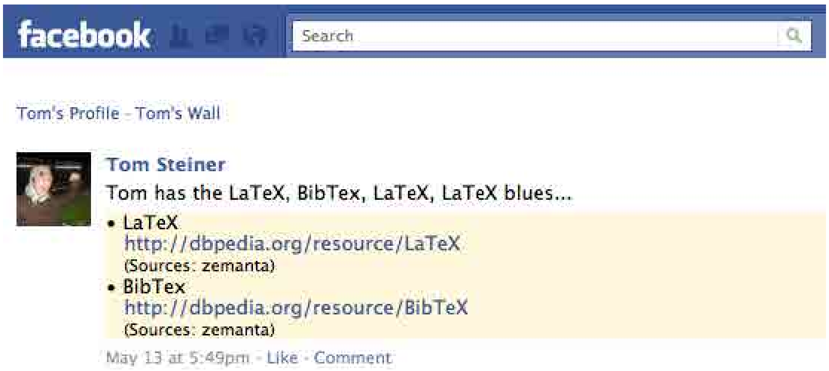
[2] `https://developers.google.com/+/api/`, accessed July 15, 2013

[3] `https://dev.twitter.com/docs/streaming-apis`, accessed July 15, 2013

[4] `http://bit.ly/facebookswarmnlp`, accessed July 15, 2013

[5] `http://bit.ly/twitterswarmnlp`, accessed July 15, 2013

entities per day, or by pivoting named entities by Web analytics data, like users' geographic locations. We remark that the shared data is completely anonymized and cannot be traced back to the originating social network users.



Tom's Profile - Tom's Wall

Tom Steiner

Tom has the LaTeX, BibTex, LaTeX, LaTeX blues...

- LaTeX
  http://dbpedia.org/resource/LaTeX
  (Sources: zemanta)
- BibTex
  http://dbpedia.org/resource/BibTeX
  (Sources: zemanta)

May 13 at 5:49pm · Like · Comment

Figure 4.1: Facebook Swarm NLP browser extension. Extracted named entities have a pale yellow background.

4.1.4 A Wrapper API for Named Entity Disambiguation

As mentioned before, in order to perform named entity extraction and disambiguation, we rely on a wrapper API that calls existing third-party NLP APIs in the background and that delivers the combined results of these APIs in a consolidated way. It is desirable (i) to credit back the contribution of each single third-party API to the joint results, and (ii) to track the provenance of the joint results in order to understand how they were formed. We will show how these two constraints can be fulfilled in a generalizable way at the concrete example of the wrapper NLP API used for our browser extensions.

4.2 Related Work

We regard related work from different angles. First, we look at different approaches for named entity disambiguation, which are relevant for adding meaning to microposts. Second, we look at efforts to mash-up Web services, which is important for tracking data provenance when using multiple APIs in combination.

4.2.1 Named Entity Disambiguation Using Lexical Databases

In [11], Choudhury *et al.* describe a framework for the semantic enrichment, ranking, and integration of Web video tags using Semantic Web technologies. This task is more related to microposts than it seems at first sight: video tags can consist of more than one word and microposts (especially on Twitter) oftentimes consist of just a few words. In order to enrich the typically sparse user-generated tag space, metadata like the recording time and location or the video title and video description are used, but also social features such as the playlists where a video appears in and related videos. Next, the tags are ranked by their co-occurrence and in a final step interlinked with DBpedia [1, 30] concepts for greater integration with other datasets. The authors disambiguate the tags based on WordNet [16, 36] synsets (groups of data elements that are considered semantically equivalent for the purpose of information retrieval) if possible, *i.e.*, if there is only one matching synset in WordNet, the corresponding WordNet URI in DBpedia is selected. If there are more than one matching synsets, the tags' and their context tags' similarity is computed to decide on an already existing tag URI.

4.2.2 Named Entity Disambiguation Using Semantic Coherence and News Trends

In [17], Fernández *et al.* examine named entity disambiguation in the context of news annotation. Their approach consists of three steps: finding the candidate instances in the NEWS ontology [18] for each entity in a news item, ranking these candidate instances using a modified version of PageRank [8], and finally retraining the algorithm with the journalist's feedback once the process is finished. The approach first takes into account the number of occurrences of candidate entities in the past in order to find news trends, and second, the occurrences of candidate entities in past articles in the same categories in order to find semantic coherences.

4.2.3 Named Entity Disambiguation Using Semantic Disambiguation Dictionaries

In [37], Nguyen *et al.* show how semantic disambiguation dictionaries can be used to disambiguate named entities using Wikipedia disambiguation pages. For a set of named entity candidates, all disambiguations are ranked using tf-idf (or cosine similarity) [32].

The approach is a hybrid and incremental process that utilizes previously identified named entities and related terms that co-occur with ambiguous names in a text for the purpose of entity disambiguation.

4.2.4 Disambiguation Using Corpuses and Probability

Cucerzan shows in [13] the use of a corpus like Wikipedia for entity disambiguation. The surrounding words of the to-be-disambiguated terms plus the tags and categories of the related Wikipedia articles are used to determine semantic coherence and thus to decide on the most probable entity candidate. This happens through a process of heuristically maximizing the agreement between contextual information extracted from Wikipedia and the context of a document.

4.2.5 Disambiguation Using Search Query Logs

In [2], Billerbeck *et al.* use click graphs and session graphs of users' search engine sessions to semantically bridge different queries in order to retrieve entities for a concrete entity retrieval query. Click graphs are created by using queries and URLs as nodes and connecting and weighting them by their click frequencies. Session graphs are created by using only queries as nodes with edges between them if they appear in the same user sessions, again weighted by co-occurrence frequencies. An exemplary entity retrieval query is *hybrid cars*, semantically bridgeable queries are consequently *toyota prius*, or *honda civic hybrid*). These entities are then ranked and returned to the user.

4.2.6 Combining Different Web Services and Provenance

In [20], Groth *et al.* describe how so-called mash-ups can be created in a dynamic, just-in-time way, combining data from different data sources through tools and technologies such as Yahoo! Pipes,[1] RSS [9], and APIs. The authors are driven by the motivation to allow for trust and confidence in mash-ups, and therefore consider it critical to be able to analyze the origin of combined results. They suggest an approach based on OWL [34] and XML [6], with a focus on process documentation. However, different from our work, where the goal is to transparently add provenance data at API invocation time, their focus is more on overall process documentation in the context of a mash-up application.

[1] `http://pipes.yahoo.com/pipes/`, accessed July 15, 2013

The focus of Carroll *et al.* in [10] is on the provenance of triples in the Semantic Web world, namely, for making statements about triples in graphs. Therefore, the authors introduce the concept of Named Graphs, an extension to RDF [27]. In contrast to our work, Carroll *et al.* focus purely on using triples to make statements about triples (*i.e.*, stay in the RDF world), whereas our approach uses RDF to make statements about potentially any API result.

Web service specifications in the context of the first-generation standards represented by WSDL [12], SOAP [21], and UDDI [40] are occasionally referred to collectively as *WS-**. In the *WS-** world, BPEL4WS, described by Curbera *et al.* in [14] provides a formal language for the specification of business processes and business interaction protocols. This allows for the combination of several APIs. However, it does not credit back concrete outputs of a combined API to the underlying APIs.

4.3 Structuring Unstructured Textual Data

When we speak of adding structure to unstructured textual data, we mean the process of extracting the main concepts in the form of named entities from a given text and the process of disambiguating those named entities, *i.e.*, the removal of uncertainty of meaning from an ambiguous named entity like *Barcelona*, which can stand for the football club, or the city of Barcelona. An *entity* is defined by WordNet [16, 36] as *"that which is perceived or known or inferred to have its own distinct existence (living or nonliving)."* Typically, named entities from a text can be persons, companies, organizations, geographies, but also things like quantities, expressions of time, books, albums, authors, *etc.* The extraction of named entities is commonly based on Natural Language Processing (NLP) combined with Machine Learning.

4.3.1 Natural Language Processing Services

WordNet [16, 36] defines *Natural Language Processing* as *"the branch of information science that deals with natural language information."* From the many NLP toolkits available, hereafter, we list some NLP Web services that link to datasets in the Linking Open Data cloud [4, 15] in order to disambiguate named entities.

OpenCalais

The OpenCalais[1] Web service automatically creates rich semantic metadata for textual documents. Using Natural Language Processing (NLP), machine learning, and other methods, OpenCalais analyzes documents and finds the entities within, and also returns the facts and events hidden within them. OpenCalais is the only of the examined Web services that provides details on occurrences in concrete sections of the submitted coherent text. This allows for the exact matching of the location in the text where a certain entity is detected. This is especially useful as OpenCalais is also oftentimes capable of recognizing references within the text to prior discovered entities (for example, in the following text, *he* is mapped back to Obama: "Obama *thanked people for their work in ensuring the victory.* He *also thanked his family."*). An OpenCalais response consists of three parts:

- a list of topics that the text is categorized in
- a list of concrete entities that occur in the text
- a list of social concept tags

The problem with the extracted entities is that they are not always uniquely disambiguated. An example is the named entity represented by the URL `http://d.opencalais.com/pershash-1/cf42394f-4ae9-3e8e-958a-088149c86565.html` that represents the concept of an entity of type `person` named *Barack Hussein Obama*. However, a `person`-type *Barack Obama* entity from the same document is also represented by the URL `http://d.opencalais.com/pershash-1/cfcf1aa2-de05-3939-a7d5-10c9c7b3e87b.html` Other services successfully disambiguated both occurrences and recognized them to stand for the same person, President Obama. A second issue is that only a tiny fraction of the returned entities link to other data sources in the LOD cloud [4, 15]. In order to discover links to the LOD cloud, each returned entity URL has to be retrieved at the expense of an HTTP request and the returned RDF checked for said links.

[1] `http://www.opencalais.com/documentation/opencalais-documentation`, accessed July 15, 2013

AlchemyAPI

AlchemyAPI[1] is capable of identifying people, companies, organizations, cities, geographic features, and other typed entities within textual documents. The service employs statistical algorithms and NLP to extract semantic richness embedded within text. AlchemyAPI differentiates between entity extraction and concept tagging. AlchemyAPI's concept tagging API is capable of abstraction, *i.e.*, understanding how concepts relate and tag them accordingly ("Hillary Clinton", "Michelle Obama", and "Laura Bush" are all tagged as "First Ladies of the United States"). In practice, the difference between named entity extraction and concept tagging is subtle. In consequence, we treat entities and concepts the same. Overall, AlchemyAPI results are very accurate and in the majority of cases well interlinked with members of the LOD cloud, among others with DBpedia [1, 30], OpenCyc [31], and Freebase [5, 33]. AlchemyAPI also provides links to other data sources, however, sometimes the returned URLs result in `404 Not found`. One example that we came across during our tests was the URL `http://umbel.org/umbel/ne/wikipedia/George_W._Bush.rdf`, which should represent the concept of the person George W. Bush. The URL does serve as a Semantic Web identifier, however, harms the third Linked Data principle, as outlined in subsection 2.5.1. AlchemyAPI also oftentimes returns thematically closely related, but for a concrete text not directly relevant entities beyond the abstract concepts from its concept tagging service, for example, in a text about the CEO of a given company, the name of the CEO of one of its competitors.

Zemanta

Zemanta[2] allows developers to query the service for contextual metadata about a given text. The returned components currently span four categories: articles, keywords, photos, in-text links, and optional component categories. The service provides high quality entities that are linked to well-known datasets of the LOD cloud, *e.g.*, DBpedia or Freebase. Zemanta convinces through very accurate entity disambiguation and thus high precision, however, at the cost of recall. Where other services return named entities of lower precision, the design objectives of Zemanta instead seem to prefer not to return anything over returning returning low-precision results.

[1] `http://www.alchemyapi.com/api/entity/`, accessed July 15, 2013

[2] `http://developer.zemanta.com/docs/`, accessed July 15, 2013

DBpedia Spotlight

DBpedia Spotlight [35] is a tool for annotating mentions of DBpedia resources in text, providing a solution for linking unstructured information sources to the LOD cloud through DBpedia. DBpedia Spotlight performs named entity extraction, including entity detection and disambiguation with adjustable precision and recall. DBpedia Spotlight allows users to customize the annotations to their specific needs through the DBpedia Ontology[1] and quality measures such as prominence, topical pertinence, contextual ambiguity, and disambiguation confidence.

4.3.2 Machine Translation

Social networking happens at a global scale. In consequence, many microposts are authored in languages different from English. In order to still make sense out of those microposts, we apply machine translation to translate non-English microposts to English. We use the Google Translate API,[2] which, if the source language parameter is left blank, tries to first detect the source language before translating to English.

4.3.3 Part-of-Speech Tagging

Our processing chain supports part-of-speech (POS) tagging based on a Brill POS tagger [7] adapted for JavaScript. Brill taggers work by assigning tags to each word and then changing them using a set of predefined rules. In an initial run, if a word is known, the tagger first assigns the most frequent tag, or, if a word is unknown, it naively assigns the tag "noun" to it. By applying the processing rules over and over again and changing the incorrect tags, a sufficiently high accuracy is achieved. In the current processing chain, POS tagging does not yet play an active role, however, we aim for leveraging the additional data for better micropost analysis in the future.

4.4 Consolidating Named Entity Disambiguation Results

In this section, we motivate the use of multiple named entity disambiguation Web services in *parallel* with the objective of obtaining named entity candidates for a textual document such as a micropost. The task of evaluating and aligning named entity

[1] `http://wiki.dbpedia.org/Ontology`, accessed July 15, 2013

[2] `https://developers.google.com/translate/v2/getting_started`, accessed July 15, 2013

extraction and disambiguation APIs and their typed output has been formally addressed by Rizzo *et al.* in the context of the NERD framework [38, 39]. We have decided for a type-agnostic approach, which we motivate in the following.

4.4.1 Identity Links on the Semantic Web

From the considered services, only OpenCalais returns data in its own namespace (`http://d.opencalais.com/*`), which is interlinked with other datasets in the LOD cloud, however, not in all cases. All other services return results either directly in the DBpedia namespace (`http://dbpedia.org/resource/*`), as in the case of DBpedia Spotlight. Alternatively, AlchemyAPI and Zemanta return results in the DBpedia namespace together with namespaces like Freebase (`http://rdf.freebase.com/rdf/*`).

In order to address the problem of different namespaces in results, an approach as presented by Glaser *et al.* in [19] based on `owl:sameAs` links could be used. In practice, however, while many resources in the Linked Data world are marked as equivalent to each other, the quality of such equivalence links is not always optimal. An example of a good equivalence link is shown in Listing 4.1.

```
<http://dbpedia.org/resource/Barack_Obama> ↩
  <http://www.w3.org/2002/07/owl#sameAs> ↩
  <http://rdf.freebase.com/rdf/en.barack_obama> .
```

Listing 4.1: Example of a good equivalence link

As Halpin *et al.* show in a study [22], the problem with `owl:sameAs` is that people tend to use it in different ways with different intentions. In [22], the authors differentiate between four separate usage styles, ranging from expressing loose relatedness to strict equivalence. Despite the different intentions, people tend to incorrectly use `owl:sameAs` habitually, according to the study. Inference is thus problematic, if not impossible, when the intention of the link creator of the particular `owl:sameAs` link is unknown.

4.4.2 Linked Data Principles Applied

We recall the Linked Data principles, that were outlined in subsection 2.5.1. In order to represent extracted named entities from social network microposts in an unambiguous way, we apply the Linked Data principles by representing named entities in microposts with HTTP URIs that can be dereferenced for retrieving the corresponding information. This is taken care of by the third-party NLP APIs that we use for our experiments, namely OpenCalais, Zemanta, DBpedia Spotlight, and AlchemyAPI. These APIs take a textual document as an input, perform named entity extraction and disambiguation on it, and finally link the detected named entities back into the LOD cloud. We use these APIs in parallel and by combining their results aim for the emergence effect[1] in the sense of Aristotle: *"the totality is not, as it were, a mere heap, but the whole is something besides the parts."*

We recall the wrapper API described in subsection 4.1.4 that calls third-party NLP Web services in order to return a combined result of consolidated entities. All NLP Web services return lists of entities with their respective types and/or subtypes, names, relevance, and URIs that interlink the entity in question with the LOD cloud. The problem is that each service has implemented its own typing system. Providing mappings for all of them is a time-consuming, cumbersome task. While Rizzo *et al.* have defined mappings in the context of the NERD framework [38, 39], we decided for a different approach. As all services provide links into the LOD cloud, the desired typing information can be retrieved from there in a true Linked Data manner if need be.

We illustrate the approach with an example: *"Google Inc. is an American multinational corporation which provides Internet-related products and services, including Internet search, cloud computing, software and advertising technologies."* If we use the just mentioned text as an input for the NLP wrapper API, among others, we expect to retrieve the named entity for the company Google, represented by, for example, the URL `http://dbpedia.org/resource/Google` as an output.

Listing 4.2 shows the output of just Zemanta in isolation, Listing 4.3 shows the output of just AlchemyAPI in isolation, and finally, Listing 4.4 shows the consolidated output of the two named entity recognition APIs together. In this example, the entity names differ ("Google Inc." vs. "Google"). However, going down the list of URLs for each entity from the two services, the consolidation algorithm matches via the URL `http://dbpedia.org/resource/Google`. Given the different two entity names ("Google Inc." vs. "Google"), the consolidated name is then an array of all detected names. Each service already includes a relevance score ranging from 0 (irrelevant) to 1 (relevant). The consolidated relevance is calculated via the averaged relevance scores of both services.

[1] Aristotle, Metaphysics, Book H 1045a 8–10

While there may be different definitions of relevance applied by each service and given that these differences are not disclosed, the arithmetic mean is a pragmatic way to deal with the situation, especially as all services use relevance scores between 0 and 1. We maintain provenance metadata for each URI on the JSON representation, as can be seen in Listing 4.4. Finally, we repeat the process for all other services.

```
[
  {
    "name": "Google Inc.",
    "relevance": 0.972007,
    "uris": [
      {
        "uri": "http://rdf.freebase.com/ns/en/google",
        "source": "zemanta"
      },
      {
        "uri": "http://dbpedia.org/resource/Google",
        "source": "zemanta"
      }
    ],
    "source": "zemanta"
  }
]
```

Listing 4.2: Output of Zemanta in isolation

```
[
  {
    "name": "Google",
    "relevance": 0.535781,
    "uris": [
      {
        "uri": "http://dbpedia.org/resource/Google",
        "source": "alchemyapi"
      },
      {
        "uri": "http://rdf.freebase.com/ns/guid.9202a8c04000641f800000000042acea",
        "source": "alchemyapi"
      }
    ],
    "source": "alchemyapi"
  }
]
```

Listing 4.3: Output of AlchemyAPI in isolation

```
[
  {
    "name": [
      "Google",
      "Google Inc."
    ],
    "relevance": 0.753894,
    "uris": [
      {
        "uri": "http://rdf.freebase.com/ns/en/google",
        "source": "zemanta"
      },
      {
        "uri": "http://dbpedia.org/resource/Google",
        "source": "zemanta"
      },
      {
        "uri": "http://rdf.freebase.com/ns/guid.9202a8c04000641f800000000042acea",
        "source": "alchemyapi"
      }
    ],
    "source": "zemanta,alchemyapi"
  }
]
```

Listing 4.4: Consolidated output of two named entity recognition APIs, namely Zemanta and AlchemyAPI

4.5 Tracking Provenance With Multiple Sources

As outlined before, we use several APIs in combination in order to add meaning to social network microposts. Extracted named entities from a micropost can in consequence be the result of up to four agreeing (or disagreeing) API calls.

4.5.1 The Need for Providing Provenance Metadata

Hartig *et al.* mention in [24] reasons that justify the need for provenance metadata. Among these reasons are linked dataset replication and distribution on the Web with not necessarily identical namespaces: based on the same source data, different publishers can create diverging copies of a linked dataset with different levels of interconnectedness. We add to this the automated conversion of unstructured data to Linked Data with heuristics, where extracted entities—albeit consolidated and backed by different data sources—might still be wrong. Especially with our wrapper approach, it is desirable to be able to track back to the concrete source where a certain piece of information came from. This enables (i) to correct the error at the root of our API (fighting the cause) or (ii) to correct the concrete error in an RDF annotation (fighting the symptom), and (iii) most importantly, to judge the trustworthiness and quality of a dataset.

In order to track the contributions of the various sources, we have opted to use the Provenance Vocabulary [23] by Hartig and Zhao with the prefix `prv`, the HTTP Vocabulary in RDF [28] by Koch *et al.* with prefix `http`, and a vocabulary for representing content in RDF [29] by the same authors with prefix `cnt`. We have chosen the HTTP Vocabulary in RDF for the fact that it is a W3C Working Draft developed by the Evaluation and Repair Tools Working Group (ERT WG), which is part of the World Wide Web Consortium (W3C) Web Accessibility Initiative (WAI). The Provenance Vocabulary was chosen because of its existing deployment in several projects, such as Pubby,[1] Triplify,[2] and D2R Server.[3]

While our wrapper API supports two output formats (`application/json` and `text/turtle`), we have added provenance information exclusively to the `text/turtle` variant. In order to represent the extracted named entities in a micropost, we use the Common Tag vocabulary [43]. A micropost is `ctag:tagged` with a `ctag:Tag`, which consists of a textual `ctag:label` and a pointer to a resource that specifies what the label `ctag:means`. The Common Tag vocabulary is well-established and developed by both industry and academic partners. In order to make statements about a bundle of triples, we group them in a named graph. We use the TriG [3] syntax, an example can be seen in Listing 4.5.

```
:G = {
  <https://www.facebook.com/Tomayac/posts/10150175940867286> ctag:tagged [
      a~ctag:Tag ;
      ctag:label "BibTeX" ;
      ctag:means <http://dbpedia.org/resource/BibTeX>
    ] .
} .
```

Listing 4.5: Example named graph in TriG syntax

[1] `http://wifo5-03.informatik.uni-mannheim.de/pubby/`, accessed July 15, 2013

[2] `http://triplify.org/Overview`, accessed July 15, 2013

[3] `http://wifo5-03.informatik.uni-mannheim.de/bizer/d2r-server/`, accessed July 15, 2013

4.5.2 The Provenance Vocabulary

In this section, we outline the required steps in order to make statements about the provenance of a group of triples contained in a named graph `:G` that was generated using several HTTP `GET` requests to third-party APIs. We use the Provenance Vocabulary [23] with prefix `prv`, the HTTP Vocabulary in RDF [28] with prefix `http`, the Identity of Resources on the Web ontology[1] (IRW) with the prefix `irw`, and the Representing Content in RDF vocabulary [29] with prefix `cnt`.

As a first step, we state that `:G` is both a `prv:DataItem` and an `rdfg:Graph`. `:G` is `prv:createdBy` the process of a `prv:DataCreation`. This `prv:DataCreation` is `prv:performedBy` a `prv:NonHumanActor`, a `prvTypes:DataProvidingService` to be precise (simplified as `http://tomayac.no.de/entity-extraction/combined` in Listing 4.6). This service is `prv:operatedBy` a human, in the concrete case ourselves, (`http://tomayac.com/thomas_steiner.rdf#me`). Time is important for provenance, so the `prv:performedAt` date of the `prv:DataCreation` needs to be saved. During the process of the `prv:DataCreation` there are `prv:usedData`, which are `prv:retrievedBy` a `prv:DataAcess` that is `prv:performedAt` a certain time, and `prv:performedBy` a non-human actor (our API) that is `prv:operatedBy` the same human as before. For the `prv:DataAccess` (there is one for each involved API), we `prv:accessedService` from a `prv:DataProvidingService` of which we `prv:accessedResource` that is available at a certain `irw:WebResource`. Therefore, we `prvTypes:exchangedHTTPMessage` which is an `http:Request` using `http:httpVersion` “1.1” and the `http:methodName` “GET”.

4.5.3 Provenance RDF Overview

This section provides a shortened overview of the provenance RDF serialized in Turtle syntax for a micropost that was automatically tagged with the label “BibTeX” and the assigned meaning `http://dbpedia.org/resource/BibTeX`. The named graph `:G` in the first part of Listing 4.6 contains the absolute data (the fact that the micropost with the URL `https://www.facebook.com/Tomayac/posts/10150177486072286` is tagged with the label “BibTeX”, which is represented by the HTTP URL `http://dbpedia.org/resource/BibTeX`). The second part with metadata about `:G` says that these facts were generated via two calls, one using the HTTP method `GET`, and the other `POST`. It

[1] `http://www.ontologydesignpatterns.org/ont/web/irw.owl#`, accessed July 15, 2013

is to be noted that statements such as in Listing 4.6 refer to the triple objects as an identifier for a Web resource (where the Web resource is a representation of the result of the API call at the time where it was `prv:performedAt`). As provenance metadata always refers to the time context in which a certain statement was made, it is essentially unimportant what representation the resource returns in future.

```
:G = {
  <https://www.facebook.com/Tomayac/posts/10150177486072286> ctag:tagged [
     a ctag:Tag ;
     ctag:label "BibTeX" ;
     ctag:means <http://dbpedia.org/resource/BibTeX> ;
  ] .
} .


:G
  a prv:DataItem ;
  a rdfg:Graph ;
  prv:createdBy [
    a prv:DataCreation ;
    prv:performedAt "2011-05-20T15:06:30Z"^^xsd:dateTime ;
    prv:performedBy <http://tomayac.no.de/entity-extraction/combined> ;
    prv:usedData [
      prv:retrievedBy [
        a prv:DataAcess ;
        prv:performedAt "2011-05-20T15:06:30Z"^^xsd:dateTime ;
        prv:performedBy <http://tomayac.no.de/entity-extraction/combined> ;
        prv:accessedService <http://spotlight.dbpedia.org/rest/annotate> ;
        prv:accessedResource
          <http://spotlight.dbpedia.org/rest/annotate?text=Tom%20has%20... ↩
              blues&confidence=0.4&support=20> ;
        prvTypes:exchangedHTTPMessage [
          a http:Request ;
          http:httpVersion "1.1" ;
          http:methodName "GET" ;
          http:mthd <http://www.w3.org/2008/http-methods#GET> ;
        ] ;
      ] ;
    ] ;
    prv:usedData [
      prv:retrievedBy [
        a prv:DataAcess ;
        prv:performedAt "2011-05-20T15:06:41Z"^^xsd:dateTime ;
        prv:performedBy <http://tomayac.no.de/entity-extraction/combined> ;
        prv:accessedService <http://api.zemanta.com/services/rest/0.0/> ;
        prv:accessedResource <http://api.zemanta.com/services/rest/0.0/> ;
        prvTypes:exchangedHTTPMessage [
          a http:Request ;
        http:httpVersion "1.1" ;
        http:methodName "POST" ;
        http:mthd <http://www.w3.org/2008/http-methods#POST> ;
        http:headers (
          [
            http:fieldName "Content-Type" ;
            http:fieldValue "application/x-www-form-urlencoded" ;
          ]
        )
        http:body [
          a cnt:ContentAsText ;
          cnt:characterEncoding "UTF-8" ;
          cnt:chars """method=zemanta.suggest_markup ↩
          &api_key=Your_API_Key ↩
          &text=Tom%20has%20...blues ↩
          &format=json ↩
          &return_rdf_links=1""" ;
        ] ;
      ] ;
    ] ;
  ] ;
] .
```

Listing 4.6: Shortened overview of the provenance RDF in Turtle syntax for an automatically annotated micropost

4.6 Conclusions

In this chapter, we have shown how the Provenance Vocabulary can be used to keep track of the original third-party Web service calls that led to the consolidated results. These references to the original calls are to be understood as the identification of Web resources, *i.e.*, the results of a request. We have shown how a concrete multi-source Web service can automatically maintain provenance metadata both for entirely machine-generated content, but also for partly (or completely) human-generated content. Being able to track back the origin of a triple is of crucial importance, especially given the network effect which is one of the Linked Data benefits. The generated triples are very verbose, and in consequence stating even relatively simple facts like that a combined result is based on two separate sub-results takes up a lot of space. The verbosity is mainly due to the used vocabularies, namely the Provenance Vocabulary and the HTTP Vocabulary in RDF, which on the one hand is good as it encourages vocabulary reuse, but on the other hand comes at the abovementioned expenses.

Future work will focus on exploring ways to drastically simplify the annotations in order to obtain less verbose provenance descriptions. While it is always easier to propose a specialized vocabulary that does one task well, broader reuse and acceptance can be gained by reusing existing vocabularies. Our ultimate goal is to make provenance annotations lightweight enough that their undoubted benefits outweigh the additional data payload overhead.

Chapter Notes

This chapter is partly based on the following publications.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Facebook Microposts via a Mash-up API and Tracking its Data Provenance". In: *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on.* Oct. 2011, pp. 342–345. URL: http://research.google.com/pubs/archive/37426.pdf.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Social Network Microposts via Multiple Named Entity Disambiguation APIs and Tracking Their Data Provenance". In: *International Journal of Computer Information Systems and Industrial Management* 5 (2013), pp. 69–78. URL: http://www.mirlabs.org/ijcisim/regular_papers_2013/Paper82.pdf.

- Seth van Hooland, Max De Wilde, Ruben Verborgh, Thomas Steiner, and Rik Van de Walle. "Named-Entity Recognition: A Gateway Drug for Cultural Heritage Collections to the Linked Data Cloud?" In: *Literary and Linguistic Computing* (2013). URL: http://freeyourmetadata.org/publications/named-entity-recognition.pdf.

References

[1] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. “DBpedia: a Nucleus for a Web of Open Data”. In: *The Semantic Web: 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference*. ISWC 2007 + ASWC 2007. Springer, 2007, pp. 722–735.

[2] Bodo Billerbeck, Gianluca Demartini, Claudiu S. Firan, Tereza Iofciu, and Ralf Krestel. “Ranking Entities Using Web Search Query Logs”. In: *Proceedings of the 14th European Conference on Research and Advanced Technology for Digital Libraries*. ECDL ’10. Springer, 2010, pp. 273–281.

[3] Chris Bizer and Richard Cyganiak. *The TriG Syntax*. `http://wifo5-03.informatik.uni-mannheim.de/bizer/trig/`, accessed July 15, 2013. 2007.

[4] Chris Bizer, Anja Jentzsch, and Richard Cyganiak. *State of the LOD Cloud*. `http://wifo5-03.informatik.uni-mannheim.de/lodcloud/state/`, accessed July 15, 2013. 2011.

[5] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. “Freebase: a collaboratively created graph database for structuring human knowledge”. In: *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. SIGMOD ’08. Vancouver, Canada: ACM, 2008, pp. 1247–1250. ISBN: 978-1-60558-102-6. DOI: `10.1145/1376616.1376746`. URL: `http://doi.acm.org/10.1145/1376616.1376746`.

[6] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, and François Yergeau. *Extensible Markup Language (XML) 1.0 (Fifth Edition)*. Recommendation. W3C, 2008.

[7] Eric Brill. “A Simple Rule-based Part of Speech Tagger”. In: *Proceedings of the Workshop on Speech and Natural Language*. HLT ’91. Association for Computational Linguistics, 1992, pp. 112–116.

[8] S. Brin and L. Page. “The Anatomy of a Large-Scale Hypertextual Web Search Engine”. In: *Computer Networks and ISDN Systems* 30.1 (1998), pp. 107–117.

[9] Rogers Cadenhead, Sterling Camden, Simone Carletti, James Holderness, Jenny Levine, Eric Lunt, et al. *RSS 2.0 Specification*. `http://www.rssboard.org/rss-specification`, accessed July 15, 2013. 2006.

[10] Jeremy J. Carroll, Christian Bizer, Patrick J. Hayes, and Patrick Stickler. “Named Graphs, Provenance and Trust”. In: *Proceedings of the 14th International Conference on World Wide Web, WWW 2005*. ACM, 2005, pp. 613–622.

[11] Smitashree Choudhury, John G. Breslin, and Alexandre Passant. “Enrichment and Ranking of the YouTube Tag Space and Integration with the Linked Data Cloud”. In: *Proceedings of the 8th International Semantic Web Conference*. ISWC ’09. Springer, 2009, pp. 747–762.

[12] Erik Christensen, Francisco Curbera, Greg Meredith, and Sanjiva Weerawarana. *Web Services Description Language (WSDL) 1.1*. Note. W3C, 2001.

[13] Silviu Cucerzan. “Large-Scale Named Entity Disambiguation Based on Wikipedia Data”. In: *EMNLP-CoNLL: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 2007, pp. 708–716.

[14] Francisco Curbera, Rania Khalaf, Nirmal Mukhi, Stefan Tai, and Sanjiva Weerawarana. “The Next Step in Web Services”. In: *Communications of the ACM* 46.10 (2003), pp. 29–34.

[15] Richard Cyganiak and Anja Jentzsch. *The Linking Open Data cloud diagram*. `http://lod-cloud.net/`, accessed July 15, 2013. 2011.

[16] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Language, Speech and Communication Series. Cambridge, MA: MIT Press, 1998.

[17] Norberto Fernández, José M. Blázquez, Luis Sánchez, and Ansgar Bernardi. “IdentityRank: Named Entity Disambiguation in the Context of the NEWS Project”. In: *Proceedings of the 4th European conference on the Semantic Web: Research and Applications*. ESWC ’07. Springer, 2007, pp. 640–654.

[18] Norberto Fernández, Damaris Fuentes, Luis Sánchez, and Jesús A. Fisteus. “The NEWS ontology: Design and Applications”. In: *Expert Systems with Applications* 37.12 (2010), pp. 8694–8704.

[19] Hugh Glaser, Afraz Jaffri, and Ian Millard. “Managing Co-reference on the Semantic Web”. In: *Proceedings of the WWW2009 Workshop on Linked Data on the Web*. Vol. 538. LDOW ’09. CEUR Workshop Proceedings, 2009.

[20] Paul Groth, Simon Miles, and Luc Moreau. “A Model of Process Documentation to Determine Provenance in Mash-ups”. In: *Transactions on Internet Technology* 9 (1 2009), pp. 1–31.

[21] Martin Gudgin, Marc Hadley, Noah Mendelsohn, Jean-Jacques Moreau, Henrik Frystyk Nielsen, Anish Karmarkar, et al. *SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)*. Recommendation. W3C, 2007.

[22] Harry Halpin, Patrick J. Hayes, James P. McCusker, Deborah L. McGuinness, and Henry S. Thompson. "When owl:sameAs Isn't the Same: An Analysis of Identity in Linked Data". In: *International Semantic Web Conference (1)*. Springer, 2010, pp. 305–320.

[23] Olaf Hartig and Jun Zhao. *Provenance Vocabulary Core Ontology Specification*. `http://purl.org/net/provenance/ns`, accessed July 15, 2013. 2012.

[24] Olaf Hartig and Jun Zhao. "Publishing and Consuming Provenance Metadata on the Web of Linked Data". In: *Provenance and Annotation of Data and Processes – Third International Provenance and Annotation Workshop*. IPAW '10. Springer, 2010, pp. 78–90.

[25] Seth van Hooland, Max De Wilde, Ruben Verborgh, Thomas Steiner, and Rik Van de Walle. "Named-Entity Recognition: A Gateway Drug for Cultural Heritage Collections to the Linked Data Cloud?" In: *Literary and Linguistic Computing* (2013). URL: `http://freeyourmetadata.org/publications/named-entity-recognition.pdf`.

[26] A. Kaushik. *Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity*. John Wiley & Sons, 2009.

[27] Graham Klyne and Jeremy J. Carroll. *Resource Description Framework (RDF): Concepts and Abstract Syntax*. Recommendation. W3C, 2004.

[28] Johannes Koch, Carlos A. Velasco, and Philip Ackermann. *HTTP Vocabulary in RDF 1.0*. Working Draft. W3C, 2011.

[29] Johannes Koch, Carlos A. Velasco, and Philip Ackermann. *Representing Content in RDF 1.0*. Working Draft. W3C, 2011.

[30] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, et al. "DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia". In: *Semantic Web Journal* (2013). URL: `http://www.semantic-web-journal.net/system/files/swj499.pdf`.

[31] Douglas B. Lenat. "CYC: a Large-scale Investment In Knowledge Infrastructure". In: *Communications of the ACM* 38.11 (Nov. 1995), pp. 33–38.

[32] C.D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. An Introduction to Information Retrieval. Cambridge University Press, 2008.

[33] John Markoff. *Start-Up Aims for Database to Automate Web Searching.* `http://www.nytimes.com/2007/03/09/technology/09data.html`, accessed July 15, 2013. 2007.

[34] Deborah L. McGuinness and Frank van Harmelen. *OWL Web Ontology Language: Overview.* Recommendation. W3C, 2004.

[35] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. "DBpedia spotlight: shedding light on the web of documents". In: *Proceedings of the 7th International Conference on Semantic Systems.* I-Semantics '11. Graz, Austria: ACM, 2011, pp. 1–8. ISBN: 978-1-4503-0621-8. DOI: `10.1145/2063518.2063519`. URL: `http://doi.acm.org/10.1145/2063518.2063519`.

[36] George A. Miller. "WordNet: a Lexical Database for English". In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[37] Hien T. Nguyen and Tru H. Cao. "Named Entity Disambiguation: a Hybrid Statistical and Rule-Based Incremental Approach". In: *The Semantic Web.* Ed. by John Domingue and Chutiporn Anutariya. Vol. 5367. Lecture Notes in Computer Science. Springer, 2008, pp. 420–433.

[38] G. Rizzo and R. Troncy. "NERD: Evaluating Named Entity Recognition Tools in the Web of Data". In: *Workshop on Web Scale Knowledge Extraction.* WEKEX '11. 2011, pp. 1–16.

[39] Giuseppe Rizzo and Raphaël Troncy. "NERD: a Framework for Unifying Named Entity Recognition and Disambiguation Extraction Tools". In: *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics.* EACL '12. Association for Computational Linguistics, 2012, pp. 73–76.

[40] Marwan Sabbouh, Stu Jolly, Dock Allen, Paul Silvey, and Paul Denning. *Interoperability, W3C Workshop on Web Services.* Position Paper. `http://www.w3.org/2001/03/WSWS-popa/paper08`, accessed July 15, 2013. 2001.

[41] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Facebook Microposts via a Mash-up API and Tracking its Data Provenance". In: *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on.* Oct. 2011, pp. 342–345. URL: `http://research.google.com/pubs/archive/37426.pdf`.

[42] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Social Network Microposts via Multiple Named Entity Disambiguation APIs and Tracking Their Data Provenance". In: *International Journal of Computer Information Systems and Industrial Management* 5 (2013), pp. 69–78. URL: `http://www.mirlabs.org/ijcisim/regular_papers_2013/Paper82.pdf`.

[43] Andraž Tori, Alex Iskold, Alexandre Passant, Vuk Miličić, et al. *Common Tag Specification.* `http://commontag.org/Specification`, accessed July 15, 2013. 2009.

5

Event Detection Based On Wikipedia Edit Spikes

5.1 Introduction

We are surrounded by events, most of which we do not care much about. In this chapter, we will show an approach towards *breaking news event detection* of relevant global or local news events that is based on concurrent Wikipedia edit spikes. Yang, Pierce, and Carbonell define [16] *event detection* as follows.

> *"Event detection is essentially a discovery problem,* i.e., *mining the data stream for new patterns in document content."*

They differentiate two types of event detection techniques.

> "Retrospective *event detection is the task of grouping stories in a corpus where each group uniquely identifies an event.* On-line *event detection is the problem of labeling each document as it arrives in sequence with a New or Old flag, indicating whether or not the current document is the first story discussing a novel event at that time."*

Allan, Papka, and Lavrenko use the following definitions [1].

> *"The goal of those tasks [new event detection and event tracking] is to monitor a stream of broadcast news stories so as to determine the relationships between the stories based on the real-world events that they describe.*

> New event detection *requires identifying those news stories that discuss an event that has not already been reported in earlier stories.* Event tracking *means starting from a few sample stories and finding all subsequent stories that discuss the same event."*

In our research, we focus on online new event detection based on Wikipedia edit spikes. We have developed an application called *Wikipedia Live Monitor* that monitors article edits on different language versions of Wikipedia—as they happen in realtime. Wikipedia articles in different languages are highly interlinked. For example, the English article `en:2013_Russian_meteor_event` on the topic of the February 15 meteoroid that exploded over the region of Chelyabinsk Oblast, Russia, is interlinked with `ru:Падение_метеорита_на_Урале_в_2013_году`, the Russian article on the same topic. As we monitor multiple language versions of Wikipedia in parallel, we can exploit this fact to detect *concurrent edit spikes* of Wikipedia articles covering the same topics both in only one and in different languages. We treat such concurrent edit spikes as signals for potential breaking news events, whose plausibility we then check with full-text cross-language searches on multiple social networks. Unlike the reverse approach of monitoring social networks first and potentially checking plausibility on Wikipedia second, the approach proposed in this chapter has the advantage of being less prone to false-positive alerts, while being equally sensitive to true-positive events, however, at only a fraction of the processing cost. A live demo of our application is available online at the URL `http://wikipedia-irc.herokuapp.com/` (accessed July 15, 2013), the source code is available under the terms of the Apache 2.0 license at `https://github.com/tomayac/wikipedia-irc` (accessed July 15, 2013).

5.1.1 Motivation

Shortly after the celebrity news website TMZ broke the premature news that the King of Pop *Michael Jackson* (MJ) had died,[1] the Internet slowed down.[2] Initially, Wikipedia's website administrators started noting abnormal load spikes [14]. Shortly afterwards, caching issues caused by a so-called edit war [2] led the site to go down: Wikipedia editors worldwide made concurrent edits to the Michael Jackson Wikipedia article,

[1] `http://www.tmz.com/2009/06/25/michael-jackson-dies-death-dead-cardiac-arrest/`, accessed July 15, 2013

[2] `http://news.bbc.co.uk/2/hi/technology/8120324.stm`, accessed July 15, 2013

doing and undoing changes regarding the tense of the article, death date, and the circumstances of the (at the time) officially still unconfirmed fatality. While Wikipedia engineers have worked hard to ensure that future load spikes do not take the site down again, there is without dispute a lot of research potential in analyzing editing activity.

5.1.2 Hypotheses and Research Questions

In this chapter, we present an application that monitors article edits of different language versions of Wikipedia in realtime in order to detect concurrent edit spikes that may be the source of breaking news events. When a concurrent edit spike has been detected, we use cross-language full-text searches on social networks as plausibility checks to filter out false-positive alerts. We are led by the following hypotheses.

(H1) Breaking news events spread over social networks, independent from where the news broke initially.

(H2) If a breaking news event is important, it will be reflected on at least one language edition of Wikipedia.

(H3) The time between when the news broke first and the news being reflected on Wikipedia is considerably short.

These hypotheses lead us to the research questions below.

(Q1) Can concurrent Wikipedia edit spikes combined with social network plausibility checks capture major breaking news events, and if so, with what delay?

(Q2) Is the approach *Wikipedia first, social networks second* at least as powerful as the reverse approach?

In this chapter, we do not answer all research questions yet, however, lay the foundation stone for future research in this area by introducing *Wikipedia Live Monitor*.

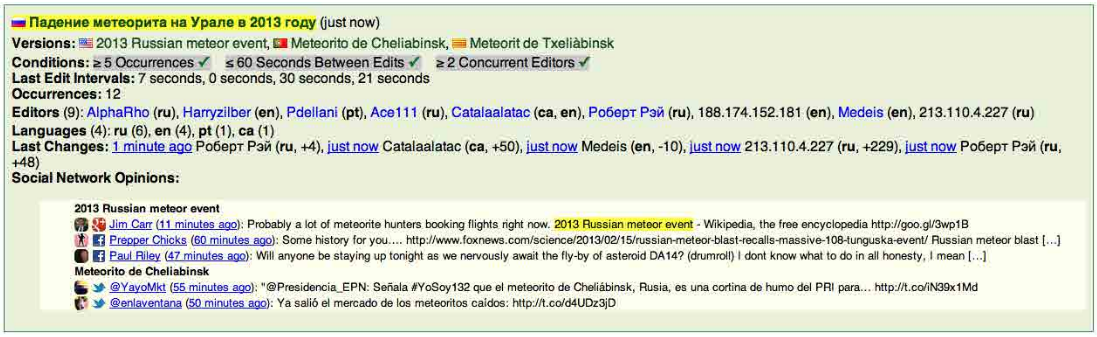
5.2 Related Work

We refer to an event as breaking news, if the event is of significant importance to a considerable amount of the population. Petrović *et al.* define [8] the goal of new event detection (or first story detection) as *"given a sequence of stories, to identify the first* *story to discuss a particular event."* They define an event as *"something that happens at some specific time and place."* Classic streaming analysis of social network microposts so far has been mainly focused on Twitter, a microblogging social network that provides access to a sampled stream of generated microposts by means of its Streaming API.[1] Petrović *et al.* explain [8]: *"in the streaming model of computation, items arrive continuously in a chronological order, and have to be processed in bounded space and time."* In the referenced paper, the authors report on a system for streaming new event detection applied to Twitter based on locality sensitive hashing. Hu *et al.* provide an analysis of how news break and spread on Twitter [5]. The task of linking news events with social media is covered by Tsagkias *et al.* in [13]. With our work, we stand on the shoulders[2] of Osborne *et al.* [7], who use Wikipedia page view statistics[3] as a means to filter spurious events stemming from event detection over social network streams. Our approach reverses theirs, however, instead of the only hourly updated page view statistics, we use realtime change notifications, as will be explained in subsection 5.3.1. *Wikipedia Live Monitor* is partly based on an application called *Wikistream*, developed by Ed Summers *et al.*, which was described in [11]. In [3], Georgescu *et al.* conduct an in-depth analysis of event-related updates in Wikipedia by examining different indicators for events including language, meta annotations, and update bursts. They then study how these indicators can be employed for automatically detecting event-related updates. In [12], ten Thij *et al.* propose a model for predicting the popularity of promoted content, inspired by the analysis of the page-view dynamics on Wikipedia. Mestyán, Yasseri, and Kertész show in [6] that box office success of movies can be predicted well in advance by measuring and analyzing the activity level of editors and viewers of corresponding articles about the movies in question on Wikipedia by applying a minimalistic predictive model for the financial success based on collective activity data of online users.

[1] `https://dev.twitter.com/docs/api/1.1/get/statuses/sample`, accessed July 15, 2013

[2] Hence the title of the publication related to this chapter.

[3] `http://dumps.wikimedia.org/other/pagecounts-raw/`, accessed July 15, 2013



Падение метеорита на Урале в 2013 году (just now)
Versions: 2013 Russian meteor event, Meteorito de Cheliabinsk, Meteorit de Txeliàbinsk
Conditions: ≥ 5 Occurrences ✓ ≤ 60 Seconds Between Edits ✓ ≥ 2 Concurrent Editors ✓
Last Edit Intervals: 7 seconds, 0 seconds, 30 seconds, 21 seconds
Occurrences: 12
Editors (9): AlphaRho **(ru)**, Harryzilber **(en)**, Pdellani **(pt)**, Ace111 **(ru)**, Catalaalatac **(ca, en)**, Роберт Рэй **(ru)**, 188.174.152.181 **(en)**, Medeis **(en)**, 213.110.4.227 **(ru)**
Languages (4): **ru** (6), **en** (4), **pt** (1), **ca** (1)
Last Changes: 1 minute ago Роберт Рэй (**ru**, +4), just now Catalaalatac (**ca**, +50), just now Medeis (**en**, -10), just now 213.110.4.227 (**ru**, +229), just now Роберт Рэй (**ru**, +48)
Social Network Opinions:

2013 Russian meteor event
Jim Carr (11 minutes ago): Probably a lot of meteorite hunters booking flights right now. 2013 Russian meteor event - Wikipedia, the free encyclopedia http://goo.gl/3wp1B
Prepper Chicks (60 minutes ago): Some history for you.... http://www.foxnews.com/science/2013/02/15/russian-meteor-blast-recalls-massive-108-tunguska-event/ Russian meteor blast [...]
Paul Riley (47 minutes ago): Will anyone be staying up tonight as we nervously await the fly-by of asteroid DA14? (drumroll) I dont know what to do in all honesty, I mean [...]
Meteorito de Cheliabinsk
@YayoMkt (55 minutes ago): "@Presidencia_EPN: Señala #YoSoy132 que el meteorito de Cheliábinsk, Rusia, es una cortina de humo del PRI para... http://t.co/iN39x1Md
@enlaventana (50 minutes ago): Ya salió el mercado de los meteoritos caídos: http://t.co/d4UDz3jD

Figure 5.1: Screenshot with an article cluster of four concurrently edited articles (ru, en, pt, ca). All breaking news criteria are fulfilled, the cluster is a breaking news candidate. Cross-language social network search results for en and pt can be seen.

5.3 Implementation Details

5.3.1 Wikipedia Recent Changes

As described earlier, our application monitors concurrent edit spikes on different language versions of Wikipedia. In the current implementation, we monitor *all* 285 different Wikipedias, 8 with ≥ 1,000,000 and 38 with ≥ 100,000 articles[1] including a long-tail of smaller Wikipedias. Changes to any single one article are communicated by a chat bot over Wikipedia's own Internet Relay Chat (IRC) server (`irc.wikimedia.org`),[2] so that parties interested in the data can listen to the changes as they happen. For each language version, there is a specific chat room following the pattern `"#" + language + ".wikipedia"`. For example, changes to Russian Wikipedia articles will be streamed to the room `#ru.wikipedia`. A special case is the room `#wikidata.wikipedia` for Wikidata [15], a platform for the collaborative acquisition and maintenance of structured data to be used by Wikimedia projects like Wikipedia. A sample chat message with the components separated by the asterisk character `'*'` announcing a change can be seen in the following. `"[[Juniata River]] http://en.wikipedia.org/w/index.php?diff=516269072&oldid=514-659029 * Johanna-Hypatia * (+67) Category:Place names of Native American origin in Pennsylvania"`. The message components are (i) article name, (ii) revision URL, (iii) Wikipedia editor handle, and (iv) change size and change description.

[1] `http://meta.wikimedia.org/wiki/List_of_Wikipedias`, accessed July 15, 2013

[2] `http://meta.wikimedia.org/wiki/IRC/Channels#Raw_feeds`, accessed July 15, 2013

5.3.2 Article Clusters

We cluster edits of articles about the same topic, but written in different languages, in article clusters. The example of the English `en:2013_Russian_meteor_event` and the corresponding Russian article `ru:Падение_метеорита_на_Урале_в_2013_году` that are both in the same cluster illustrate this. We use the Wikipedia API to retrieve language links for a given article. The URL pattern for the API is as follows. `http://$LANGUAGE.wikipedia.org/w/api.php?action=query&prop=langlinks&titles=$ARTICLE&format=json`. We work with the JSON representation.

5.3.3 Comparing Article Revisions

The Wikipedia API provides means to retrieve the actual changes that were made during an edit including additions, deletions, and modifications in a `diff`-like manner. The URL pattern is as follows. `http://$LANGUAGE.wikipedia.org/w/api.php?action=compare&torev=$TO&fromrev=$FROM&format=json`. This allows us to classify edits in categories like, *e.g.*, negligible trivial edits (punctuation correction) and major important edits (new paragraph for an article), which helps us to disregard seemingly concurrent edits in order to avoid false-positive alerts.

5.3.4 Breaking News Criteria

Our application *Wikipedia Live Monitor* puts detected article clusters in a monitoring loop in which they remain until their time-to-live (240 seconds) is over. In order for an article cluster in the monitoring loop to be identified as breaking news candidate, the following breaking news criteria have to be fulfilled.

≥ 5 Occurrences: An article cluster must have occurred in at least 5 edits.

≤ 60 Seconds Between Edits: An article cluster may have at maximum 60 seconds in between edits.

≥ 2 Concurrent Editors: An article cluster must have been edited by at least 2 concurrent editors.

≤ 240 Seconds Since Last Edit: An article cluster's last edit may not be longer ago than 240 seconds.

The exact parameters of the breaking news criteria above were *determined empirically* by analyzing Wikipedia edits over several hours and repeatedly adjusting the settings until major news events happening at the same time were detected. The resulting dataset split into three chunks has been made publicly available.[1]

5.3.5 Social Network Plausibility Checks

When a breaking news candidate has been identified, we use cross-language full-text social network searches on the social networks Twitter, Facebook, and Google+ as a plausibility check. As the *article titles* of all language versions of the particular article's cluster are know, we use these very article titles as search queries for cross-language searches, as can be seen in Figure 5.1. This approach greatly improves the recall of the social network search, however, requires either machine translation or an at least basic understanding of the languages being searched in. Currently the plausibility checking step is not yet fully automated, as the search results are for the time being meant to be consumed by *human evaluators*. Driven by (H1), we assume breaking news events are being discussed on social networks. We will show arguments for this assumption in section 5.4. For now, we expect social networks to be a short period ahead of Wikipedia. In consequence, if the human rater can find positive evidence for a connection between social network activities and Wikipedia edit actions, the breaking news candidate is confirmed to indeed represent breaking news.

5.3.6 Application Pseudocode

The *Wikipedia Live Monitor* application has been implemented in Node.js, a server side JavaScript software system designed for writing scalable Internet applications. Programs are created using event-driven, asynchronous input/output operations to minimize overhead and maximize scalability. Listing 5.1 shows the pseudocode of the two main event loops of the *Wikipedia Live Monitor* application. The actual implementation is based on Martyn Smith's Node.js IRC library[2] and the WebSockets API and protocol [4], wrapped by Guillermo Rauch's library Socket.IO.[3]

[1] `https://www.dropbox.com/sh/2qsg1zhb8p35fxf/Dghn55y0kh`, accessed July 15, 2013

[2] `https://github.com/martynsmith/node-irc`, accessed July 15, 2013

[3] `http://socket.io/`, accessed July 15, 2013

```
Input:  irc, listening on Wikipedia recent changes
Output:  breakingNewsCandidates, breaking news candidates

monitoringLoop = articleClusters = breakingNewsCandidates = {}

# Event loop 1:
# When a new message arrives
irc.on.message do (article)
  langRefs = getLanguageReferences(article)
  articleRevs = getArticleRevisions(article)
  cluster = clusterArticles(article, langRefs)

  # Create new cluster for previously unseen article
  if cluster not in monitoringLoop
    monitoringLoop.push(cluster)
    articleClusters.push(cluster)
    updateStatistics(cluster)
    emit.newCluster(cluster, articleRevs)
  # Update existing cluster, as the article was seen before
  else
    updateStatistics(cluster)
    emit.existingCluster(cluster, articleRevs)
    # Check breaking news criteria
    if cluster.occurrences >= 5
      if cluster.secsBetweenEdits <= 60
        if cluster.numEditors >= 2
          if cluster.secsSinceLastEdit <= 240
            socialNetworks.search(langRefs)
            breakingNewsCandidates.push(cluster)
            emit.breakingNewsCandidate(cluster)
          end if
        end if
      end if
    end if
  end if
  return breakingNewsCandidates
end do

# Event loop 2:
# Remove too old clusters regularly
timeout.every.240seconds do
  for each cluster in monitoringLoop
    if cluster.secsSinceLastEdit >= 240
      monitoringLoop.remove(cluster)
      articleClusters.remove(cluster)
    end if
  end for
end do
```

Listing 5.1: Two main event loops of the application

5.4 Evaluation

In subsection 5.1.2, we have set up three hypotheses. (H1) has been proven by Hu *et al.* in [5] for Twitter. We argue that it can be generalized to other social networks and invite the reader to have a look at our dataset, where the lively discussions about breaking news candidates on the considered social networks Twitter, Facebook, and Google+ support the argument. It is hard to prove (H2), as the concept of *important breaking news* is vague and dependent on one's personal background, however, all evidence suggests that (H2) indeed holds true, as, to the best of our knowledge and given our background, what the authors consider *important breaking news* is represented on at least one language version of Wikipedia. (H3) has been examined by Osborne *et al.* in [7]. In the paper, they suggest that Wikipedia lags about two hours behind Twitter. It has to be noted that they look at hourly accumulated page (article) *view* logs, where we look at realtime article *edit* log streams. Our experiments suggest that the lag time of two hours proposed by Osborne *et al.* may be too conservative. A conservative estimation at this stage is that the lag time for breaking news is more in the range of 30 minutes, and for global breaking news like celebrity deaths in the range of five minutes and less, albeit the edits by our experience will be small and iterative (*e.g.*, "X is a" to "X was a," or the addition of a death date), followed by more consistent thorough edits.

The (at time of writing) recent breaking news event of the resignation of *Pope Benedict XVI* helps respond to (Q1). The three first edit times of the Pope's English Wikipedia article[1] after the news broke on February 11, 2013 are as follows (all times in UTC): 10:58, 10:59, 11:02. The edit times of the French article[2] are as follows: 11:00, 11:00, 11:01. This implies that by looking at only two language versions of Wikipedia (the actual number of monitored versions is 285) of the Pope article, the system would have reported the news at 11:01. The official Twitter account of Reuters announced[3] the news at 10:59. Vatican Radio's announcement[4] was made at 10:57:47.

Not all breaking news events have the same global impact as the Pope's resignation, however, the proposed system was shown to work very reliably also for smaller events

[1] `http://en.wikipedia.org/w/index.php?title=Pope_Benedict_XVI&action=history`, accessed July 15, 2013

[2] `http://fr.wikipedia.org/w/index.php?title=Beno%C3%AEt_XVI&action=history`, accessed July 15, 2013

[3] `https://twitter.com/Reuters/status/300922108811284480`, accessed July 15, 2013

[4] `http://de.radiovaticana.va/Articolo.asp?c=663810`, accessed July 15, 2013

of more regional impact, for example, when *Indian singer Varsha Bhosle* committed suicide[1] on October 8, 2012. A systematic evaluation of (Q1) compulsorily can only be done by random samples, which has turned out positive results so far. Again, we invite the reader to explore our dataset and to conduct own experiments. A systematic evaluation of (Q2) requires a commonly shared dataset, which we have provided, however, at this point in time, we do not have access to the system of Osborne *et al.*

Regarding *Wikipedia Live Monitor*'s scalability, we already scale the monitoring system up to currently *all* 285 Wikipedias on a standard consumer laptop (mid-2010 MacBook Pro, 2.66 GHz Intel Core 2, 8 GB RAM), which proves the efficiency of the Node.js architecture for this kind of event-driven applications. In practice, the majority of the smaller Wikipedias being very rarely updated, we note that limiting ourselves to the Wikipedias with ≥ 100,000 articles results in no remarkable loss of recall.

5.5 Future Work

Future work will mainly address two areas. First, the *automated categorization of edits on Wikipedia* needs to be more fine-grained. In the context of breaking news detection, not all edits are equally useful. An image being added to an article is an example of an edit that usually will not be important. In contrast, the category "Living people" being removed from an article is a strong indicator of breaking (sad) news. Second, the *connection between social network search and Wikipedia edits* needs to be made clearer. In an initial step, the concrete changes to an article, as detailed in subsection 5.3.3, can be compared with social network microposts using a cosine similarity measure. More advanced steps can exploit the potential knowledge from Wikipedia edits (*e.g.*, category "Living people" removed implies a fatality).

5.6 Conclusions

In this chapter, we have shown an application called *Wikipedia Live Monitor* and released its source code under the Apache 2.0 license. This application monitors article edits on 285 different language versions of Wikipedia. It detects breaking news candidates according to well-defined breaking news criteria, whose exact parameters were

[1] `http://en.wikipedia.org/wiki/Varsha_Bhosle`, accessed July 15, 2013

determined empirically and the corresponding dataset made available publicly. We have shown how cross-language full-text social network searches are used as plausibility checks to avoid false-positive alerts. Concluding, our approach has revealed very promising results and actionable next steps in future work for improving the application.

Chapter Notes

This chapter is partly based on the following publications.

- Thomas Steiner, Seth van Hooland, and Ed Summers. "MJ no more: using concurrent wikipedia edit spikes with social network plausibility checks for breaking news detection". In: *Proceedings of the 22nd international conference on World Wide Web companion*. WWW '13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 791–794. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2488049`.

- Thomas Steiner, Ruben Verborgh, and Michael Hausenblas. "Crowdsourcing Event Detection in YouTube Videos". In: *Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011), Workshop in conjunction with the 10th International Semantic Web Conference 2011 (ISWC 2011), Bonn, Germany, October 23, 2011.* Ed. by Marieke van Erp, Willem Robert van Hage, Laura Hollink, Anthony Jameson, and Raphaël Troncy. Vol. 779. CEUR Workshop Proceedings ISSN 1613-0073. Oct. 2011, pp. 58–67. URL: `http://ceur-ws.org/Vol-779/derive2011_submission_8.pdf`.

References

[1] James Allan, Ron Papka, and Victor Lavrenko. "On-line new event detection and tracking". In: *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval.* SIGIR '98. Melbourne, Australia: ACM, 1998, pp. 37–45. ISBN: 1-58113-015-5.

[2] Claudine Beaumont. *Michael Jackson's death sparks Wikipedia editing war.* `http://bit.ly/Michael-Jacksons-death-sparks-Wikipedia-editing-war`, accessed July 15, 2013. June 2009.

[3] Mihai Georgescu, Nattiya Kanhabua, Daniel Krause, Wolfgang Nejdl, and Stefan Siersdorfer. "Extracting Event-related Information from Article Updates in Wikipedia". In: *Proceedings of the 35th European conference on Advances in Information Retrieval.* ECIR'13. Moscow, Russia: Springer-Verlag, 2013, pp. 254–266. ISBN: 978-3-642-36972-8.

[4] Ian Hickson. *The WebSocket API.* Candidate Recommendation. W3C, Sept. 2012.

[5] Mengdie Hu, Shixia Liu, Furu Wei, Yingcai Wu, John Stasko, and Kwan-Liu Ma. "Breaking News on Twitter". In: *Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems.* CHI '12. Austin, Texas, USA: ACM, 2012, pp. 2751–2754. ISBN: 978-1-4503-1015-4.

[6] M. Mestyán, T. Yasseri, and J. Kertész. "Early Prediction of Movie Box Office Success based on Wikipedia Activity Big Data". In: *Computing Research Repository* abs/1211.0970 (Nov. 2012).

[7] Miles Osborne, Saša Petrović, Richard McCreadie, Craig Macdonald, and Iadh Ounis. "Bieber no more: First Story Detection using Twitter and Wikipedia". In: *Proceedings of the SIGIR Workshop on Time-aware Information Access.* 2012.

[8] Saša Petrović, Miles Osborne, and Victor Lavrenko. "Streaming First Story Detection with Application to Twitter". In: *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics.* HLT '10. Los Angeles, California: Association for Computational Linguistics, 2010, pp. 181–189. ISBN: 1-932432-65-5.

[9] Thomas Steiner, Seth van Hooland, and Ed Summers. "MJ no more: using concurrent wikipedia edit spikes with social network plausibility checks for breaking news detection". In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW '13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 791–794. ISBN: 978-1-4503-2038-2. URL: http://dl.acm.org/citation.cfm?id=2487788.2488049.

[10] Thomas Steiner, Ruben Verborgh, and Michael Hausenblas. "Crowdsourcing Event Detection in YouTube Videos". In: *Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011), Workshop in conjunction with the 10th International Semantic Web Conference 2011 (ISWC 2011), Bonn, Germany, October 23, 2011.* Ed. by Marieke van Erp, Willem Robert van Hage, Laura Hollink, Anthony Jameson, and Raphaël Troncy. Vol. 779. CEUR Workshop Proceedings ISSN 1613-0073. Oct. 2011, pp. 58–67. URL: http://ceur-ws.org/Vol-779/derive2011_submission_8.pdf.

[11] Ed Summers. *An Ode to Node.* http://inkdroid.org/journal/2011/11/07/an-ode-to-node/, accessed July 15, 2013. Nov. 2011.

[12] Marijn ten Thij, Yana Volkovich, David Laniado, and Andreas Kaltenbrunner. "Modeling and predicting page-view dynamics on Wikipedia". In: *Computing Research Repository* abs/1212.5943 (2012).

[13] Manos Tsagkias, Maarten de Rijke, and Wouter Weerkamp. "Linking Online News and Social Media". In: *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining.* WSDM '11. Hong Kong, China: ACM, 2011, pp. 565–574. ISBN: 978-1-4503-0493-1.

[14] Brion Vibber. *Current events and traffic spikes.* http://blog.wikimedia.org/2009/06/25/current-events/, accessed July 15, 2013. June 2009.

[15] Denny Vrandečić. "Wikidata: A New Platform for Collaborative Data Collection". In: *Proceedings of the 21st International Conference Companion on World Wide Web.* WWW '12 Companion. Lyon, France: ACM, 2012, pp. 1063–1064. ISBN: 978-1-4503-1230-1.

[16] Yiming Yang, Tom Pierce, and Jaime Carbonell. "A study of retrospective and on-line event detection". In: *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval.* SIGIR '98. Melbourne, Australia: ACM, 1998, pp. 28–36. ISBN: 1-58113-015-5.

6

Media Item Extraction

6.1 Introduction

Before the rise of social networks, event coverage was mostly an affair of professional news agencies. The widespread availability of mobile phones with higher resolution cameras has transformed citizens into witnesses who are used to comment and share media illustrating events on social networks. Some examples with global impact include the shootings in Utøya,[1] which first appeared on Twitter, the capture and arrest of Muammar Gaddafi,[2] which first appeared on YouTube, or the emergency ditching of a plane in the Hudson river,[3] which first appeared on Twitpic. Some news communities[4] have even specialized in aggregating and brokering such user-generated content. Events, such as sports matches or concerts are largely illustrated by social media, albeit distributed over many social networks.

In this chapter, we tackle the challenge of reconciling social media data that illustrates known events, but that is spread over various social networks, all with the objective of creating visual event summaries. We propose a social-network-agnostic approach for the extraction of photos and videos covering events. We want to emphasize that in this chapter we do *not* put the focus on event detection (we have done that in Chapter 5). The events we are dealing with in this chapter were known beforehand and we use specific human-chosen search terms to find illustrating media.

[1] `http://en.wikipedia.org/wiki/2011_Norway_attacks`, accessed July 15, 2013

[2] `http://en.wikipedia.org/wiki/Death_of_Muammar_Gaddafi`, accessed July 15, 2013

[3] `http://en.wikipedia.org/wiki/US_Airways_Flight_1549`, accessed July 15, 2013

[4] `http://www.citizenside.com/`, accessed July 15, 2013

We first recall the definitions previously made in section 3.1 and add formal definitions of the terms *event*, *media item extraction*, *Application Programming Interface*, and *Web scraping.*

Social Network: A social network is an online service or media platform that focuses on building and reflecting relationships among people who share common interests and/or activities.

Media Item: A media item is defined as a photo[1] or video file that is publicly shared or published on at least one social network.

Micropost: A micropost is defined as a textual status message on a social network that can optionally be accompanied by a media item.

Event: An event is defined as a phenomenon that has happened or that is scheduled to happen. It is an observable occurrence grouping persons, places, times, and activities while being often documented by people through different media [11].

Media Item Extraction: The process of leveraging search functionalities of social networks to find references to media items, which allows for storing those media items in binary form.

Application Programming Interface (API): An API is a programmatic specification intended to be used as an interface by software components on client and server to communicate with each other.

Web scraping The term Web scraping means the process of automatedly extracting information from Web pages. Web scraping involves practical solutions based on existing technologies that are often entirely *ad hoc.* Examples of such technologies are regular expressions, Document Object Model (DOM) parsing [10], or CSS selectors [7]. The difference between *Web scraping* and the related concept of *screen scraping* is that screen scraping relies on the visual layout of a Web page, while Web scraping relies on the textual and/or hierarchical structure.

[1] We choose the term *photo* over the term *image* as Facebook, Twitter, and Google+ use it.

6.2 Related Work

Related work covers research that aims to collect, align, and organize media for trends or events. Liu *et al.* combine semantic inferencing and visual analysis to automatically find media to illustrate events [11]. They interlink large datasets of event metadata and media with the Linking Open Data Cloud [3, 6]. In [12], they show how visual summaries of past events providing viewers with a more compelling feeling of the event's atmosphere can be created based on a method to automatically detect and identify events from social media sharing websites. Approaches to alignment use visual, temporal, and spacial similarity measures to map multiple photo streams of the same events [20]. Other ways to collect and order media from social networks use media-driven metadata such as geospatial information [4]. Becker *et al.* show in [2] how to exploit the rich context associated with social media content, including user-provided annotations and automatically generated information. Using this rich context, they define similarity metrics to enable online clustering of media to events. In [1], the same authors develop recall-oriented query formulation strategies based on noisy event metadata from event aggregation platforms.

6.3 Social Networks and Media Items

Most social networks offer a search functionality that allows for content to be retrieved based on search terms, with or without more advanced search operators such as exclusion, inclusion, phrase search, *etc.* Each social network has special constraints regarding the supported search operators or filtering options.

Social networks are often perceived as *walled gardens* [15] due to the full control of the network operator over content and media on the social network in question, oftentimes accessible exclusively by social network members. This network lock-in effect was excellently illustrated by David Simonds in a cartoon that first appeared in the English-language weekly news and international affairs publication *The Economist*, reproduced in Figure 6.1. While some social networks (*e.g.*, Twitter) have full read and write access via specified APIs, other social networks (*e.g.*, Google+) currently only have read API access. In some cases, however, API access is limited, so that not all desired pieces of information is exposed (*e.g.*, view counts with Img.ly), which forces people interested in that data to fall back to Web scraping. It is to be noted that if the directives in the `robots.txt` file are respected, Web scraping *per se* is not an illegal practice, as only public information is being accessed, comparable to the level of access that common Web search engines have. The Robot Exclusion Standard, also referred to as `robots.txt` protocol, is a widely respected convention to prevent cooperating Web crawlers and other Web robots from accessing all or part of a website that is otherwise publicly viewable.

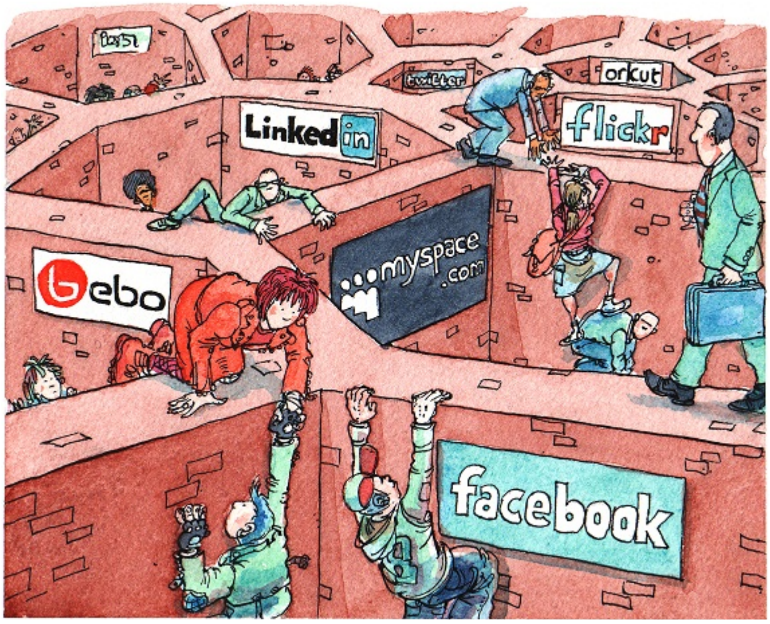


Figure 6.1: Social networks as walled gardens illustrated by David Simonds

6.4 Media Extractor

In this section, we first introduce a common data format that we have developed as an abstraction layer on top of the native data formats used by the considered social networks. We then explain the architecture of different kinds of media item extractors. Finally, we describe the processing steps that we apply to each extracted media item.

6.4.1 Abstraction Layer Data Format

Each social network uses a different data representation schema. While all social networks with API access are JSON-based [5], the differences in both supported social network features and media item support level, as was outlined in detail in section 3.2

and section 3.4, are also reflected in the returned JSON data. We therefore propose a common abstraction layer on top of the native data formats of all considered social networks. It is in the nature of any abstraction that it can only represent the greatest common divisor of all social networks. We show the abstraction layer in the following with the help of a concrete example, stemming from a query to the media extractor that will be explained in more detail in the upcoming subsection 6.4.2. The media extractor was used to query for media items that match the search term *hamburg*. Listing 6.1 shows sample output of the media extractor for a Facebook post, which was processed with named entity extraction and disambiguation as was detailed in Chapter 4.

`mediaUrl` Deep link to a media item

`posterUrl` Deep link to a thumbnail for photos or still frame for videos

`micropostUrl` Deep link to the micropost on the social network

`micropost` Container for a micropost

> `html` Text of the micropost, possibly with HTML markup
>
> `plainText` Text of the micropost with potential HTML markup removed
>
> `entities` Extracted and disambiguated named entities from the micropost text

`userProfileUrl` Deep link to the user's profile on the social network

`type` Type of the media item, can be `photo` or `video`

`timestamp` Number of milliseconds between 1 January 1970 00:00:00 UTC and the moment when the micropost was published

`publicationDate` Date in ISO 8601 format (YYYY-MM-DDTHH:MM:SSZ) when the micropost was published

`socialInteractions` Container for social interactions

> `likes` Number of times a micropost was liked, or `unknown`
>
> `shares` Number of times a micropost was shared, or `unknown`
>
> `comments` Number of comments a micropost received, or `unknown`
>
> `views` Number of views a micropost reached, or `unknown`

```json
{
  "mediaUrl": "http://video.ak.fbcdn.net/...",
  "posterUrl": "http://external.ak.fbcdn.net/...",
  "micropostUrl": "https://www.facebook.com/permalink.php?story_fbid=
    231781590231029&id=1254772464",
  "micropost": {
    "html": "Videoed between Hamburg and Snyder. Thought I would share.",
    "plainText": "Videoed between Hamburg and Snyder. Thought I would share.",
    "entities": [
      [
        {
          "name": "Hamburg",
          "relevance": 0.82274,
          "uri": "http://dbpedia.org/resource/Hamburg"
        },
        {
          "name": "Snyder",
          "relevance": 0.857,
          "uri": "http://dbpedia.org/resource/Snyder,_Texas"
        }
      ]
    ]
  },
  "userProfileUrl": "https://www.facebook.com/profile.php?id=1254772464",
  "type": "video",
  "timestamp": 1326371479000,
  "publicationDate": "2012-01-12T12:31:19Z",
  "socialInteractions": {
    "likes": 0,
    "shares": 0,
    "comments": 3,
    "views": null
  }
}
```

Listing 6.1: Sample output of the media extractor showing a Facebook post processed with named entity extraction and disambiguation (slightly shortened for legibility)

6.4.2 Media Item Extractors

We have developed a combined media extractor composed of separate media item extractors for the seven social networks Google+, Myspace, Facebook, Twitter, Instagram, YouTube, and Flickr, with additional support for the media sharing platforms Img.ly, Imgur, Lockerz,[1] Yfrog, MobyPicture, and Twitpic. The media extractor takes as input

[1] Dysfunctional since April 2013 when the service shut down its API access

a search term that is relevant to a known event, *e.g.*, the term *boston celtics* for a recent match of the Basketball team Boston Celtics. This search term gets forwarded to the search APIs of all social networks in parallel. Each social network has a 30 seconds timeout window to deliver its results. When the timeout is reached or when all social networks have responded, the available results are aligned according to the data format defined in subsection 6.4.1. Media items and the relevant metadata like view count, comments, *etc.* are retrieved either directly or via Web scraping. For some social networks, *e.g.*, Img.ly, a combination of Web scraping and API access is required since the API does not return all necessary fields of our data format. While we could default to Web scraping to obtain all relevant data, it is more robust to use API access wherever possible and only fall back to the more brittle Web scraping for the parts not covered by API access.

Special Role of Twitter: Twitter (subsection 3.2.12) plays a special role, as it can be used as a third-order support social network, as was detailed previously in section 3.4. This means that the micropost text is located on Twitter, but the referenced media items are located on third-party media platforms. Due to the length limitation for tweets of 140 characters, short URLs are used on the service. We search for the search term in question (*e.g.*, following up from the example before, *boston celtics*), but combine it with the short URL domain parts of the media platforms. For example, the short domain URL of the social network Flickr (subsection 3.2.2) is `flic.kr`, where the long domain URL is `flicker.com`. The short domain URL of Instagram (subsection 3.2.6) is `instagr.am`, where the long domain URL is `instagram.com`, *etc.* We have created a list of all known short domain URLs for the considered media platforms so that the complete search query for Twitter is the actual search term, combined with this list of short domain URLs:

boston celtics AND (flic.kr OR instagr.am OR ...)

The complete data flow is illustrated in the architectural diagram in Figure 6.2. As a side note, Twitter on its website now has its own media extractor based on Twitter Cards [18] with support for some of of the media platforms, however, our own media extractor goes beyond Twitter's offer, especially since Facebook-owned Instagram's latest break-up with Twitter.[1]

[1] `http://techcrunch.com/2012/12/05/kevin-systrom-on`, accessed July 15, 2013

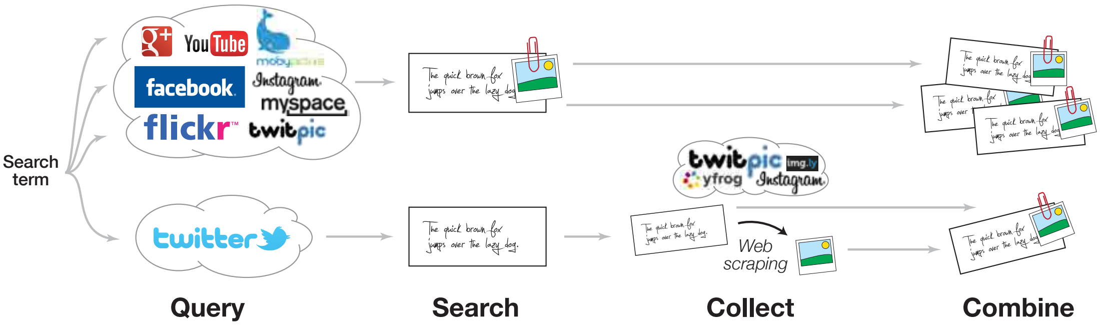


Figure 6.2: Overview of the media extractor: hybrid approach to the media item extraction task using a combination of API access and Web scraping

6.5 Evaluation

We have run experiments in the time period of January 10 to 19, 2012, during which we have randomly selected nine events that received broad social media coverage. For these events, we have collected media items and microposts using our media extractor. In the following, we will provide a short summary of the nine selected events in order to give the reader the necessary background knowledge.

Assad Speech On January 10, 2012, Syrian President Bashar al-Assad delivered a televised talk defending his government's actions and motivations, despite world pressure on his government for its 10-month crackdown on protesters. Activists say the operation has led to nearly 6,000 or more estimated deaths.[1]

CES Las Vegas The International Consumer Electronics Show (CES) is a major technology-related trade show held each January in the Las Vegas Convention Center. Not open to the public, the Consumer Electronics Association-sponsored show typically hosts previews of products and new product announcements. CES Las Vegas took place from January 11 to 13, 2012.[2]

Cut the Rope Launch: On January 10, 2012 during Steve Ballmer's final keynote at the International Consumer Electronics Show, the HTML5 version of the popular mobile game *Cut the Rope* was announced. This is a sub-event of CES Las Vegas.[3]

[1] `http://www.cnn.com/2012/01/10/world/meast/syria-unrest/`, accessed July 15, 2013

[2] `http://www.cesweb.org/`, accessed July 15, 2013

[3] `http://ces.cnet.com/8301-33377_1-57356403/`, accessed July 15, 2013

Ubuntu TV Launch: Ubuntu TV by Canonical, based on the user interface Unity, is a variant of the Ubuntu operating system, designed to be a Linux distribution specially adapted for embedded systems in televisions. It was announced by Canonical on January 10, 2012, at CES.[1]

Costa Concordia Disaster The Costa Concordia is an Italian cruise ship that hit a reef and partially sank on January 13, 2012 off the Italian coast. The vessel ran aground at Isola del Giglio, Tuscany, resulting in the evacuation of 4,211 people.[2]

Dixville Notch Dixville Notch is an unincorporated village in Dixville township of Coos County, New Hampshire, USA, best known in connection with its long-standing middle-of-the-night vote in the U.S. presidential election. In a tradition that started in the 1960 election, all the eligible voters in Dixville Notch gather at midnight in the ballroom of The Balsams. This year, on January 10, 2012, the voters cast their ballots and the polls officially closed one minute later.[3]

Free Mobile Launch Free Mobile is a French mobile broadband company, part of the Iliad group. On January 10, 2012, a long-awaited mobile phone package for 19.99 € with calls included to 40 countries, texts, multimedia messages and Internet was announced by the Iliad group's Chief Strategy Officer Xavier Niel.[4]

Blackout SOPA The Stop Online Piracy Act (SOPA) is a bill of the United States proposed in 2011 to fight online trafficking in copyrighted intellectual property and counterfeit goods. On January 18, the English Wikipedia, and several other Internet companies coordinated a service blackout to protest SOPA and its sister bill, the Protect IP Act (PIPA). Other companies, including Google, posted links and photos in an effort to raise awareness.[5]

Christian Wulff Case Since December 2011, former German President Christian Wulff faces controversy over discrepancies in statements about a loan while being governor of Lower Saxony. It was revealed that he had applied pressure on Springer

[1] `http://www.theverge.com/2012/1/9/2695387/ubuntu-tv-video-hands-on`, accessed July 15, 2013

[2] `http://en.wikipedia.org/wiki/Costa_Concordia_disaster`, accessed July 15, 2013

[3] `http://www.washingtonpost.com/2012/01/09/gIQANslKnP_story.html`, accessed July 15, 2013

[4] `http://www.nytimes.com/2012/01/11/technology/iliad-takes-aim-at-top-mobile-operators-in-france.html`, accessed July 15, 2013

[5] `http://sopablackout.org/learnmore/`, accessed July 15, 2013

> Press to delay revelations on the issue until he was back from a visit abroad. When Wulff found out that a tabloid was going to break the story, he left a message on their voice mail in which he threatened to take legal action.[1]

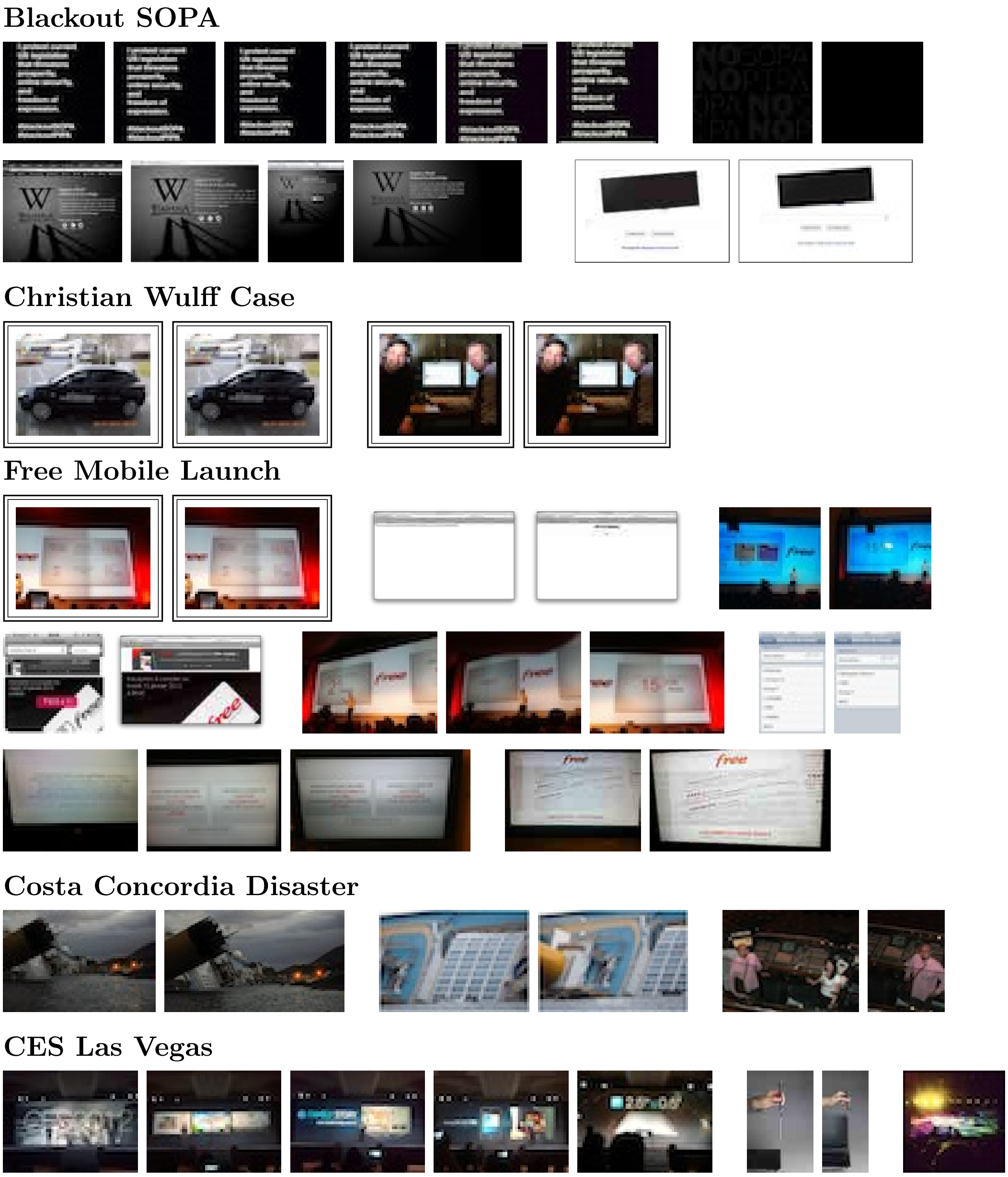
6.5.1 Dataset

Our data set contained 448 photos with an average file size of ~0.7MB and 143 videos. Some videos are no longer available due to either account termination or video takedown by the user (Assad, Dixville). Table 6.1 shows the total numbers of retrieved photos and videos of the media extractor. Table cell values marked with n+ signify that there were more results, but that only n results were considered. We have calculated the precisions for each event for both video and photo separately; the overall photo precision was 0.73, and the overall video precision was 0.54. We note that these values were calculated *before* any pruning step, *i.e.*, before taking into account the additional textual information from microposts like potential extracted named entities. The dataset is very diverse with respect to photo quality, photo format, and naturally, content. It ranges from entirely sharp screenshots in all sorts of formats (*e.g.*, screenshots of the Google homepage for the Blackout SOPA event to screenshots of a wide banner advertisement), over to blurry cell phone photos in standard photo formats (*e.g.*, photos of the stage for the Free Mobile Launch event). Figure 6.3 shows sample photos for some of the considered nine events. We have observed that more than one search session with different combinations of search terms [1, 2] is necessary in order to obtain a satisfactory recall. Query strategies developed by Becker [1] that combine different combinations of event title, event venue, and event city work consistently well.

6.5.2 The Need for Media Item Deduplication

Given our broad approach to retrieve media items across multiple social networks, we observed many exact-duplicate or near-duplicate media items. Oftentimes, these duplicates stem from users who cross-post to several social networks. Instead of trying to filter out cross-posted items, we rather keep them and cluster them. We are especially interested in social interactions that media items can trigger. For example, if one and the same photo is cross-posted to separate networks, it can retrieve shares, likes, views,

[1] `http://www.spiegel.de/international/germany/0,1518,804631,00.html`, accessed July 15, 2013



Blackout SOPA

Christian Wulff Case

Free Mobile Launch

Costa Concordia Disaster

CES Las Vegas

Figure 6.3: Sample photos for some of the considered nine events (showing only exact- or near-duplicate media items)

or comments independently on each of those networks. By clustering media items, we get a higher-level view on a media item cluster's overall performance on different networks. We also observed media items that were near-duplicates, for example, from people who attended the same event like a concert and who took photos of the stage from almost the same angle. Similar to exact-duplicates, by clustering near-duplicate media items, we can treat them like exact-duplicates to get the same network-agnostic viewpoint. We will examine reasons for exact-duplicate and near-duplicate media item content and ways to deal with it in Chapter 8.

6.5.3 The Need for Ranking Media Items

Our ultimate goal is to generate media galleries that *visually* and *audially* summarize events. Especially given high-recall search terms, we need a way to rank and prune media items. Popular media items can be displayed bigger, longer, or with a special decoration like a thicker border in comparison to less popular media items. For videos, the audio part poses a challenge. In our experiments, we observe that intermixing the audio of all videos of an event often generates a very characteristic *noise cloud* that *audially* conveys the event's atmosphere very well. A good example is the Assad Speech event, where a mix of Arabic voices blends nicely with the speech of a US politician. A different example is the CES Las Vegas event, where the atmosphere of a big exposition with music, announcements, and technical analysis becomes alive. We will have a closer look at media item ranking in Chapter 9.

6.6 Conclusions

In this chapter, we have presented a generic media extractor for extracting media items shared on social networks to illustrate known events. We have proposed a common abstraction layer on top of the social networks' native data formats to align search results. Our approach to extracting media items and associated textual microposts covers already most of the Western world's social networks. Context-aware multimedia analysis will bring a new range of parameters into play since many media items contain a message that is complementary to the text. For example, facial detection [17] and eventually recognition [19] can signify the presence of specific people in a media item. Optical Character Recognition (OCR) can generate additional textual signals from media items. As visual recognition systems grow more powerful, more objects will eventually be recognizable by machines [14], which would allow for generating *visual hashtags* that describe the content *inside* of the media item. Extracted features in all three categories (*textual*—from the micropost, *visual*—from the media item, and *social*—from the social network in the form of social interactions) can serve as ranking criteria, be it in isolation or in combination by introducing a ranking formula. As a result, this will also positively influence the diversity of automated summarizations.

Nonetheless, it remains important to view the media and the accompanying microposts as a whole, since the text could convey a sentiment about, or an explanation of the visual data. Using named entity recognition as outlined in Chapter 4, the important semantic elements in the micropost get identified. The content of the message can subsequently be used to narrow down the search space for visual factors enabling cross-fertilization between the textual and visual analysis, which results in effective context-aware analysis possibilities [13, 16]. Finally, by leveraging the *LOD* cloud, we can use that knowledge to get a more diverse view on events. At time of writing, the so-called *Operation Pillar of Defense*[1] by the Israeli armed forces causes ongoing conflicts between Palestinians and Israelis. Using the LOD cloud, promising search terms like, for example, *gaza*, can be easily looked up in different languages like Hebrew or Arabic. In practice, these additional search terms return interesting new media items that a pure monolingual search would not have revealed—oftentimes, and especially in the concrete case, at the expense of neutrality. We are confident that the additional coverage from more angles helps sharpen one's own viewpoint of an event, especially with the option of translating microposts authored in foreign languages, which is supported by our approach.

[1] `http://en.wikipedia.org/wiki/Operation_Pillar_of_Defense`, accessed July 15, 2013

Social	Assad		CES		Concordia		Dixville		Free		Ropes		SOPA		Ubuntu		Wulff	
Network	**P**	**V**	**P**	**V**	**P**	**V**	**P**	**V**	**P**	**V**	**P**	**V**	**P**	**V**	**P**	**V**	**P**	**V**
Google+	3	2	5	3	15	1	4	1	6	0	5	1	5	0	6	1	7	0
Myspace	0	0	0	0	10+	0	9	0	1	0	6	0	0	0	0	0	8	0
Facebook	0	0	0	1	0	1	0	0	0	0	0	0	0	2	0	0	0	0
Twitter	2	0	2	0	3	0	3	0	2	0	4	0	5	0	0	0	2	0
Instagram	0	0	20+	0	20+	0	0	0	20+	0	20+	0	20+	0	0	0	2	0
YouTube	0	10+	0	10+	0	10+	0	3	0	10+	0	10+	0	10+	0	10+	0	10+
Flickr	10+	0	10+	6	10+	10+	10+	10+	10+	0	10+	10+	10+	0	10+	9	10+	2
MobyPic	0	0	1	0	4	0	0	0	2	0	20+	0	1	0	2	0	3	0
Twitpic	0	0	20+	0	18	0	1	0	20+	0	20+	0	19	0	2	0	20+	0
Total	15	12	58	20	80	22	27	14	61	10	85	21	60	12	20	20	52	12
Relevant	12	7	39	18	61	15	8	2	46	4	76	14	43	5	18	13	39	7
Precision	.80	.58	.67	.90	.76	.55	.30	.14	.75	.40	.89	.67	.71	.42	.90	.65	.75	.58

Table 6.1: Number of photos and videos collected for nine events happening between January 10–19, 2012 grouped by social networks, separated in photo (P) and video (V) results. Overall **photo precision: 0.73**. Overall **video precision: 0.54**. Note that this is before post-processing.

Chapter Notes

This chapter is partly based on the following publications.

- Giuseppe Rizzo, Thomas Steiner, Raphaël Troncy, Ruben Verborgh, José Luis Redondo García, and Rik Van de Walle. "What Fresh Media Are You Looking For?: Retrieving Media Items From Multiple Social Networks". In: *Proceedings of the 2012 International Workshop on Socially-aware Multimedia.* SAM '12. Nara, Japan: ACM, 2012, pp. 15–20. ISBN: 978-1-4503-1586-9. URL: `http://www.eurecom.fr/~troncy/Publications/Troncy-saw12.pdf`.

- Houda Khrouf, Ghislain Atemezing, Giuseppe Rizzo, Raphaël Troncy, and Thomas Steiner. "Aggregating Social Media for Enhancing Conference Experience". In: *Real-Time Analysis And Mining of Social Streams, Papers from the 2012 ICWSM Workshop.* Ed. by Arkaitz Zubiaga, Maarten de Rijke, Markus Strohmaier, and Mor Naaman. AAAI Technical Report WS-12–02. June 2012. URL: `http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/download/4779/5086`.

- Houda Khrouf, Ghislain Atemezing, Thomas Steiner, Giuseppe Rizzo, and Raphaël Troncy. *Confomaton: A Conference Enhancer with Social Media from the Cloud.* 2012. URL: `http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_343.pdf`.

References

[1] Hila Becker, Dan Iter, Mor Naaman, and Luis Gravano. "Identifying Content for Planned Events Across Social Media Sites". In: *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. WSDM '12. ACM, 2012, pp. 533–542.

[2] Hila Becker, Mor Naaman, and Luis Gravano. "Learning Similarity Metrics for Event Identification in Social Media". In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. WSDM '10. ACM, 2010, pp. 291–300.

[3] Chris Bizer, Anja Jentzsch, and Richard Cyganiak. *State of the LOD Cloud*. `http://wifo5-03.informatik.uni-mannheim.de/lodcloud/state/`, accessed July 15, 2013. 2011.

[4] David J. Crandall, Lars Backstrom, Daniel Huttenlocher, and Jon Kleinberg. "Mapping the World's Photos". In: *Proceedings of the 18th International Conference on World Wide Web*. WWW '09. ACM, 2009, pp. 761–770.

[5] Douglas Crockford. *Introducing JSON*. `http://json.org/`, accessed July 15, 2013. 2006.

[6] Richard Cyganiak and Anja Jentzsch. *The Linking Open Data cloud diagram*. `http://lod-cloud.net/`, accessed July 15, 2013. 2011.

[7] Lachlan Hunt and Anne van Kesteren. *Selectors API Level 1*. Working Draft. W3C, 2012.

[8] Houda Khrouf, Ghislain Atemezing, Giuseppe Rizzo, Raphaël Troncy, and Thomas Steiner. "Aggregating Social Media for Enhancing Conference Experience". In: *Real-Time Analysis And Mining of Social Streams, Papers from the 2012 ICWSM Workshop*. Ed. by Arkaitz Zubiaga, Maarten de Rijke, Markus Strohmaier, and Mor Naaman. AAAI Technical Report WS-12–02. June 2012. URL: `http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/download/4779/5086`.

[9] Houda Khrouf, Ghislain Atemezing, Thomas Steiner, Giuseppe Rizzo, and Raphaël Troncy. *Confomaton: A Conference Enhancer with Social Media from the Cloud*. 2012. URL: `http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_343.pdf`.

[10] Arnaud Le Hors, Philippe Le Hégaret, Lauren Wood, Gavin Nicol, Jonathan Robie, Mike Champion, et al. *Document Object Model (DOM) Level 3 Core Specification, Version 1.0*. Recommendation. W3C, 2004.

[11] Xueliang Liu, Raphaël Troncy, and Benoit Huet. "Finding Media Illustrating Events". In: *Proceedings of the 1st ACM International Conference on Multimedia Retrieval.* ICMR '11. ACM, 2011, pp. 1–8.

[12] Xueliang Liu, Raphaël Troncy, and Benoit Huet. "Using Social Media to Identify Events". In: *Proceedings of the 3rd ACM SIGMM International Workshop on Social Media.* WSM '11. 2011, pp. 3–8.

[13] Giuseppe Rizzo, Thomas Steiner, Raphaël Troncy, Ruben Verborgh, José Luis Redondo García, and Rik Van de Walle. "What Fresh Media Are You Looking For?: Retrieving Media Items From Multiple Social Networks". In: *Proceedings of the 2012 International Workshop on Socially-aware Multimedia.* SAM '12. Nara, Japan: ACM, 2012, pp. 15–20. ISBN: 978-1-4503-1586-9. URL: `http://www.eurecom.fr/~troncy/Publications/Troncy-saw12.pdf`.

[14] Thomas Serre, Lior Wolf, Stanley Bileschi, Maximilian Riesenhuber, and Tomaso Poggio. "Robust Object Recognition with Cortex-Like Mechanisms". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.3 (2007), pp. 411–426.

[15] David Simonds. *Everywhere and nowhere.* `http://www.economist.com/node/10880936`, accessed July 15, 2013. 2008.

[16] Ruben Verborgh, Davy Van Deursen, Erik Mannens, Chris Poppe, and Rik Van de Walle. "Enabling Context-aware Multimedia Annotation by a Novel Generic Semantic Problem-solving Platform". In: *Multimedia Tools and Applications* 61.1 (2012), pp. 105–129.

[17] Paul Viola and Michael J. Jones. "Robust Real-Time Face Detection". In: *International Journal of Computer Vision* 57.2 (2004), pp. 137–154.

[18] Tian Wang. *Search for a new perspective.* `http://blog.twitter.com/2012/11/search-for-new-perspective.html`, accessed July 15, 2013. 2012.

[19] John Wright, Allen Y. Yang, Arvind Ganesh, S. Shankar Sastry, and Yi Ma. "Robust Face Recognition via Sparse Representation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.2 (2009), pp. 210–227.

[20] Jianchao Yang, Jiebo Luo, Jie Yu, and Thomas S. Huang. "Photo Stream Alignment for Collaborative Photo Collection and Sharing in Social Media". In: *Proceedings of the 3rd ACM SIGMM International Workshop on Social Media.* WSM '11. ACM, 2011, pp. 41–46.

7

Camera Shot Boundary Detection

7.1 Introduction

In the previous chapter, we have motivated the need for deduplication of exact-duplicate and near-duplicate media items. This chapter focuses on camera shot boundary detection, which is a first step towards media item deduplication for videos and photos contained in videos. In video production and filmmaking, a *camera shot* is a series of frames that runs for an uninterrupted period of time. Shots are always filmed with a single camera and can be of any duration. Shot boundary detection (also called *cut detection*, *shot transition detection*, or simply *shot detection*) is a field of research of video processing. Its subject is the automated detection of transitions between shots with hard or soft cuts as the boundaries in digital video, with the purpose of temporal segmentation of videos.

In this chapter, we present a browser-based, client-side, and on-the-fly approach to this challenge based on modern HTML5 [2] Web APIs. Once a video has been split into shots, shot-based video navigation becomes possible, more fine-grained playing statistics can be created, and finally, shot-based video comparison is possible. The algorithm developed in the context of our research has been incorporated in a browser extension so that it can run transparently on a major online video portal. Figure 7.1 shows detected camera shots for a sample video.

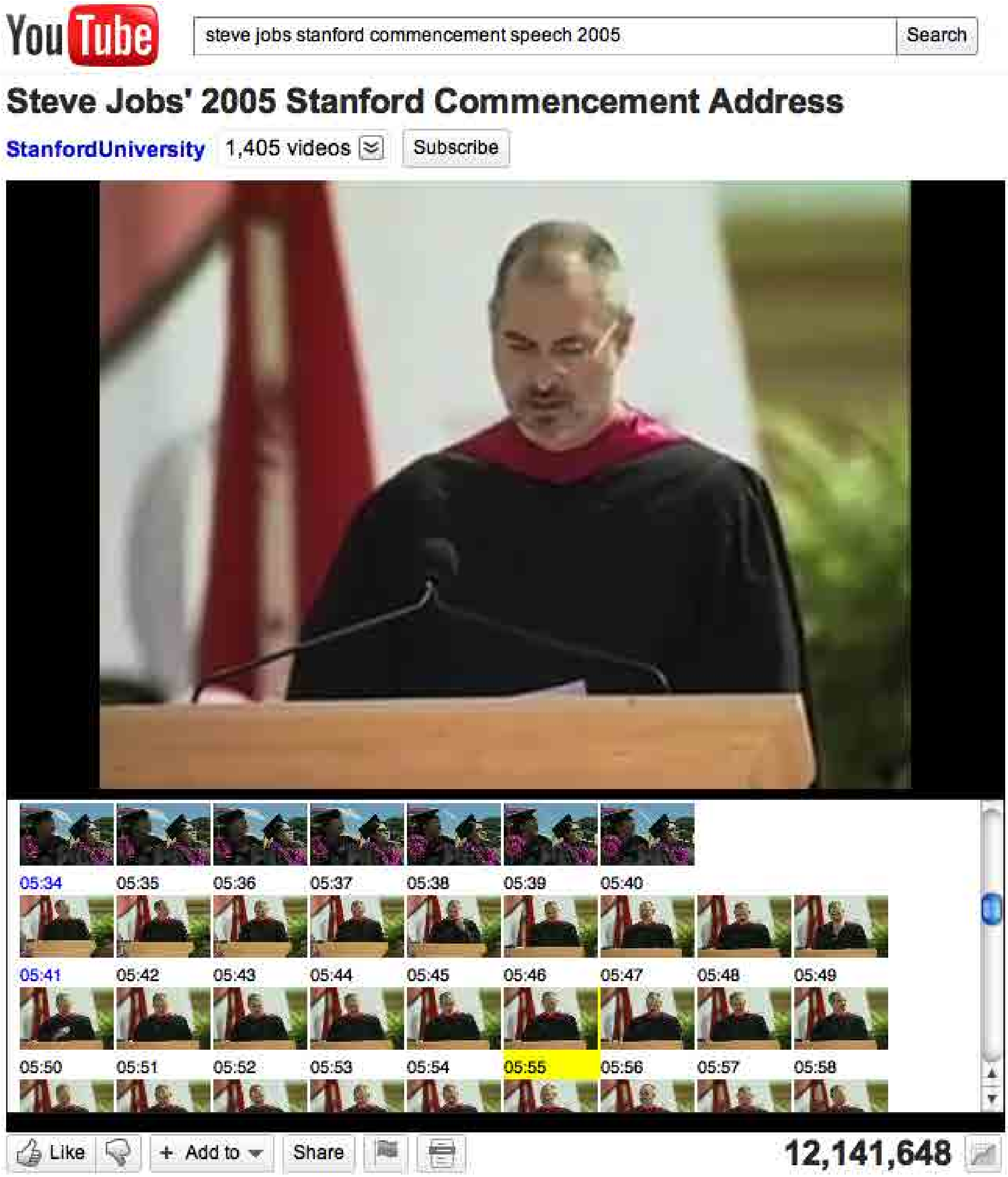


Figure 7.1: Camera shots for a sample video on a major online video portal, detected on-the-fly via our shot boundary algorithm incorporated in a browser extension

7.2 Related Work

As outlined before, video fragments consist of shots, which are sequences of consecutive frames from a single viewpoint, representing a continuous action in time and space. The topic of shot boundary detection has already been described extensively in literature. While some specific issues still remain open (notably detecting gradual transitions and detected false-positives due to large movement or illumination changes), the problem is considered resolved for many cases [6, 12]. Below, we present an overview of several well-known categories of shot boundary detection techniques.

Pixel Comparison Methods: Pixel comparison methods [5, 14] construct a discontinuity metric based on differences in color or intensity values of corresponding pixels in successive frames. This dependency on spatial location makes this technique very sensitive to (even global) motion. Various improvements have been suggested, such as prefiltering frames [15], but pixel-by-pixel comparison methods proved inferior, which has steered research towards other directions.

Histogram Analysis: A related method to pixel comparison methods is histogram analysis [9], where changes in frame histograms are used to justify shot boundaries. Their insensitivity to spatial information within a frame makes histograms less prone to partial and global movements in a shot.

Hybrid Approaches: As a compromise, a third group of methods consists of a trade-off between the above two categories [1]. Different histograms of several, non-overlapping blocks are calculated for each frame, thereby categorizing different regions of the frame with their own color-based, space-invariant fingerprint. The results are promising, while computational complexity is kept to a minimum, which is why we have chosen to base our algorithm on a variation of this approach.

Comparison of Mean and Standard Deviations: Other approaches to shot boundary detection include the comparison of mean and standard deviations of frame intensities [8]. Detection using other features such as edges [13] and motion [3] have also been proposed. Edge detection transforms both frames to edge pictures, *i.e.*, it extracts the probable outlines of objects. However, Gargi *et al.* have shown that these more complex methods do not necessarily outperform histogram-based approaches [4]. A detailed comparison can be found in Yuan *et al.*[12].

7.3 On-the-fly Shot Boundary Detection Algorithm

As outlined in the previous section, shot boundary detection is mostly considered a solved problem and many efficient approaches exist. However, to the best of our knowledge, none of the proposed solutions deals with the specific issue of detecting camera shots in *streaming* video in the context of a Web browser and on-the-fly. Streaming (HTML5) video has no notion of frames, but only allows for time-based navigation via the `currentTime` attribute. The algorithm we propose in the sequence of this chapter deals effectively with these limitations and we also show that it works efficiently.

7.3.1 Details of the Algorithm

In this section, we discuss our shot boundary detection algorithm, which falls in the category of histogram-based algorithms. Since visually dissimilar video frames can have similar global histograms, we take local histograms into account instead. We therefore split video frames in freely configurable rows and columns, *i.e.*, lay a grid of tiles over each frame. The user interface that can be seen in Figure 7.2 currently allows for anything from a *1* × *1* grid to a *20* × *20* grid. The limits are imposed by the reasonable processing time on consumer PCs. For each step, we examine a frame *f* and its direct predecessor *f* – *1* and calculate their tile histograms. We recall that HTML5 streaming video has no notion of frames, so by frame we mean a frame that we have navigated to via setting the `currentTime` attribute. Apart from the per-tile average histogram distance, the frame distance function further considers a freely configurable number of *most different* and *most similar* tiles. This is driven by the observation that different parts of a video have different intensities of color changes, dependent on the movements from frame to frame. The idea is thus to boost the influence of movements in the frame distance function, and to limit the influence of permanence. In the debug view of our approach that can be seen in Figure 7.2, blue boxes indicate movements, while red boxes indicate permanence. In the concrete example, Steve Jobs' head and shoulders move as he talks, which can be clearly seen thanks to the blue boxes in the particular tiles. Additional movements come from a swaying flag on the left, and a plant on the right. In contrast, the speaker desk, the white background, and the upper part of his body remain static, resulting in red boxes. We use a grid layout of 20×20 tiles ($nTiles = 400$), and a $tileLimit = 133 = 20 \times 20 * 1/3$ of most different or similar tiles, *i.e.*, we treat one third of all tiles as most different tiles, one third as normal tiles, and one third as most similar tiles, and apply boosting and limiting factors to the most different and most similar tiles respectively. We work with values of 1.1 for the *boostingFactor*, which slightly increases the impact of the most different tiles, and 0.9 for the *limitingFactor*, which slightly decreases the impact of the most similar tiles. These algorithm parameters were empirically determined to deliver solid results on a large corpus of videos, albeit for each individual video the optimal settings can be manually tweaked to take into account the particular video's special characteristics. The algorithm pseudocode can be seen in Listing 7.1.

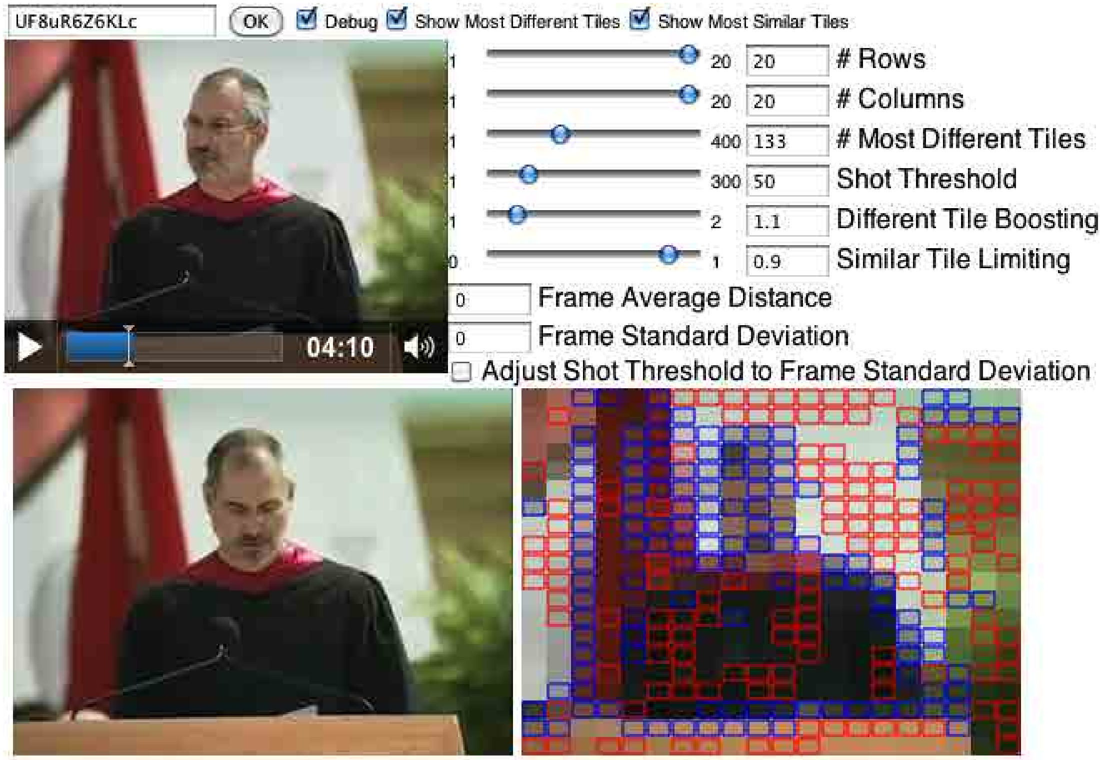
We define the average histogram distance between two frames f and $f - 1$ as $avgHisto_f$. In a first step, we have examined the histogram distance data statistically and observed that while the overall average frame distance $avgDist_f$, defined as

$$avgDist_f = \frac{1}{nTiles} \sum_{t=1}^{nTiles} avgHisto_{f,t}$$

is very intuitive to human beings, far more value lies in the standard deviation $stdDev_f$, based on the definition of the overall average frame distance $avgDist_f$

$$stdDev_f = \sqrt{\frac{1}{nTiles} \sum_{t=1}^{nTiles} (avgHisto_{f,t} - avgDist_f)^2}$$

We use the standard deviation as a value for the shot splitting threshold [8] to obtain very accurate shot splitting results. We found the boosting and limiting factors to have an overall positive quality impact on more lively videos and a negative quality impact on more monotone videos. Optimal results can be achieved if, after changing either the boosting or the limiting factors for the most similar or different tiles, the value of the shot splitting threshold is adapted to the new resulting standard deviation. The user interface can optionally do this automatically.



UF8uR6Z6KLc ☑ Debug ☑ Show Most Different Tiles ☑ Show Most Similar Tiles

1 20 20 # Rows

1 20 20 # Columns

1 400 133 # Most Different Tiles

1 300 50 Shot Threshold

1 2 1.1 Different Tile Boosting

0 1 0.9 Similar Tile Limiting

0 Frame Average Distance

0 Frame Standard Deviation

□ Adjust Shot Threshold to Frame Standard Deviation

Figure 7.2: Debug view of the shot boundary detection process. Blue boxes highlight tiles with most differences to the previous frame, red boxes those with most similarities.

```
for frame in frames
  f = frame.index
  for tile in tiles of frame
    avgHisto[f][tile] = getTilewiseDiff()

  mostDiffTiles = getMostDiffTiles(avgHisto[f])
  mostSimTiles = getMostSimTiles(avgHisto[f])

  for tile in tiles of frame
    factor = 1
    if tile in mostDiffTiles
      factor = boostingFactor
    else if tile in mostSimTiles
      factor = limitingFactor
    avgHisto[f][tile] = avgHisto[f][tile] * factor
  avgDist[f] = avg(avgHisto[f])
```

Listing 7.1: Pseudocode of the shot boundary detection algorithm

7.3.2 Implementation Details

The complete video analysis process happens fully on the client side. We use the HTML5 JavaScript APIs of the `<video>` and `<canvas>` tags. In order to obtain a video still frame from the `<video>` tag at the current video position, we use the `drawImage()` function of the 2D context of the `<canvas>` tag, which accepts a video as its first parameter. We then analyze the video frame's pixels per tile and calculate the histograms. In order to retrieve the tile-wise pixel data from the 2D context of the `<canvas>`, we use the `getImageData()` function. For processing speed reasons, we currently limit our approach to a resolution of one second, *i.e.*, for each analysis step, seek the video in *1s* steps. We then calculate the frame distances as outlined in section 7.3. For each frame, we can optionally generate an `<img>` tag with a base64-encoded data URI representation of the video frame's data that can serve for filmstrip representations of the video.

We have implemented the shot boundary detection algorithm as a stand-alone Web application and as a browser extension for the popular video hosting platform YouTube. Browser extensions are small software programs that users can install to enrich their browsing experience with their browser. They are typically written using a combination of standard Web technologies, such as HTML, JavaScript, and CSS. There are several types of extensions; for this work we focus on extensions based on so-called *content scripts*. Content scripts are JavaScript programs that run in the context of Web pages via dynamic code injection. By using the standard Document Object Model (DOM) [7], they can modify details of Web pages.

7.4 Evaluation

On-the-fly shot detection in streaming video comes with its very own challenges that were briefly outlined before. First, it is a question of streaming speed. Especially with High Definition (HD) video, this can be very demanding. We do not attach the analysis `<video>` and `canvas` tags to the DOM tree [7] so that the browser does not have to render them and thus can save some CPU cycles, however, the video playing logic still has to seek the video position ahead in one-second steps and process the encountered still frame. Even on a higher-end computer (our experiments ran on a MacBook Pro, Intel Core 2 Duo 2,66 GHz, 8 GB RAM), the process of in parallel analyzing and displaying a *1280* × *720* HD video of media type *video/mp4; codecs="avc1.64001F, mp4a.40.2"* caused an average CPU load of about 70%. The HTML5 [2] specification states that *"when the playback rate is not exactly 1.0, hardware, software, or format limitations can cause video frames to be dropped."* In practice, this causes the analysis environment to be far from optimal. In our experiments we differentiated between false-positives, *i.e.*, shot changes that were detected, but not existent, and misses, *i.e.*, shot changes that were existent, but not detected. Compared to a set of videos with manually annotated shot changes, our algorithm detected fewer false-positives than misses. The reasons were gradual transitions and shots shorter than one second (below our detection resolution) for misses, and large movements in several tiles for false-positives. Overall, we reached an accuracy of about 86%, which is not optimal, but given the challenges sufficient for our use case of detecting near- or exact-duplicate videos.

7.5 Future Work

There is potential for optimization of the analysis speed by dynamically selecting lower quality analysis video files, given that videos are oftentimes available in several resolutions, like Standard Definition (SD) or High Definition (HD). We have checked in how far analysis results differ for the various qualities, with the result that SD quality is sufficient. We have made the shot detection application available online at `http://tomayac.com/youpr0n/` (accessed July 15, 2013) and invite the reader to compare the results, *e.g.*, the SD video `http://tomayac.com/youpr0n/videos/vsfashionshow_sd.mp4` (accessed July 15, 2013) with the HD version `http://tomayac.com/youpr0n/videos/vsfashionshow_hd.mp4` (accessed July 15, 2013).

Second, more advanced heuristics for the various user-definable options in the analysis process are possible. While there is no optimal configuration for all types of videos, there are some key indicators that can help categorize videos into classes and propose predefined known working settings based on the standard deviation $stdDev_f$ and the overall average frame distance $avgDist_f$. Both are dependent on the values of *boostingFactor*, *limitingFactor*, *rows*, and *columns*. Interpreting our results, there is evidence that low complexity settings are sufficient in most cases, *i.e.*, a number of *rows* and *columns* higher than *2* does not necessarily lead to more accurate shot boundary detection results. The same applies to the number of to-be-considered most different or similar tiles *tileLimit*. We had cases where not treating those tiles differently at all, *i.e.*, setting *boostingFactor = limitingFactor = 1*, led to better results; for example with screencast-type videos, typically used to demonstrate and teach the use of software features that were not recorded with a real camera, but directly recorded from the computer's screen, with "camera shots" then later added with video editing software.

7.6 Conclusions

In this chapter, we have introduced an algorithm for video shot boundary detection that was implemented as a stand-alone Web application and as a browser extension that adds shot boundary detection to YouTube videos. While the task of shot boundary detection is considered resolved for many cases, this is not true for the case of streaming online Web video. With this research, we have proposed and evaluated an approach that was shown to deliver consistently good results for all sorts of online videos. The biggest remaining challenge is finding *the* optimal algorithm settings for a given video. Promising directions for improving the shot boundary detection results are video categorization (fast-moving, slow-moving, color, black-and-white, *etc.*) prior to the actual shot detection process. By publicly sharing our implementation under a permissive open-source license, we open the door for future researchers to build upon our current results.

Chapter Notes

This chapter is partly based on the following publications.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, Michael Hausenblas, Raphaël Troncy, and Rik Van de Walle. *Enabling on-the-fly Video Shot Detection on YouTube.* Apr. 2012.

- Thomas Steiner, Ruben Verborgh, and Michael Hausenblas. “Crowdsourcing Event Detection in YouTube Videos”. In: *Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011), Workshop in conjunction with the 10th International Semantic Web Conference 2011 (ISWC 2011), Bonn, Germany, October 23, 2011*. Ed. by Marieke van Erp, Willem Robert van Hage, Laura Hollink, Anthony Jameson, and Raphaël Troncy. Vol. 779. CEUR Workshop Proceedings ISSN 1613-0073. Oct. 2011, pp. 58–67. URL: `http://ceur-ws.org/Vol-779/derive2011_submission_8.pdf`.

References

[1] M Ahmed, A Karmouch, and S Abu-Hakima. "Key Frame Extraction and Indexing for Multimedia Databases". In: *Proceedings of Visual Interface Conference.* 1999, pp. 506–511.

[2] Robin Berjon, Travis Leithead, Erika Doyle Navara, Edward O'Connor, and Silvia Pfeiffer. *HTML5: a vocabulary and associated APIs for HTML and XHTML.* Working Draft. W3C, 2012.

[3] P. Bouthemy, M. Gelgon, and F. Ganansia. "A Unified Approach To Shot Change Detection And Camera Motion Characterization". In: *IEEE Transactions on Circuits and Systems for Video Technology* 9 (1997), pp. 1030–1044.

[4] Ullas Gargi, Rangachar Kasturi, and Susan H. Strayer. "Performance Characterization of Video-Shot-Change Detection Methods". In: *IEEE Transactions on Circuits and Systems for Video Technology* 10.1 (2000), pp. 1–13.

[5] A. Hampapur, T. Weymouth, and R. Jain. "Digital Video Segmentation". In: *Proceedings of the Second ACM International Conference on Multimedia.* MULTIMEDIA '94. ACM, 1994, pp. 357–364.

[6] Alan Hanjalic. "Shot-Boundary Detection: Unraveled and Resolved?" In: *IEEE Transactions on Circuits and Systems for Video Technology* 12.2 (2002), pp. 90–105.

[7] Arnaud Le Hors, Philippe Le Hégaret, Lauren Wood, Gavin Nicol, Jonathan Robie, Mike Champion, et al. *Document Object Model (DOM) Level 3 Core Specification, Version 1.0.* Recommendation. W3C, 2004.

[8] R. Lienhart. "Comparison of automatic shot boundary detection algorithms". In: *Storage and Retrieval for Image and Video Databases.* Jan. 1999, pp. 290–301.

[9] Colin O'Toole, Alan Smeaton, Noel Murphy, and Sean Marlow. "Evaluation of Automatic Shot Boundary Detection on a Large Video Test Suite". In: *Proceedings of the 1999 International Conference on Challenge of Image Retrieval.* IM '99. British Computer Society, 1999, pp. 3–15.

[10] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, Michael Hausenblas, Raphaël Troncy, and Rik Van de Walle. *Enabling on-the-fly Video Shot Detection on YouTube.* Apr. 2012.

[11] Thomas Steiner, Ruben Verborgh, and Michael Hausenblas. "Crowdsourcing Event Detection in YouTube Videos". In: *Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011), Workshop in conjunction with the 10th International Semantic Web Conference 2011 (ISWC 2011), Bonn, Germany, October 23, 2011.* Ed. by Marieke van Erp, Willem Robert van Hage, Laura Hollink, Anthony Jameson, and Raphaël Troncy. Vol. 779. CEUR Workshop Proceedings ISSN 1613-0073. Oct. 2011, pp. 58–67. URL: `http://ceur-ws.org/Vol-779/derive2011_submission_8.pdf`.

[12] Jinhui Yuan, Huiyi Wang, Lan Xiao, Wujie Zheng, Jianmin Li, Fuzong Lin, et al. "A Formal Study of Shot Boundary Detection". In: *IEEE Transactions on Circuit and Systems for Video Technology* 17.2 (2007), pp. 168–186.

[13] Ramin Zabih, Justin Miller, and Kevin Mai. "A Feature-based Algorithm for Detecting and Classifying Scene Breaks". In: *Proceedings of the 3rd ACM International Conference on Multimedia.* ACM, 1995, pp. 189–200.

[14] Hongjiang Zhang, Atreyi Kankanhalli, and Stephen W. Smoliar. "Automatic Partitioning of Full-Motion Video". In: *Multimedia Systems* 1.1 (1993), pp. 10–28.

[15] Hongjiang Zhang, Chien Yong Low, and Stephen W. Smoliar. "Video Parsing and Browsing Using Compressed data". In: *Multimedia Tools and Applications* 1.1 (1995), pp. 89–111.

8

Media Item Deduplication

8.1 Introduction

In Chapter 6, we have motivated the need for media item deduplication. By clustering media items, we get a higher-level view on a media item cluster's overall performance on different networks. As detailed in section 3.1, media items can be photos or videos. WordNet [12, 23] defines the term *duplicate* as *"a copy that corresponds to an original exactly."* The corresponding verb *to duplicate* is defined as to *"make a duplicate or duplicates of."* The derived term *deduplication* in consequence refers to the act of eliminating duplicate or redundant information.

In this chapter, we will treat video and photo deduplication separately. Our goal is to deduplicate media items *on-the-fly* at the very moment they are extracted from social networks. Due to this limitation, we cannot rely on any preprocessing that state-of-the-art algorithms rely on. Our approaches to video and photo near-duplicate and exact-duplicate detection are founded on a tile-wise histogram-based pixel comparison algorithm that was partly introduced in the previous chapter.

8.1.1 Definitions

We have defined a social media item as either a photo (image) or video that was *publicly* shared or published on at least one social network. In the following, we will use the shorter term media item rather than the full term and define what we mean with *duplicate media items* for various cases.

Exact-duplicates for Photos: We define two media items of type photo as *exact-duplicates* if their pixel contents are exactly the same. This implies that by our definition a scaled or recompressed version of the same photo is *not* considered an exact-duplicate. Similarly, a rotated version of a photo is also *not* considered an exact-duplicate. In contrast, two photo files with different file names or different Exchangeable image file format[1] (Exif) data are considered exact-duplicate if their pixel contents are exactly the same. exact-duplicate photos typically occur if users share content from one social network on another, for example, if one user posts a photo on Instagram that then someone else (or even the same user) posts on Facebook.

Near-Duplicates for Photos: We define two media items of type photo as *near-duplicates* if their pixel contents differ no more than a given threshold after resampling. Examples of near-duplicate photos are scaled versions of the same photo, photos shot from a slightly different angle, rotated photos up to a certain degree, *etc.* Near-duplicate photos typically occur if event attendants stand close to each other and thus take photos from a similar standpoint. Another scenario is when a user applies a photo effect to a photo (like an Instagram filter) and in the following shares both the modified and the unmodified version.

Duplicates for Videos: We define two media items of type video as *exact-duplicates* if their pixel contents are frame by frame exactly the same. In practice, we lower this condition and instead of every frame only consider frames at shot boundaries. We make *no* requirements on the audio, *i.e.*, a video that has been dubbed in two different languages, but that fulfills the pixel contents equality condition, is considered exact-duplicate. Typical scenarios where exact-duplicate videos can occur is, for example, two users sharing the same YouTube video independently from each other.

Near-Duplicates for Videos: We define two media items of type video as *near-duplicates* if their pixel contents per frame differ no more than a given threshold. In practice, we lower this condition and instead of every frame only consider frames at shot boundaries. Typical scenarios where near-duplicate videos can occur is through logo or

[1] `http://www.cipa.jp/english/hyoujunka/kikaku/pdf/DC-008-2010_E.pdf`, accessed July 15, 2013

subtitle insertion, resizing, re-encoding, or aspect-ration changes. Note that we do not consider video subsegments near-duplicates, so a shortened version of an existing video is considered different, as a manual processing step was involved.

Special Case of Photo Contained in a Video: We define the special case of *a photo being contained in a video* if the pixel contents of a photo media item differ no more than a given threshold from the pixel contents of any of the frames of a video media item. In practice, we lower this condition and instead of every frame only consider frames at shot boundaries. Typically, this phenomenon occurs if two event attendants of the same event both cover it from almost the same standpoint, however, if the one attendant takes a video, while the other attendant takes a photo.

8.2 Related Work

Related work in the field of media item deduplication and clustering can be separated in different areas, which reflects the grouping of our definitions above. We further show related work on media fragments, digital storytelling, and Natural Language Generation, which we combine for a novel algorithm debugging approach.

Image Deduplication and Clustering: Work on ordinal measures that serve as a general tool for image matching was performed by Bhat *et al.* in [6]. Chum *et al.* have proposed a near-duplicate image detection method using MinHash and term frequency-inverse document frequency (tf-idf) weighting [10]. They use a visual vocabulary of vector quantized local feature descriptors based on Scale-Invariant Feature Transform (SIFT) [22]. Gao *et al.* [14] have proposed an image clustering method in the context of Web image clustering, which clusters images based on the consistent fusion of the information contained in both low-level features and surrounding texts. Also in the context of Web pages, Cai *et al.* [8] have proposed a hierarchical clustering method using visual, textual, and link analysis. Goldberger *et al.* [15] have combined discrete and continuous image models based on a mixture of Gaussian densities with a generalized version of the information bottleneck principle for unsupervised hierarchical image set clustering. Chen *et al.* [9] have introduced an image retrieval approach, which tackles the semantic gap problem by learning similarities of images of the same semantics.

Video Deduplication and Clustering: Specialized methods for video deduplication exist, for example [24, 43] by Min *et al.* who, given the observation that transformations tend to preserve the semantic information conveyed by the video content, propose an approach for identifying near-duplicate videos by making use of both low-level visual features and high-level semantic features detected using trained classifiers. In [26], Oliveira *et al.* report on four large-scale online surveys wherein they have confirmed that humans perceive videos as near-duplicates based on both non-semantic features like different image or audio quality, but also based on semantic features like different videos of similar content. A survey of video deduplication methods has been conducted by Lian *et al.* in [20]. In [16], Guil *et al.* have proposed a method for detecting copies of a query video in a videos database that groups frames with similar visual content while maintaining their temporal order. In [25], Okamoto *et al.* have proposed an approach that is based on fixed length video stream segments. By generating spatio-temporal images, they employ co-occurrence matrices to express features in the time dimension explicitly. Yi *et al.* have proposed motion histograms [45], where the motion content of a video at pixel level is represented as a Pixel Change Ratio Map (PCRM), which captures the motion intensity, spatial location, and size of moving objects in a video.

Image *and* Video Deduplication and Clustering: A method for both images *and* videos has been proposed by Yang *et al.* [44]. The authors describe a system for detecting duplicate images and videos in a large collection of multimedia data that uses local difference patterns as the unified feature to describe both images and videos. It has been demonstrated that the proposed method is robust against common image-processing tasks used to produce duplicates.

Media Fragments: There are many online video hosting platforms that have some sort of media fragments support. In the following, we present two representative ones. The video hosting platform YouTube[1] allows for deep-linking into videos via a proprietary URL parameter `t`, whose value has to match the regular expression `\d+m\d+s` (for minutes and seconds), as documented in [38]. Dailymotion[2] has similar URL parameters `start` and `end`, whose values have to match the regular expression `\d+` (for

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

[2] `http://www.dailymotion.com/`

seconds). The CSS Backgrounds and Borders Module Level 3 specification [7] defines the `background-size` property that can be used to crop media items visually and thus create the illusion of a spatial media fragment when combined with a wrapping element. Media Fragments URI [39] specifies a syntax for constructing media fragments URIs and explains how to handle them when used over the HTTP protocol [13]. The syntax is based on the specification of particular name-value pairs that can be used in URI query strings and URI fragment identifiers to restrict a media resource to a certain fragment. The temporal and spatial dimensions are currently supported in the basic version of Media Fragments URIs. Combinations of dimensions are also possible.

Digital Storytelling: Pizzi and Cavazza report in [28] on the development of an authoring technology on top of an interactive storytelling system that originated as a debugging[1] tool for a planning system. Alexander and Levine define in [2] the term *Web 2.0 storytelling*, where people create *microcontent*—small chunks of content, with each chunk conveying a primary idea—that gets combined with social media to form coherent stories. We use Media Fragments URIs to help human annotators understand the results of an algorithm by converting dry software debugging data to digital stories.

Natural Language Generation: Natural language generation is the NLP task of generating natural language from a machine representation system. This field is covered in great detail by Reiter and Dale in [31]. They divide the task into three stages: document planning, microplanning, and realization. *Document planning* determines the content and structure of a document. *Microplanning* decides which words, syntactic structures, *etc.* are used to communicate the chosen content and structure. *Realization* maps the abstract representations used by microplanning into text.

8.3 Photo Deduplication

We determine the popularity of media items shared across social networks. This task involves the deduplication of extracted media items. In Chapter 7, we have presented an algorithm for on-the-fly shot boundary detection for video media items. In this chapter, we will show how components of this algorithm can be used to deduplicate

[1] Pizzi and Cavazza use the term *debugging* in the non-IT sense: to check for redundancy, dead-ends, consistency, *etc.* in authored stories.

photos. Our work is situated in the broader context of summarizing events based on social network data. In order to get an overview of a given event based on a *potentially large* set of event-related media items, this set of media items needs to be *pruned* to exclusively contain highly relevant media items that are as representative for the event as possible. Rather than showing the viewer all media items, clusters of similar media items need to be formed. Within each cluster, the most representative media item has to be decided on according to well-defined criteria. Undesired exact-duplicate or near-duplicate content in the context of social networks arises in a number of situations that we will illustrate in the following.

8.3.1 Exact-Duplicate Content

Duplicate content in the context of social networks arises whenever people either share exactly the same, or an exact copy of a given media item. An example of the latter can be one user uploading the same media item to the two different social networks Google+ and Facebook. An example of the prior can be two users sharing the same YouTube video independently from each other, or re-sharing each other's content.

8.3.2 Near-Duplicate Content

Near-duplicate content in the context of social networks arises in a number of situations that we will illustrate in the following. All photos are real examples of media items shared on social networks that were clustered correctly as near-duplicates by our clustering algorithm, which we will detail in subsection 8.3.3.

Different Viewing Angle: When two people attend the same event and create media items at roughly the same time covering the same scene, their media items will be similar and—the capturing devices' quality aside—only differ in the viewing angles. Figure 8.1 shows a concrete example.

Logo, Watermark, Lower Third, or Caption Insertion: Oftentimes, organizations or individuals insert logos, watermarks, lower thirds, or captions into media items to highlight their origin, to convey related information, or to claim ownership of a media item. An example of caption, logo, and lower third insertion can be seen in Figure 8.2.

Cropping: Cropping refers to the removal of the outer parts of a media item to improve framing, accentuate subject matter, or to (lossily) change the aspect ratio. Cropping either happens manually via an image editing application or, more often, by the social networks themselves to obtain a square aspect ratio that better fits the timeline view of users, as can be seen in the example in Figure 8.3.

Different Keyframes: We have shown an approach to camera shot boundary detection in Chapter 7 and [34]. Different frames stemming from the same camera shot can occur on social networks when preview heuristics attempt to auto-select a representative poster frame from a video with different approaches, typically resulting in varying frames for different social networks. Figure 8.4 shows an example of this phenomenon.

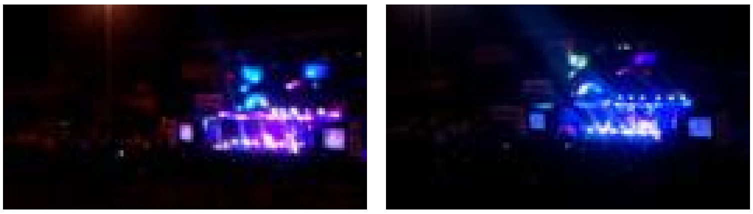
Aspect Ratio Changes with Squeezing or Stretching: Aspect ratio changes can either happen combined with cropping (and thus losing parts of the media item) and/or combined with squeezing or stretching (and thus deforming the media item). Figure 8.5 shows an example where a media item gets stretched.

Photo Filters: With the raising popularity of Instagram with its 90 million monthly active users,[1] photo filters that, *e.g.*, emulate retro Polaroid™ or tilt-shift effects are a considerable reason for near-duplicate media content on social networks. Figure 8.6 shows a typical example.

8.3.3 Near-Duplicate Photo Clustering Algorithm

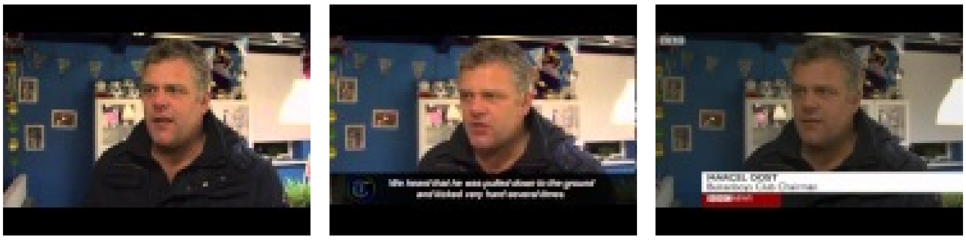
In the previous section, we have outlined reasons and sources for exact-duplicate and near-duplicate content. In this section, we describe an algorithm tailored to deduplicating and clustering exact-duplicate and near-duplicate media items. Design goals for the algorithm include the capability to detect exact-duplicate and near-duplicate media items in a timely, entirely *ad hoc* manner without any pre-calculation. In general—and especially for big events—event coverage on social networks is very broad, *i.e.*, there exist more media items than one could consume in a reasonable time. In consequence, it is tolerable for the algorithm to cluster media items aggressively rather than leaving too many media items unclustered. The algorithm has a twofold approach to clustering:

[1] `http://instagram.com/press/`, accessed July 15, 2013



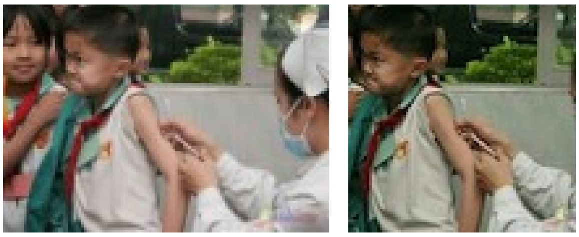
(a) Viewing angle 1 (b) Viewing angle 2

Figure 8.1: Slightly different viewing angles of a concert stage



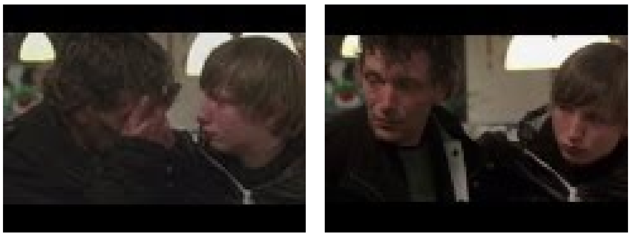
(a) Blank (b) Caption (c) Logo, lower third

Figure 8.2: Caption, logo, and lower third insertion for a speaker



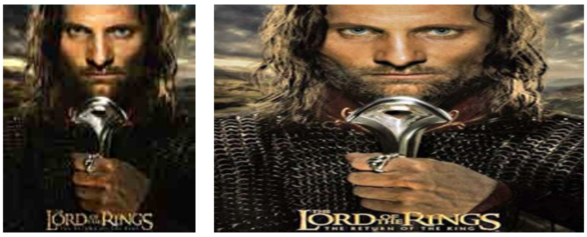
(a) Original (b) Cropped

Figure 8.3: Original and cropped version of a photo (including a slight color variation)



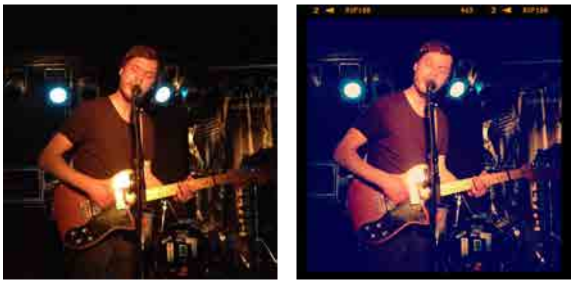
(a) Frame 1 (b) Frame 2

Figure 8.4: Two different frames stemming from the same camera shot, with the left frame appearing slightly earlier in the video



(a) Original **(b)** Stretched

Figure 8.5: Original and stretched version of a photo



(a) Original **(b)** With photo filter

Figure 8.6: Original and version with an applied photo filter of a photo

low-level analysis by looking at tile-wise pixel data, combined with *high-level* analysis by detecting faces in media items. In the following, we describe the face detection component of our media item clustering algorithm.

8.3.4 Face Detection

Face detection is a computer vision technology that determines the regions of faces in media items. Rotation-invariant face detection aims to detect faces with arbitrary rotation angles and is crucial as the first step in automatic face detection for general applications, as face images on social media are seldom upright and frontal. Face detection is a subclass of the broader class of object detection. The Viola-Jones object detection framework proposed in 2001 by Paul Viola and Michael Jones [40, 41] provides competitive object detection rates in realtime and was motivated primarily by the problem of face detection. We use an algorithm that further improves Viola-Jones, based on work by Huang *et al.* [17] and Abramson *et al.* [1] in a JavaScript implementation made available by Liu [21]. This algorithm runs in the context of a Web browser and, given the relatively small size of social network media items, is fast enough to be applied to hundreds of media items in well less than a second overall processing time on a standard laptop (mid-2010 MacBook Pro, 2.66 GHz Intel Core 2, 8 GB RAM).

8.3.5 Algorithm Description

Our near-duplicate media item clustering algorithm belongs to the family of tile-wise histogram-based clustering algorithms. As an additional semantic feature, the algorithm considers detected faces as described above. For two media items to be clustered, the following conditions have to be fulfilled.

1. Out of m tiles of a media item with n tiles ($m \leq n$), at most *tiles_threshold* tiles may differ not more than *similarity_threshold* from their counterpart tiles.
2. The numbers f_1 and f_2 of detected faces in both media items have to be the same. We note that we do not *recognize* faces, but only *detect* them.

```
Input: mediaItems, a list of media items
Output: clusters, a list of clustered media items


# Algorithm settings
ROWS = 10
COLS = 10
TILES_THRESHOLD = ceil(ROWS * COLS * 2/3)
SIMILARITY_THRESHOLD = 10


init:

# Calculates tile-wise histograms
histograms = {}
faces = {}
for item in mediaItems
  faces[item] = getFaces(item)

  histograms[item] = {}
  for tile in item
    histograms[item][tile] = getHistogram(tile)
  end for
end for
```

```
# Calculates tile-wise distances
distances = {}
for outerItem in mediaItems
  distances[outerItem] = {}
  for innerItem in mediaItems
    distances[outerItem][innerItem] = {}
    for tile in histograms[outerItem]
      distances[outerItem][innerItem][tile] =
          abs(histograms[outerItem][tile] -
              histograms[innerItem][tile])
    end for
  end for
end for


# Calculates clusters
clusters = {}
for outerItem in mediaItems
  clusters[outerItem] = []
  for innerItem in mediaItems
    if outerItem == innerItem then continue
    similarTiles = 0
    distance = distances[outerItem][innerItem]
    for tile in distance
      if distance[tile] <= SIMILARITY_THRESHOLD then
        similarTiles++
      end if
    end for
    # Check condition 1 (tiles)
    if similarTiles >= TILES_THRESHOLD then
      # Check condition 2 (faces)
      if faces[outerItem] == faces[innerItem] then
        clusters[outerItem].push(innerItem)
      end if
    end if
  end for
end for

return clusters
```

Listing 8.1: Simplified pseudocode of the exact- and near-duplicate media item deduplication and clustering algorithm

The algorithm pseudocode can be seen in Listing 8.1. In the actual implementation some speed improvements, for example, looking up already calculated distances[1] have been applied; these were omitted in the listing for legibility reasons. We calculate the histograms and distances only once initially. The clusters are then recalculated dynami-

[1] `distances[outerItem][innerItem] = distances[innerItem][outerItem]`

cally whenever either *tiles_ threshold* or *similarity_ threshold* change. The given values of $rows = cols = 10$ and $tiles\_threshold = 67 = \lceil rows \cdot cols \cdot 2/3 \rceil$ and $similarity\_threshold = 15$ were determined empirically on a large corpus of event-related media items and are known to deliver solid results. The corpus has been made available publicly, see subsection 8.3.7 for the details.

8.3.6 Experiments

We have evaluated the near-duplicate photo clustering on two events from (at time of writing) recent history with high social network coverage that we will briefly describe in the following.

Grammy Awards Nominations 2013: The Grammy Award—or short Grammy—is an award by the National Academy of Recording Arts and Sciences of the United States to recognize outstanding achievement in the music industry. The annual ceremony features performances by prominent artists. Some of the awards are presented in a widely viewed televised ceremony. On December 5, 2012, the nominees for the 55th Annual Grammy Awards were announced at an event broadcasted live by the broadcast network CBS titled *Grammy Nominations Concert Live*,[1] during which Taylor Swift and LL Cool J revealed the nominees in the so-called Big Four categories Album, Record and Song of the Year, and Best New Artist. CBS suggested the hashtag `#GRAMMYNoms`.

Victoria's Secret Fashion Show 2012: The *Victoria's Secret Fashion Show*[2] is an annual event sponsored by Victoria's Secret, a brand of lingerie and sleepwear. The show features some of the world's leading fashion models and is used by the brand to promote and market its goods in high-profile settings. The show is a lavish event with elaborate costumed lingerie and varying music by leading entertainers that attracts hundreds of celebrities and entertainers, with special performers and acts every year. The 2012 edition of the show, which was previously taped on November 7, 2012 was aired on December 4, 2012 on CBS to an audience of 9.48 million viewers. CBS suggested the hashtag `#VSFashionShow` for the event.

[1] `http://en.wikipedia.org/wiki/2013_Grammy_Awards`, accessed July 15, 2013

[2] `http://en.wikipedia.org/wiki/Victoria's_Secret_Fashion_Show`, accessed July 15, 2013

8.3.7 Evaluation

We have collected and made available[1] datasets for both events with 379 photos for the *Victoria's Secret Fashion Show 2012* event and 949 photos for the *Grammy Awards Nominations 2013* event. These photos were collected using the media item extraction framework described in Chapter 6 using a mix of hashtag searches with the official event hashtags combined with full-text searches for event titles and variations thereof [4, 5]. Due to the short-lived nature of social networks, the returned results of the media item extraction process itself are not reproducible. Additionally, our focus in this chapter is on media item deduplication and clustering, not extraction. The concrete clustering parameters for the algorithm were set as listed below.

1. $rows = cols = 10$

2. $tiles\_threshold = 67$

3. $similarity\_threshold = 15$

In the following, we discuss the clustering and deduplication results. Figure 8.7 and Figure 8.8 show the top clusters for the *Victoria's Secret Fashion Show 2012* and the *Grammy Awards Nominations 2013* events respectively. We then pick some representative examples from both events and have a closer look at the clustering algorithm's strengths and weaknesses.

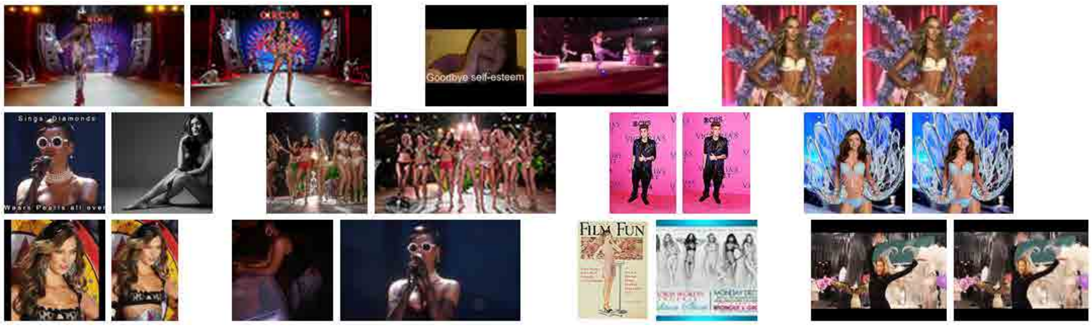


Figure 8.7: Top clusters for the *Victoria's Secret Fashion Show 2012* event

[1] https://www.dropbox.com/sh/2llvjaut32juwrx/7eGLodfP_2, accessed July 15, 2013

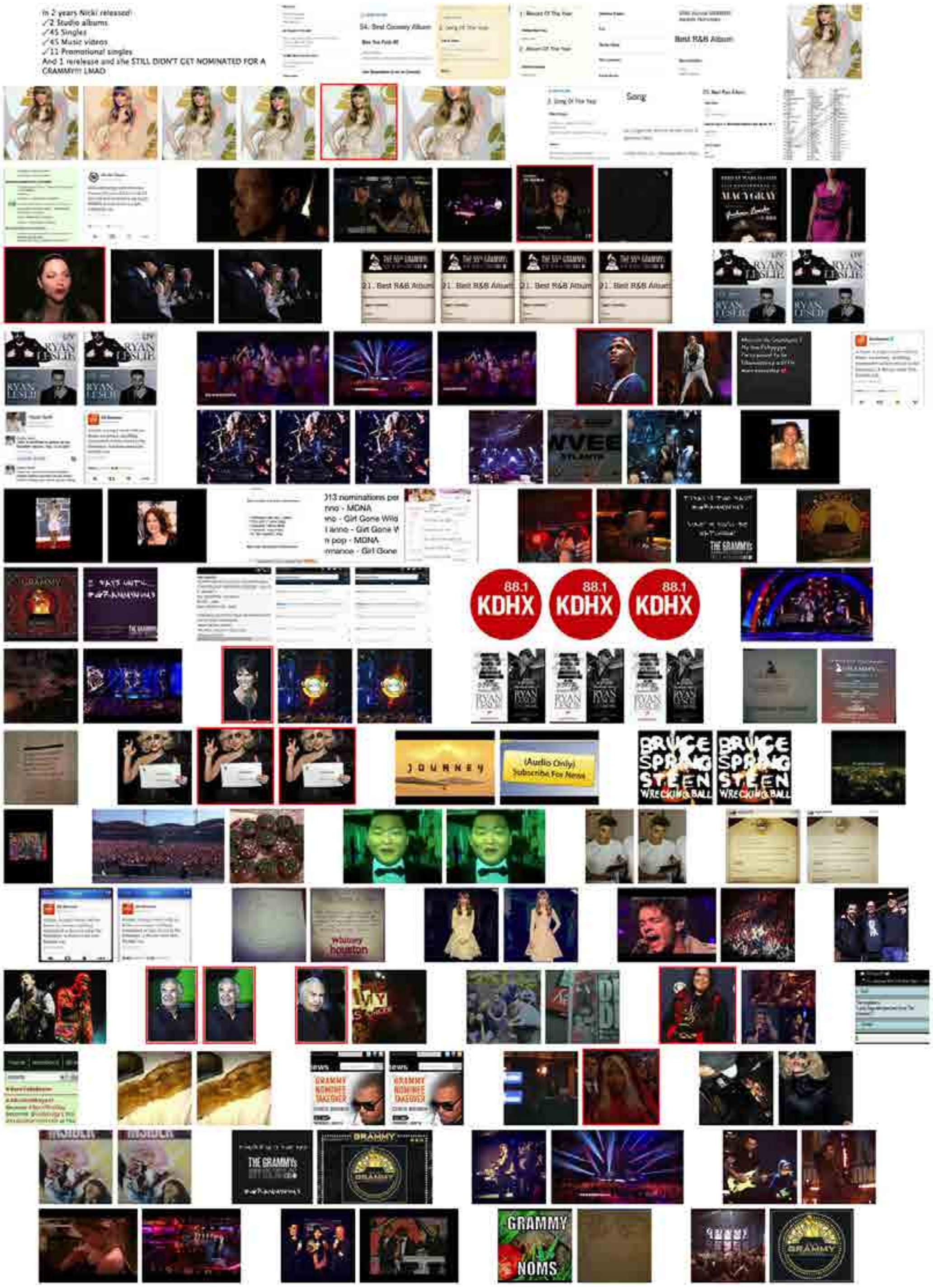


Figure 8.8: Top clusters for the *Grammy Awards Nominations 2013* event

Algorithm Strengths: Figure 8.9 has a brunette fashion model in a pink robe and a heart-shaped pink spotlight as central elements of both photos. Even though the model is shot from different angles and at different times, the photos are successfully clustered due to the very identifying colors and the high tile-wise similarity.

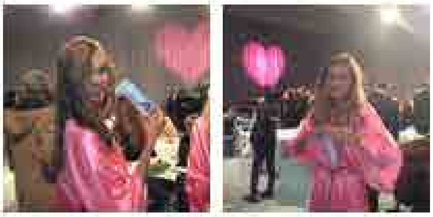


Figure 8.9: High tile-wise similarity of a dominating color

Figure 8.10 shows two views of a stage taken at slightly different times. The left photo covers a detail of the scene, whereas the right photo covers the entire stage. Due to the tile-wise similarity of the scene detail, the photos are successfully clustered.

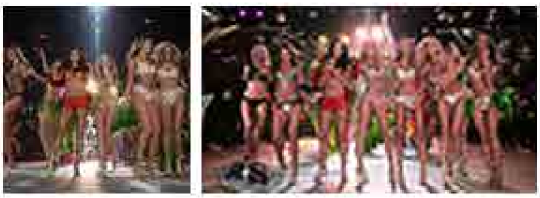


Figure 8.10: Cropped view of a stage scene

Figure 8.11 shows two views of a stage under different lighting conditions. Due to the tile color tolerances and the tile-wise similarity, the photos are successfully clustered.

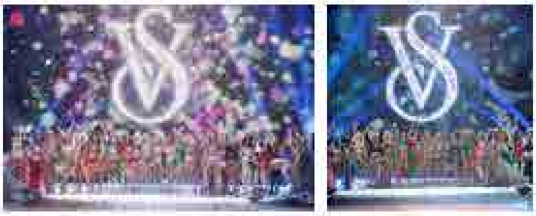


Figure 8.11: Stage and detail of a stage under different lighting conditions

Figure 8.12 shows two photos of the same fashion model, where the left photo is a zoomed version of the right photo with added black bars so that the resulting photo has a square aspect ratio. Despite the differences, the photos are successfully clustered.

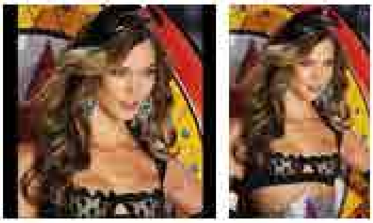


Figure 8.12: Zoomed view of a model with black bars left and right

Figure 8.13 shows two views of the same stage, however, with a different person. Due to the dominating tile-wise similarity of the stage tiles, the photos are clustered.

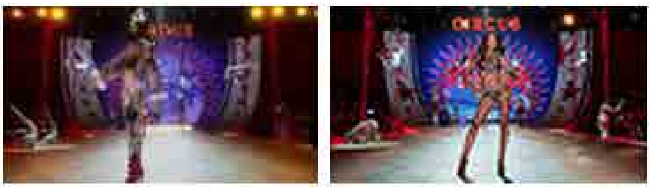
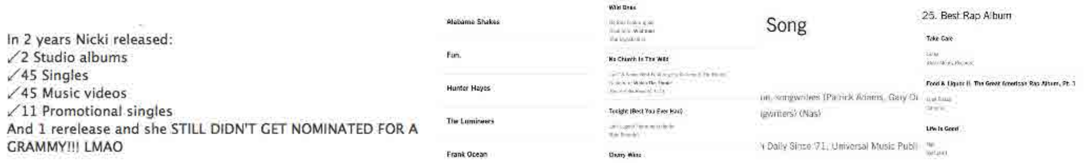


Figure 8.13: Two views of the same stage with different person

Algorithm Weaknesses: Figure 8.14 shows five media items with pure white as the dominating color and a pure black font stemming from screenshots of the Grammy results from Web pages. The algorithm in its previously described form clusters such media items. This may or may not be desired.



In 2 years Nicki released:
✓2 Studio albums
✓45 Singles
✓45 Music videos
✓11 Promotional singles
And 1 rerelease and she STILL DIDN'T GET NOMINATED FOR A GRAMMY!!! LMAO

Alabama Shakes
Fun.
Hunter Hayes
The Lumineers
Frank Ocean

Wild Ones
No Church In The Wild
Tonight (Best You Ever Had)
Diamonds

Song
songwriters (Patrick Adams, Gary Dr
songwriters) (Nas)
Daily Since '71, Universal Music Publ

25. Best Rap Album
Take Care
Food & Liquor II: The Great American Rap Album, Pt. 1
Life Is Good

Figure 8.14: Pure white as dominating color stemming from screenshots (bad quality caused by down-scaling via the originating social networks)

Likewise, at the other end of the color spectrum, Figure 8.15 shows two media items of a woman with pure black as the dominating color, one time with and the other time without added black bars to fit a letterbox aspect ratio. In its previously described form, the algorithm does *not* cluster such media items (unless a very small number *tiles_ threshold* of required similar tiles is selected). In the majority of cases, though, clustering such media items *is* desired. Our response to both issues is to ignore a certain part of the color spectrum in the algorithm's similarity measure. In the concrete case, ignoring pure white and pure black correctly fixed the clustering in our chosen example events in all but one cases, without negatively impacting previously formed clusters.

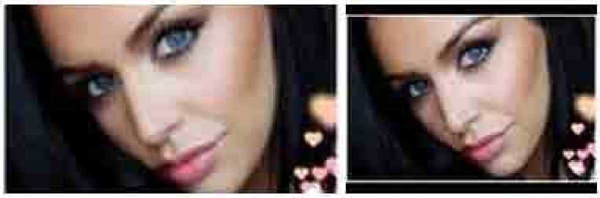


Figure 8.15: Black bars added to fit a 16:9 image in a 4:3 letterbox (the white border is part of the original photo)

Finally, Figure 8.16 shows two entirely different media items that were incorrectly clustered as the tile histograms were similar enough under the chosen similarity threshold. The explanation for this is twofold. First, the original source media items were very small thumbnail-like images, which hindered face recognition (there is actually an *unequal* number of faces in each image). Second, the way the algorithm works causes the tiles of very tiny media items like the ones in question to blur.

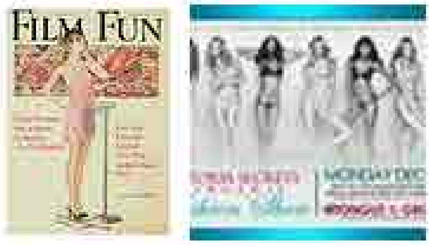


Figure 8.16: Entirely different photos with similar tile histograms

We have experienced in our experiments that there is no single perfect combination of algorithm parameters, so the only way to address this issue (besides ignoring too small media items, which in practice might be the easiest and best solution) is to make the parameters flexible. In our graphical user interface, we have created sliders that let the user interactively preview clustering changes. As noted before, a screenshot of the application is available online at the URL `http://twitpic.com/c02qfs/full` (accessed July 15, 2013).

8.3.8 Algorithm Performance Analysis

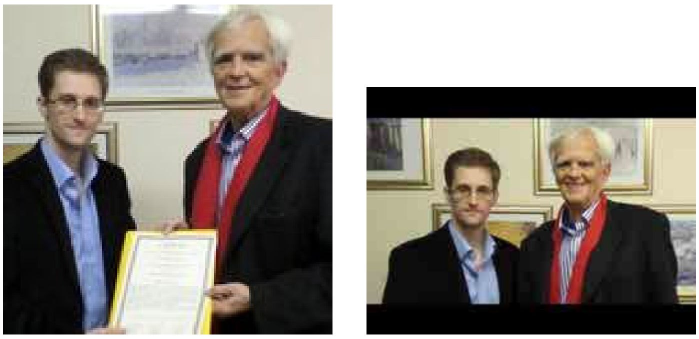
As our application is meant to be used interactively on the Web, a high, real-time performance of the clustering algorithm is crucial. We have thoroughly evaluated its performance on our public dataset of 379 media items for the *Victoria's Secret Fashion Show 2012* event and 949 media items for the *Grammy Awards Nominations 2013* event.

Processing Speed Considerations

The results for the combined diverse set of 1,328 media items are as follows.

- **Face Detection:** the task of detecting faces took on average 325 ms per media item.
- **Histogram Calculation:** the task of calculating 100 tile histograms (10 rows · 10 columns = 100 tiles) took on average 7 ms per media item.
- **Distance Calculation:** the task of calculating the distances from each media item to all others took on average 2 ms per media item.

In consequence, the overall average processing time per media item was roughly 1/3 of a second, resulting in less than 8 minutes processing time for all 1,328 media items. We have compared our algorithm that is based on the *low-level* feature of tile histograms combined with the *high-level* feature of face detection to the three state-of-the-art feature detection algorithms SIFT [22], ASIFT [46], and SURF [3]. A concrete example of all four algorithms applied to the same two near-duplicate media items from Figure 8.17, which are common, highly representative social media items of type photo and video from recent history, can be seen in Figure 8.18. We especially highlight the differences in runtime. Compared to SIFT, our algorithm runs about 15 times faster, compared to Affine-SIFT, about 23 times faster, and finally compared to SURF, still about 3 times faster. We also note that our algorithm is implemented as interpreted JavaScript in the context of a Web browser using the `canvas` element, whereas SIFT, Affine-SIFT, and SURF are implemented as compiled, native C and C++ applications.

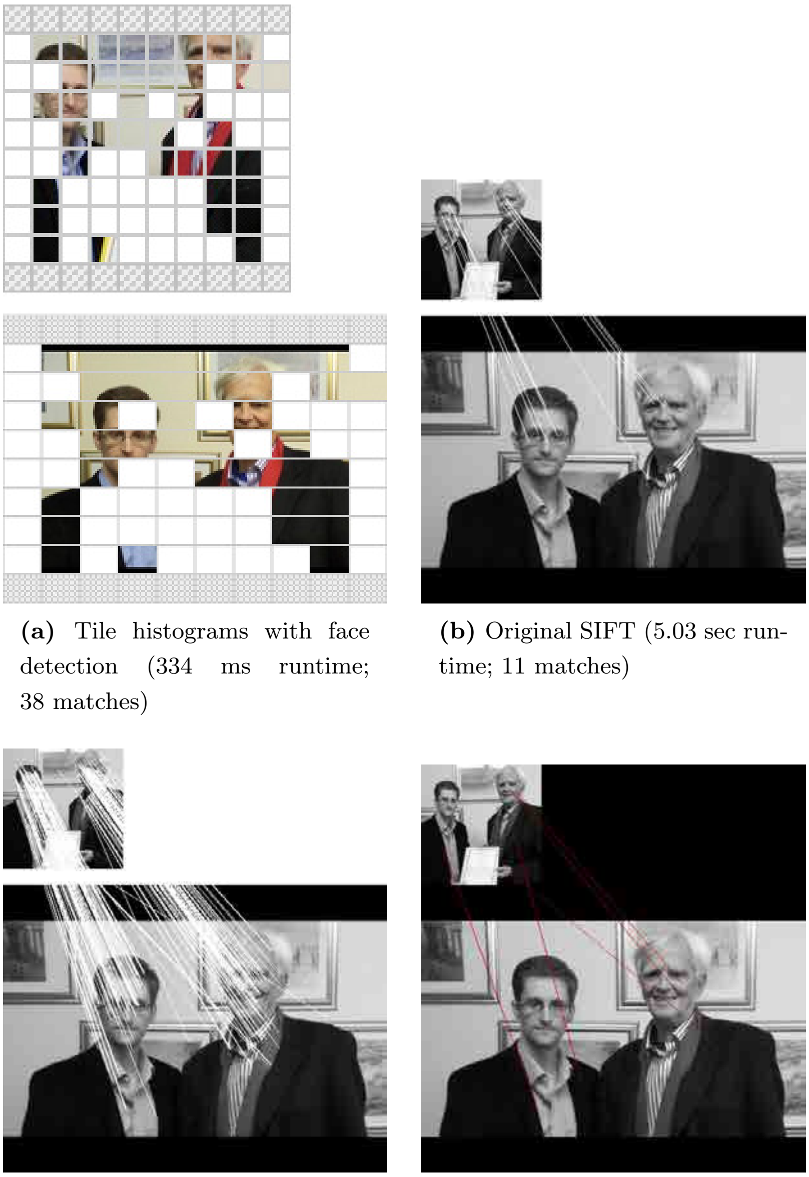


(a) Photo (b) Video keyframe

Figure 8.17: Two near-duplicate media items (photo and video)

Algorithm Accuracy Considerations

The focus of SIFT and ASIFT is on *local* image features, where the feature descriptor is invariant to uniform scaling, orientation, and partially invariant to affine distortion and illumination changes. SURF [3], which stands for Speeded Up Robust Features, is a speed-optimized high-performance scale- and rotation-invariant interest point detector and descriptor. Considering the number of matching features in Figure 8.18 with 38 matching features for our approach against 5 for SURF, 11 for SIFT, and 200 for ASIFT, our algorithm performs well compared to the more advanced feature detection algorithms, especially considering the fast execution time. This also applies in the general case with other media items and is mainly due to the observed reasons for duplicate and near-duplicate content on social networks (subsection 8.3.2), where orientation-invariance does not play a central role. Given our concrete context of social networks, the trade-off of lost accuracy regarding orientation-invariance against faster performance is justified by the enormous processing speed gains. The algorithm still maintains scale-invariance to the necessary extent, which can be seen in Figure 8.12. SIFT, ASIFT, and SURF operate on a black-and-white representation of the media item in question, whereas our algorithm, apart from face detection, works with color histograms. By taking color features into account, our algorithm is invariant to illumination changes, which is well visible in Figure 8.11. At the same time, maximum visual diversity is assured, which is as an important aesthetic feature of media galleries [37].



(a) Tile histograms with face detection (334 ms runtime; 38 matches)

(b) Original SIFT (5.03 sec runtime; 11 matches)

(c) Affine-SIFT (7.83 sec runtime; 200 matches)

(d) SURF (1.21 sec runtime; 5 matches)

Figure 8.18: Our approach compared to the state-of-the art feature detection algorithms SIFT, Affine-SIFT, and SURF

8.4 Video Deduplication

In the previous section, we have introduced an algorithm for photo deduplication. In the upcoming section, we will outline the conceptual framework of how this algorithm can be combined with the previously introduced video shot boundary detection algorithm from Chapter 7. This will allows us to on the one hand directly deduplicate videos on a shot boundary frame basis or on the other hand to detect whether a given photo is contained in a video.

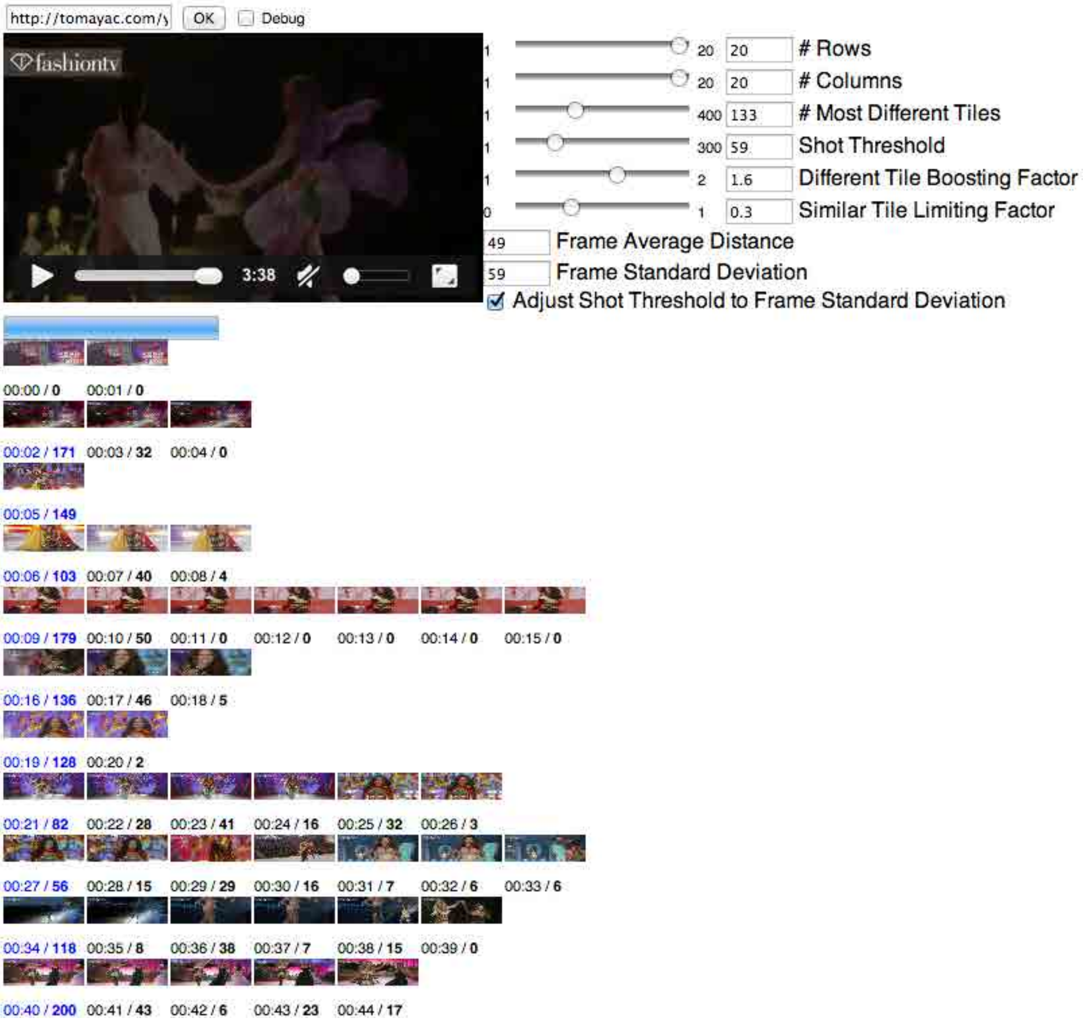
8.4.1 Photo-contained-in-Video Workflow

In a first step, for a given video, we detect shot boundaries as described before in section 7.1. To illustrate this, Figure 8.19 shows an excerpt of detected shot boundaries for a video related to the *Victoria's Secret Fashion Show 2012* event. The first photo of each shot boundary film stripe is selected as the particular shot's representative photo. To detect whether a given photo stemming from social networks is contained in the video in question, the set of extracted shot representative photos is compared with all social network photos, some of which are shown in Figure 8.7. We note, however, that especially for longer videos (about 4 minutes and longer) this approach does not scale due to the sheer number of camera shots in common videos shared on social networks, which causes the process to consume too much time in practice. At the expense of exactness, (the few) poster still frames that are typically returned by video hosting platform APIs can be used rather than extracting (all) shot boundaries manually.

8.4.2 Video-contained-in-Video Workflow

To detect whether a given video is contained in another, we follow a similar approach as outlined in the previous subsection, with the sole difference being that we need to compare all detected shot boundary representative photos of the source video with the ones from the other. Naturally, this approach is even less scalable with regard to system response time. The practicable work-around is, as before, to limit oneself to poster frames delivered by the video hosting platforms. Our experiments have shown that this approach works very well for common social network user behavior. For example, the 3:19 minutes long video of Mark Zuckerberg explaining the design and engineering challenges behind Facebook's recently announced Graph Search product was initially published on Facebook,[1] however, people republished the same video multiple times on YouTube. As the YouTube-generated poster frames were similar enough and even

[1] `https://www.facebook.com/about/graphsearch`, accessed July 15, 2013



http://tomayac.com/y OK □ Debug

1 20 20 # Rows
1 20 20 # Columns
1 400 133 # Most Different Tiles
1 300 59 Shot Threshold
1 2 1.6 Different Tile Boosting Factor
0 1 0.3 Similar Tile Limiting Factor
49 Frame Average Distance
59 Frame Standard Deviation
☑ Adjust Shot Threshold to Frame Standard Deviation

```
00:00 / 0   00:01 / 0
00:02 / 171 00:03 / 32 00:04 / 0
00:05 / 149
00:06 / 103 00:07 / 40 00:08 / 4
00:09 / 179 00:10 / 50 00:11 / 0  00:12 / 0  00:13 / 0 00:14 / 0 00:15 / 0
00:16 / 136 00:17 / 46 00:18 / 5
00:19 / 128 00:20 / 2
00:21 / 82  00:22 / 28 00:23 / 41 00:24 / 16 00:25 / 32 00:26 / 3
00:27 / 56  00:28 / 15 00:29 / 29 00:30 / 16 00:31 / 7 00:32 / 6 00:33 / 6
00:34 / 118 00:35 / 8  00:36 / 38 00:37 / 7  00:38 / 15 00:39 / 0
00:40 / 200 00:41 / 43 00:42 / 6  00:43 / 23 00:44 / 17
```

Figure 8.19: Excerpt of detected shot boundaries in a *Victoria's Secret Fashion Show 2012* event video

if the other video metadata like title and description were different, we were able to effectively deduplicate the videos with the described work-around approach. We note that this approach works reasonably well as online videos are still relatively short, also given that YouTube per default has an (increasable) limit of 15 minutes per video.[1]

[1] `https://support.google.com/youtube/answer/71673?hl=en`, accessed July 15, 2013

8.5 Describing Media Item Differences with Media Fragments URI and Speech Synthesis

In this section, we describe how media item differences can be described with media fragments URIs and speech synthesis. We will combine the two techniques for the purpose of introducing a novel algorithm debugging approach, illustrated with our previously described media item clustering algorithm.

8.5.1 Algorithm Debug View

In order to illustrate the way the algorithm clusters media items, Figure 8.20 shows a debug view of the algorithm for two clustered media items related to the *Grammy Awards Nominations 2013* event. The red border around the media item indicates at least one detected face. Independent from the actual media item's aspect ratio, the tile-wise comparison always happens based on a potentially squeezed square aspect ratio version. The two slightly different media items (caption insertion, lighting change) were clustered, because out of the $10 \cdot 10 = 100$ tiles, 85 of the minimum required *tiles_threshold* of 67 tiles differed not more than the *similarity_threshold* of 15 per tile. In both media items, exactly 1 face was detected. A screenshot of the complete media item clustering application (with a different event) is available online at `http://twitpic.com/c02qfs/full` (accessed July 15, 2013).

8.5.2 Media Fragments Requirements

A media fragment is a part that was separated from its parent media item. In order to make statements about such media fragments, we need to uniquely identify them. In the context of our research on media item deduplication and clustering, media fragments identifiers need to be capable of expressing the following concepts.

1. Given a rectangular media item with the dimensions $width \times height$, express that in turn rectangular tiles of smaller dimensions are part of the original media item.
2. Given detected faces at the granularity level of bounding rectangles, express that these bounding rectangles are within the dimensions of the original media item and that each bounding rectangle contains a face.

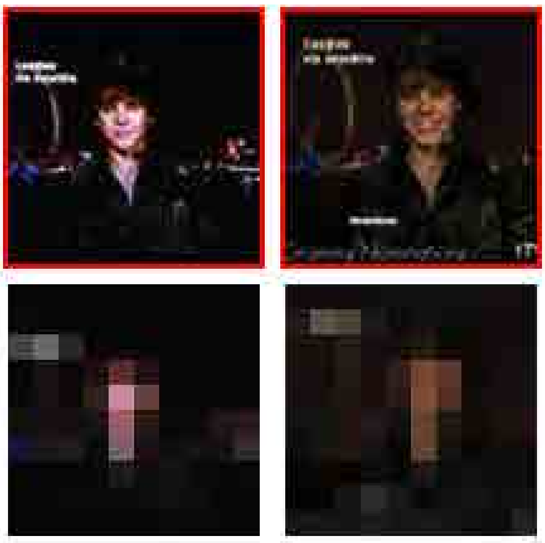


Figure 8.20: Algorithm debug view for two clustered media items related to the *Grammy Awards Nominations 2013* event (the red border around the media items indicates at least one detected face)

3. Requirements *i* and *ii* need to be fulfilled for both types of media items, *i.e.*, photos and videos. In case of the latter, video subsegments of any length—including video still frames—need to be supported.

Media Fragments URI [39] as described in the basic version of the specification supports all three requirements. The *temporal dimension* is denoted by the parameter name `t` and specified as an interval with a begin time and an end time. Either one or both parameters may be omitted, with the begin time defaulting to 0 seconds and the end time defaulting to the duration of the source media item. The interval is half-open: the begin time is considered part of the interval, whereas the end time is considered to be the first time point that is not part of the interval. If only a single value is present, it corresponds to the begin time, except for when it is preceded by a comma, which indicates the end time. The temporal dimension is specified in the Normal Play Time (NPT, [32]) format.

The *spatial dimension* selects an area of pixels from media items. In the current version of the specification, only rectangular selections are supported. Rectangles can be specified as pixel coordinates or percentages. Rectangle selection is denoted by the parameter name `xywh`. The value is either `pixel:` or `percent:` (defaulting to pixel:) and four comma-separated integers. The integers denote x, y, *width*, and *height* respectively, with $x = 0$ and $y = 0$ being the top left corner of the media item. If percent: is used, x and *width* are interpreted as a percentage of the width of the original media item, while y and *height* are interpreted as a percentage of the original height. While (at time of writing) the temporal dimension is implemented natively in common Web browsers, this is not the case for the spatial dimension.

The intent of the Ontology for Media Resources [19] by Lee *et al.* is to bridge different description methods of media resources and to provide a core set of descriptive properties. Combined with Media Fragments URI, this allows for making statements about media items and fragments thereof. An example in RDF Turtle syntax [30] is given in Listing 8.2.

```
@base <http://example.org/> .
@prefix ma: <http://www.w3.org/ns/ma-ont> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix db: <http://dbpedia.org/resource/> .
@prefix dbo: <http://dbpedia.org/ontology/> .
@prefix col: <http://purl.org/colors/rgb/> .

<video> a ma:MediaResource .
<video#t=,10&xywh=0,0,30,40> a ma:MediaFragment ;
                             foaf:depicts db:Face .
<video#t=,10&xywh=0,0,10,10> a ma:MediaFragment ;
                             dbo:colour col:f00 .
```

Listing 8.2: Description of two 10 sec long media fragments: *(i)* a tile of dimensions 30×40 pixels starting at pixel coordinates $(0, 0)$ that contains a face; and *(ii)* a tile of dimensions 10×10 pixels starting at pixel coordinates $(0, 0)$ of red color

8.5.3 Algorithm Debug Properties

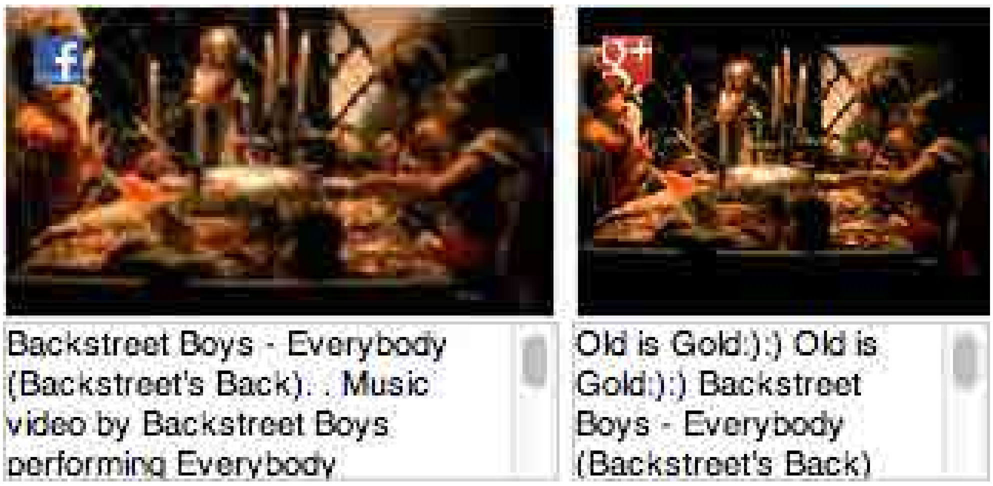
The deduplication algorithm described in this chapter belongs to the family of tile-wise histogram-based clustering algorithms. As an additional semantic feature, the algorithm considers detected faces. It is capable of deduplicating media items of type video and/or photo. In the case of video, frames at camera shot boundaries are used. To illustrate the algorithm debugging mechanics, we use a running example of two media items related to a music video by the band Backstreet Boys, which can be seen in Figure 8.21. For media items to be clustered, the following clustering conditions have to be fulfilled.

Cond. 1 Out of m tiles of a media item with n tiles ($m \leq n$), the average color of at most *tiles_ threshold* tiles may differ not more than *similarity_ threshold* from their counterpart tiles.

Cond. 2 The numbers f_1 and f_2 of detected faces in both media items have to be the same. We note that the algorithm does not *recognize* faces, but only *detects* them.

Cond. 3 If the average colors of a tile and its counterpart tile are within the black-and-white tolerance *bw_ tolerance*, these tiles are not considered and *tiles_ threshold* is decreased accordingly (we will talk about *effective_ tiles_ threshold* in section 8.5.3).

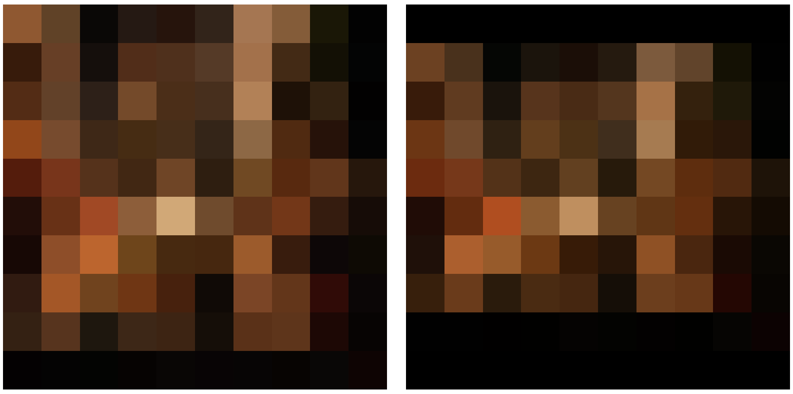
The black-and-white tolerance *bw_ tolerance* avoids media items to be clustered when the particular tiles are too dark (*e.g.*, for the video borders in Figure 8.21) or too bright (*e.g.*, for screenshots of Web pages or applications, which frequently appear on social networks). In order to illustrate the way the algorithm deduplicates media items, Figure 8.22 shows a debug view of the algorithm for the two clustered media items related to the previous example around the *Backstreet Boys* music video. Independent of the actual media items' aspect ratios, the tile-wise comparison always happens based on a potentially squeezed square aspect ratio version.



Backstreet Boys - Everybody
(Backstreet's Back). . Music
video by Backstreet Boys
performing Everybody

Old is Gold:):) Old is
Gold:):) Backstreet
Boys - Everybody
(Backstreet's Back)

Figure 8.21: *Near-duplicate* music video *Everybody* by the *Backstreet Boys* shared independently on Facebook and Google+



(a) From Facebook user

(b) From Google+ user

Figure 8.22: Debug view of the media item deduplication algorithm: since no faces are detected in Figure 8.21, the clustering is based on tile similarity; pure black tiles are not considered due to the chosen black-and-white tolerance)

Debugging the Algorithm

In this subsection, we consider the following three debug scenarios that occurred most frequently during our previous experiments with human raters. They correspond to situations where, given a set of deduplicated and clustered media items, a human annotator wanted to understand the specific details leading to the decisions taken by the algorithm that they were unsure about or had decided on differently.

Clustering Consent. Two or more media items are clustered by the algorithm and the human rater also agrees. The human rater wants to understand why they were clustered.

Clustering Dissent. Two or more media items are clustered by the algorithm, but the human rater thinks that they should not have been clustered. The human rater wants to understand why they were incorrectly clustered.

Non-Clustering Dissent. Two or more media items are not clustered by the algorithm, but the human rater thinks that they should have been clustered. The human rater wants to understand why they were not clustered.

In order to provide answers to these human raters' information needs, different levels of the algorithm's internals have to be debugged. Is the *tiles_threshold* (*i.e.*, the number of tiles that may differ) too high or too low? Complementary to this, is the *similarity_ threshold* (*i.e.*, the maximum amount two tiles may differ) too high or too low (**Cond. 1**)? Are the number of detected faces f_1 and f_2 the same? Are all faces correctly detected, or should the face matching condition be temporarily disregarded, *e.g.*, with too tiny media items, where faces fail to be detected (**Cond. 2**)? If the media items to be compared have very dark and/or very bright parts, is the *bw_ tolerance* too high or too low (**Cond. 3**)?

Low-Level Debug Output

As a consequence of the previous observations, the low-level debug output must include the currently selected *tiles_ threshold* and *similarity_ threshold* and how many tiles with the present algorithm settings currently fulfill **Cond. 1**. In addition to that, the debug output has to contain the number of detected faces f_1 and f_2 in each media item, *i.e.*, whether **Cond. 2** is fulfilled, as well as the number of not considered tiles (according to *bw_ tolerance*), which implies fulfillment of **Cond. 3** and potentially impacts **Cond. 1** in form of the *effective_ tiles_ threshold*. For instance, consider the low-level debug output for the media items from the running example of the *Backstreet Boys* media items.

```
- Similarity threshold: 15 (Cond. 1)
- Tiles threshold: 67 (Cond. 1)
- Similar tiles: 52 (Cond. 1)
- Faces left: 0. Faces right: 0 (Cond. 2)
- BW tolerance: 1 (Cond. 3)
- Not considered tiles: 22 (Cond. 3)
- Effective tiles threshold: 45 (Cond. 3)
```

8.5.4 From Debug Output to Story

While this low-level debug output is sufficient to respond to the polar question (yes/no question) whether media items are clustered at all or not, it does not help with the non-polar *why* question (the linguistic term for this type of questions is *wh–question*). In order for human raters to get answers to the question on *why* media items are clustered, we need to lift the low-level debug output to a high-level natural language story for the previously defined debug scenarios *Clustering Consent*, *Clustering Dissent*, and *Non-Clustering Dissent*. This results in a natural language generation task, whose three stages according to Reiter's and Dale's architecture [31] will be detailed below.

Generating Natural Language

Document Planning: In our context, the document is a set of low-level debug data as illustrated in section 8.5.3. The natural language generation task is thus manageable. We need to convey the currently selected *tiles_ threshold* and *similarity_ threshold*, the number of detected faces f_1 and f_2 in each media item, and the number of tiles not considered given the *bw_ tolerance* parameter.

Microplanning: The microplanning task is driven by the debug scenarios that were described previously. Initially, we need to decide on a matching condition aspect of the algorithm that will be first highlighted. Typically, this will be the overall tiles statistics. Afterwards, we need to elaborate on secondary matching conditions such as detected faces and black-and-white tolerance. The grammatical number (plural or singular) needs to be taken into account when statements about tile(s) or face(s) are planned. Some values, *e.g.*, the percentage of matching tiles, are calculated. The microplanner needs to decide when exactness (*e.g.*, *"99% of all tiles"*) and when approximation of calculated values (*e.g.*, *"roughly 50%"*) better suits the human evaluators' information needs. Neutral non-judgmental statements (*e.g.*, *"45 tiles"*) and biased judgmental statements (*e.g.*, *"not a single one [tile]"*) need to be carefully balanced. Finally, in the interest of a more naturally sounding phrase composition, the microplanner needs to be aware of contrasting juxtaposition (*e.g.*, *"Both the left and the right media item contain one detected face." vs. "The left media item contains no detected faces, while the right media item contains one detected face."*).

Realization: We show examples of sentences that are actually generated for the three different debug scenarios (Quotes 1–3). For the sake of completeness, we provide one additional example (Quote 4) for the debug scenario **Non-Clustering Consent**. The running example of the *Backstreet Boys* media items for the music video *Everybody* is represented by Quote 1.

Clustering Consent (Quote 1). *"The two media items are near-duplicates. Out of overall 100 tiles, 52 from the minimum required 45 tiles were similar enough to be clustered. However, 22 tiles were not considered, as they are either too bright or too dark, which is a common source of clustering issues. Neither the left, nor the right media item contain detected faces."*

Clustering Dissent (Quote 2). *"The two media items are near-duplicates. Out of overall 100 tiles, 41 from the minimum required 41 tiles were similar enough to be clustered. However, 26 tiles were not considered, as they are either too bright or too dark, which is a common source of clustering issues. Neither the left, nor the right media item contain detected faces."*

Non-Clustering Dissent (Quote 3). *"The two media items are different. Out of overall 100 tiles, only 8 from the minimum required 67 tiles were similar enough to be clustered. This corresponds to 8 percent of all tiles. The left media item contains 2 detected faces, while the right media item contains 1 detected face."*

(Non-Clustering Consent) (Quote 4). *"The two media items are different. Out of overall 100 tiles, not a single one was similar enough to be clustered. Neither the left, nor the right media item contain detected faces."*

Technical Implementation

Text-to-Speech: The generated texts are converted to speech using a text-to-speech system. We use the eSpeak [11] speech synthesizer that was originally developed by Jonathan Duddington in a JavaScript port called Speak.js, made available by Alon Zakai [47]. This speech synthesizer uses the formant synthesis method, which allows for many languages to be provided in a small size. Rather than using human speech samples at runtime, the synthesized speech output is created using additive synthesis and an acoustic model, where parameters such as fundamental frequency, voicing, and noise levels are varied over time to create a waveform of artificial speech. The speech is clear and can be used at high speeds. However, it is not as natural or smooth as larger synthesizers that are based on speech recordings.

Visual Media Fragments Highlighting: We treat and address each tile of a media item as a spatial media fragment. Figure 8.23 shows a grid of similar, different, and not considered tiles from the *Backstreet Boys* media items for the *Everybody* music video. While the speech synthesizer reads the generated text, the corresponding tiles (*e.g.*, the matching tiles or the due to the black-and-white tolerance not considered tiles) are visually highlighted to support the human evaluators' understanding, as can be seen in Figure 8.24 and in a screencast available at `http://youtu.be/DWqwEnhqTSc` (accessed July 15, 2013). Spatial Media Fragments URIs are currently not implemented in any common Web browser [42]. In order to nonetheless support spatial media fragments, we use a so-called JavaScript polyfill for Media Fragments URI that was developed in the context of this thesis.[1] In Web development, a polyfill is downloadable code that provides facilities by emulating potential future features or APIs that are not built-in to a Web browser [33]. Our polyfill—in contrast to an additional earlier spatial Media Fragments URI polyfill implementation [42] by Fabrice Weinberg—supports more browsers and both image *and* video.

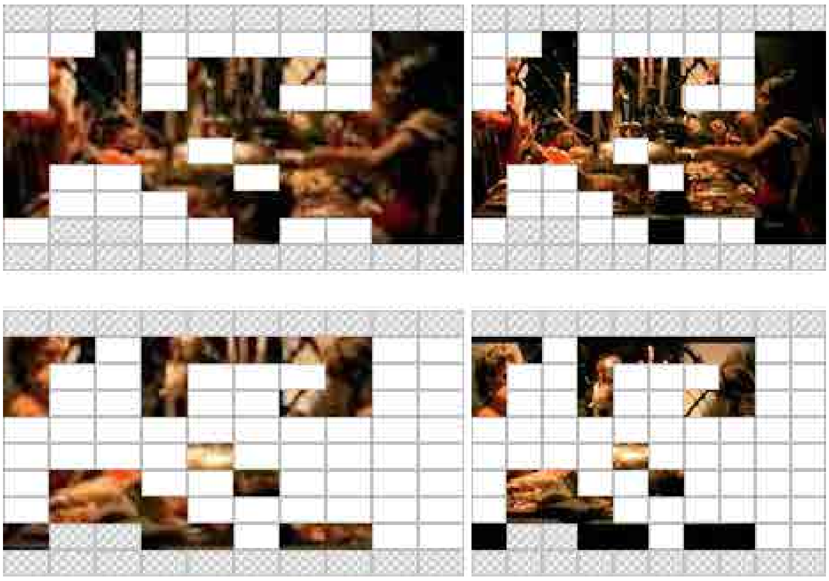


Figure 8.23: Similar (upper row) and different (lower row) corresponding tile pairs for the media items from a Facebook (left column) and a Google+ user (right column); checkered tiles are not considered due to the black-and-white tolerance

[1] `https://github.com/tomayac/xywh.js` accessed July 15, 2013

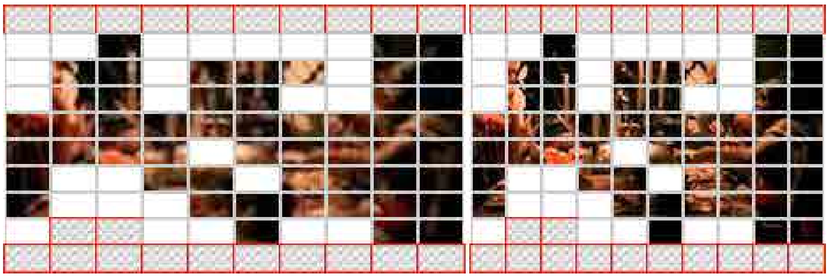


Figure 8.24: Due to the black-and-white tolerance not considered checkered tiles, as the text-to-speech system explains: *"However, 22 tiles were not considered, as they are either too bright or too dark, which is a common source of clustering issues."*

8.5.5 Evaluation

Evaluating Natural Language Generation Systems: For the evaluation of natural language generating systems, there are three basic techniques. First, the *task-based* or *extrinsic* evaluation, where the generated text is given to a person who evaluates how well it helps with performing a given task [29]. Second, there are *automatic metrics* such as BLEU [27], where the generated text is compared to texts written by people based on the same input data. Finally, there are *human ratings*, where the generated text is given to a person who is asked to rate the quality and usefulness of the text. For our evaluation, we have chosen the third approach of human ratings, as we do not evaluate the natural language generating system in isolation, but in *combination with a visual representation* that makes use of spatial Media Fragments URIs (Figure 8.23 and Figure 8.24).

Evaluating Subjective Data: A common subjective evaluation technique is the *Mean Opinion Score* (MOS, [18]). Traditionally, MOS is used for conducting subjective evaluations of telephony network transmission quality, however, more recently, MOS has also found wider usage in the multimedia community for evaluating inherently subjective things like perceived quality from the users' perspective. Therefore, a set of standard subjective tests are conducted, where a number of users rate the quality of test samples with scores ranging from 1 (worst) to 5 (best). The actual MOS is then the arithmetic mean of all individual scores.

Evaluation Results: In the context of this research, we have conducted MOS test sessions with five external human raters. We generated artificially modified deduplicated media item sets around media items about the *Backstreet Boys* that were shared on social networks during the time of writing. These media item sets were curated by yet another independent two external persons, assisted by a previously developed software system that implements the deduplication algorithm described in this chapter. We asked the two persons to provoke dissent and consent clustering situations for the five human raters, *i.e.*, obviously correct clustering (**Clustering Consent**), obviously incorrect clustering (**Clustering Dissent**), and obviously incorrect non-clustering (**Non-Clustering Dissent**). We then asked the five human raters to have the system automatically explain the algorithm results to them as described in subsection 8.5.4. The raters gave MOS scores ranging from 2 to 5, with the overall average values as follows: **Clustering Consent**: 4.3, **Clustering Dissent**: 3.3, and **Non-Clustering Dissent**: 4.1. The human raters appreciated the parallel explanation approach, where the visual and the audial parts synchronously described what the algorithm was doing. They uttered that the not considered tiles (due to the black-and-white tolerance) as well as erroneously not detected faces were sources of error in the algorithm that they easily understood thanks to the human language description. They sometimes wished for more diversification in the generated texts. Without exception, they liked the system and encouraged future development.

8.6 Conclusions

In this chapter, we have treated the topic of media item deduplication from different angles. We have first defined the meaning of *exact* and *near-duplicate* for both photos and videos, including the special case of a photo being contained in a video. In a previous chapter, we have introduced an algorithm and application for video shot boundary detection whose foundations then served for a more general photo deduplication algorithm with semantic features in the present chapter. We have evaluated the algorithm for two recent events that had broad social media coverage. Further, we have outlined how the photo deduplication algorithm can be used for basic video deduplication, albeit minimum system response time requirements hinder its full applicability in practice. Finally, we have successfully demonstrated the feasibility of making the task of debugging a complex algorithm more human-friendly by means of a combined visual and audial approach. We have used Media Fragments URI together with a natural language generation framework realized through a speech synthesizer to visually and audially describe media item differences. The approach was successfully evaluated for its helpfulness and utility with the evaluation method Mean Opinion Score (MOS). Our contribution also includes a polyfill implementation of spatial Media Fragments URIs.

Media item deduplication of both exact- and near-duplicate media items is a fundamental step in dealing with huge amounts of social media and media overload in general. Highly popular media items not only tend to retrieve many social interactions on the social network they were initially shared on, but also on other social networks. Derivates of popular media items further add noise to the social media sharing landscape. Based on our media item deduplication algorithms, we have contributed effective and efficient tools to deal with social media overload and to identify the few needles in the cross network haystack.

Chapter Notes

This chapter is partly based on the following publications.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Near-duplicate Photo Deduplication in Event Media Shared on Social Networks". In: *Proceedings of the International Conference on Advanced IT, Engineering and Management*. Feb. 2013, pp. 187–188.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, Erik Mannens, and Rik Van de Walle. "Clustering Media Items Stemming from Multiple Social Networks". In: *The Computer Journal* (2013). DOI: `10.1093/comjnl/bxt147`. eprint: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.full.pdf+html`. URL: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.abstract`.

References

[1] Yotam Abramson, Bruno Steux, and Hicham Ghorayeb. “Yet Even Faster (YEF) Real-time Object Detection”. In: *International Journal of Intelligent Systems Technologies and Applications* 2.2/3 (Feb. 2007), pp. 102–112.

[2] B. Alexander and A. Levine. “Web 2.0 Storytelling: Emergence of a New Genre”. In: *EDUCAUSE Review* 43.6 (2008), pp. 40–56.

[3] Herbert Bay, Tinne Tuytelaars, and Luc Gool. “SURF: Speeded Up Robust Features”. In: *Computer Vision – ECCV 2006*. Vol. 3951. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006, pp. 404–417. ISBN: 978-3-540-33832-1. DOI: 10.1007/11744023_32. URL: http://dx.doi.org/10.1007/11744023_32.

[4] Hila Becker, Dan Iter, Mor Naaman, and Luis Gravano. “Identifying Content for Planned Events Across Social Media Sites”. In: *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. WSDM ’12. ACM, 2012, pp. 533–542.

[5] Hila Becker, Mor Naaman, and Luis Gravano. “Learning Similarity Metrics for Event Identification in Social Media”. In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. WSDM ’10. ACM, 2010, pp. 291–300.

[6] Dinkar N. Bhat and Shree K. Nayar. “Ordinal Measures for Image Correspondence”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.4 (1998), pp. 415–423.

[7] B. Bos, E. J. Etemad, and B. Kemper. *CSS Backgrounds and Borders Module Level 3*. Candidate Recommendation. http://www.w3.org/TR/css3-background/, accessed July 15, 2013. W3C, 2012.

[8] Deng Cai, Xiaofei He, Zhiwei Li, Wei-Ying Ma, and Ji-Rong Wen. “Hierarchical clustering of WWW image search results using visual, textual and link information”. In: *Proceedings of the 12th Annual ACM International Conference on Multimedia*. MULTIMEDIA ’04. New York, NY, USA: ACM, 2004, pp. 952–959. ISBN: 1-58113-893-8.

[9] Yixin Chen, James Z. Wang, and Robert Krovetz. “Content-based image retrieval by clustering”. In: *Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval*. MIR ’03. Berkeley, California: ACM, 2003, pp. 193–200. ISBN: 1-58113-778-8.

[10] Ondrej Chum, James Philbin, and Andrew Zisserman. “Near Duplicate Image Detection: min-Hash and tf-idf Weighting”. In: *Proceedings of the British Machine Vision Conference 2008*. British Machine Vision Association, 2008.

[11] J. Duddington. *eSpeak Text to Speech*. `http://espeak.sourceforge.net/`, accessed July 15, 2013. 2012.

[12] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Language, Speech and Communication Series. Cambridge, MA: MIT Press, 1998.

[13] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, et al. *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616. IETF, 1999.

[14] Bin Gao, Tie-Yan Liu, Tao Qin, Xin Zheng, Qian-Sheng Cheng, and Wei-Ying Ma. “Web image clustering by consistent utilization of visual features and surrounding texts”. In: *Proceedings of the 13th Annual ACM International Conference on Multimedia*. MULTIMEDIA ’05. Hilton, Singapore: ACM, 2005, pp. 112–121. ISBN: 1-59593-044-2.

[15] J. Goldberger, S. Gordon, and H. Greenspan. “Unsupervised image-set clustering using an information theoretic framework”. In: *Image Processing, IEEE Transactions on* 15.2 (Feb. 2006), pp. 449–458.

[16] N. Guil, J.M. González-Linares, J.R. Cózar, and E.L. Zapata. “A Clustering Technique for Video Copy Detection”. In: *Pattern Recognition and Image Analysis*. Ed. by Joan Martí, José Miguel Benedí, Ana Maria Mendonça, and Joan Serrat. Vol. 4477. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, pp. 451–458. ISBN: 978-3-540-72846-7.

[17] Chang Huang, Haizhou Ai, Yuan Li, and Shihong Lao. “High-Performance Rotation Invariant Multiview Face Detection”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 29.4 (Apr. 2007), pp. 671–686.

[18] International Telecommunication Union, Telecommunication Standardization Sector. *ITU-T Recommendation P.800: Methods for Subjective Determination of Transmission Quality*. `http://www.itu.int/rec/T-REC-P.800-199608-I/en`, accessed July 15, 2013. Aug. 1998.

[19] W. Lee, W. Bailer, T. Bürger, P.-A. Champin, J.-P. Evain, V. Malaisé, et al. *Ontology for Media Resources 1.0*. Recommendation. `http://www.w3.org/TR/mediaont-10/`, accessed July 15, 2013. W3C, 2012.

[20] Shiguo Lian, Nikolaos Nikolaidis, and HusrevTaha Sencar. “Content-Based Video Copy Detection — a Survey”. In: *Intelligent Multimedia Analysis for Security Applications*. Vol. 282. Studies in Computational Intelligence. Springer, 2010, pp. 253–273.

[21] Liu Liu. *JavaScript Face Detection Explained*. `http://liuliu.me/eyes/javascript-face-detection-explained/` accessed July 15, 2013. 2012.

[22] David G. Lowe. “Object Recognition from Local Scale-Invariant Features”. In: *Proceedings of the International Conference on Computer Vision*. Vol. 2. ICCV ’99. IEEE Computer Society, 1999, pp. 1150–1157.

[23] George A. Miller. “WordNet: a Lexical Database for English”. In: *Communications of the ACM* 38.11 (1995), pp. 39–41.

[24] Hyun-Seok Min, Jae Young Choi, Wesley De Neve, and Yong Man Ro. “Bimodal Fusion of Low-level Visual Features and High-level Semantic Features for Near-duplicate Video Clip Detection”. In: *Signal Processing: Image Communication* 26.10 (2011), pp. 612–627.

[25] H. Okamoto, Y. Yasugi, N. Babaguchi, and T. Kitahashi. “Video clustering using spatio-temporal image with fixed length”. In: *Multimedia and Expo, 2002. ICME ’02. Proceedings. 2002 IEEE International Conference on*. Vol. 1. 2002, pp. 53–56.

[26] Rodrigo De Oliveira, Mauro Cherubini, and Nuria Oliver. “Looking at Near-Duplicate Videos from a Human-Centric perspective”. In: *ACM Transactions on Multimedia Computing, Communications, and Applications* 6.3 (2010), pp. 1–22.

[27] K. Papineni, S. Roukos, T. Ward, and W.J. Zhu. “BLEU: A Method for Automatic Evaluation of Machine Translation”. In: *40th Annual Meeting on Association for Computational Linguistics (ACL’02)*. Philadelphia, Pennsylvania, 2002, pp. 311–318.

[28] D. Pizzi and M. Cavazza. “From Debugging to Authoring: Adapting Productivity Tools to Narrative Content Description”. In: *1st Joint International Conference on Interactive Digital Storytelling (ICIDS’08)*. Erfurt, Germany, 2008, pp. 285–296.

[29] F. Portet, E. Reiter, A. Gatt, J. Hunter, S. Sripada, Y. Freer, et al. “Automatic Generation of Textual Summaries from Neonatal Intensive Care Data”. In: *Artificial Intelligence* 173.7–8 (2009), pp. 789–816.

[30] E. Prud’hommeaux, G. Carothers, D. Beckett, and T. Berners-Lee. *Turtle – Terse RDF Triple Language*. Candidate Recommendation. `http://www.w3.org/TR/turtle/`, accessed July 15, 2013. W3C, 2013.

[31] E. Reiter and R. Dale. *Building Natural Language Generation Systems*. Studies in Natural Language Processing. Cambridge University Press, 2000.

[32] H. Schulzrinne, A. Rao, and R. Lanphier. *Real Time Streaming Protocol RTSP*. RFC 2326. `http://www.ietf.org/rfc/rfc2326.txt`, accessed July 15, 2013. IETF, 1998.

[33] R. Sharp. *What is a Polyfill?* `http://remysharp.com/2010/10/08/what\discretionary{-}{}{}-is-a-polyfill/`, accessed July 15, 2013. 2010.

[34] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, Michael Hausenblas, Raphaël Troncy, and Rik Van de Walle. *Enabling on-the-fly Video Shot Detection on YouTube*. Apr. 2012.

[35] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Near-duplicate Photo Deduplication in Event Media Shared on Social Networks". In: *Proceedings of the International Conference on Advanced IT, Engineering and Management*. Feb. 2013, pp. 187–188.

[36] Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, Erik Mannens, and Rik Van de Walle. "Clustering Media Items Stemming from Multiple Social Networks". In: *The Computer Journal* (2013). DOI: `10.1093/comjnl/bxt147`. eprint: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.full.pdf+html`. URL: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.abstract`.

[37] Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, and Rik Van de Walle. "Defining Aesthetic Principles for Automatic Media Gallery Layout for Visual and Audial Event Summarization based on Social Networks". In: *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*. July 2012, pp. 27–28. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/defining-aesthetic-principles-for-automatic-media-gallery-layout-qomex2012.pdf`.

[38] The YouTube Team. *Link To The Best Parts In Your Videos*. `http://youtube-global.blogspot.com/2008/10/link-to-best-parts-in-your-videos.html`, accessed July 15, 2013. 2008.

[39] R. Troncy, E. Mannens, S. Pfeiffer, D. Van Deursen, M. Hausenblas, P. Jägenstedt, et al. *Media Fragments URI 1.0 (basic)*. Recommendation. `http://www.w3.org/TR/media-frags/`, accessed July 15, 2013. W3C, 2012.

[40] P. Viola and M.J. Jones. “Rapid Object Detection Using a Boosted Cascade of Simple Features”. In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on.* Vol. 1. 2001, pp. 511–518.

[41] Paul Viola and Michael J. Jones. “Robust Real-Time Face Detection”. In: *International Journal of Computer Vision* 57.2 (2004), pp. 137–154.

[42] F. Weinberg. *Media Fragments URI—Spatial Dimension.* `http://css-tricks.com/\discretionary{-}{}{}media-fragments-uri-spatial-dimension`, accessed July 15, 2013. 2013.

[43] Xiao Wu, Chong-Wah Ngo, Alexander G. Hauptmann, and Hung-Khoon Tan. “Real-time Near-Duplicate Elimination for Web Video Search with Content and Context”. In: *IEEE Transactions on Multimedia* 11.2 (2009), pp. 196–207.

[44] Xin Yang, Qiang Zhu, and Kwang-Ting Cheng. “Near-duplicate Detection for Images and Videos”. In: *1^{st} ACM Workshop on Large-Scale Multimedia Retrieval and Mining.* LS–MMRM ’09. 2009, pp. 73–80.

[45] Haoran Yi, Deepu Rajan, and Liang-Tien Chia. “A new motion histogram to index motion content in video segments”. In: *Pattern Recognition Letters* 26.9 (2005), pp. 1221–1231. ISSN: 0167-8655.

[46] Guoshen Yu and Jean-Michel Morel. “ASIFT: An Algorithm for Fully Affine Invariant Comparison”. In: *Image Processing On Line* 2011 (2011). Electronic access: `http://dx.doi.org/10.5201/ipol.2011.my-asift`. DOI: `10.5201/ipol.2011.my-asift`.

[47] Alon Zakai. *Speak.js.* `https://github.com/kripken/speak.js`, accessed July 15, 2013. 2012.

9

Media Item Ranking

9.1 Introduction

In the previous chapter, we have introduced methods to deduplicate exact-duplicate and near-duplicate media items. In this chapter, we introduce ranking criteria and methods to put deduplicated media clusters in a well-defined order. The application screenshots that can be seen in Figure 8.7 and Figure 8.8 show the most intuitive ranking criterion one can imagine (besides publication time): ranking by occurrence popularity. The more often a media item (or a near-duplicate) appears in any of the considered social networks, the higher it should be ranked. However, ranking by occurrence popularity (or media item cluster size) disregards one of the most valuable features of social networks: the social aspects. In consequence, in this chapter, we will introduce further media item ranking criteria that, together with media item cluster size, allows us to propose more adequate social media item ranking mechanisms.

9.2 Media Item Ranking Criteria

In this section, we describe several criteria that can serve to rank media items retrieved from social networks. We base these criteria on the information available via the media item extractors described in Chapter 6 and the deduplication and clustering algorithm described in Chapter 8. Given event-related search terms, via these approaches, we extract raw binary media items and associated textual microposts and detected named entities as described in Chapter 4 from multiple social networks.

Textual Ranking Criteria: This category regards the microposts that accompany media items. Typically, microposts provide a description of media items. Using named entity disambiguation tools, textual content can be linked to *LOD* cloud concepts [10]. We have described micropost annotation in detail in Chapter 4.

Visual Ranking Criteria: This category regards the contents of photos and videos. We distinguish *low-* and *high-level* visual ranking criteria. High-level criteria include logo detection, face recognition, and camera shot separation. Low-level criteria include file size, resolution, duration of a video, geolocation, time, and more. Via *Optical Character Recognition (OCR)*, contained texts can be treated as textual feature.

Audial Ranking Criteria: This category regards the audio track of videos. *High-level* ranking criteria are the presence or absence of silence, music, speech, or a mixture thereof in videos. Similar to visual features before, audial *low-level* features are the average bit rate, volume, possibly distorted areas, *etc.* Through audio transcription, speech can be treated as textual feature.

Social Ranking Criteria: This category regards social network effects like shares, mentions, view counts, expressions of (dis)likes, user diversity, *etc.* in a network-agnostic way across *multiple* social networks. We will detail social aspects later in this chapter.

Aesthetic Ranking Criteria: This category regards the desired outcome after the ranking, *i.e.*, the media gallery that illustrates a given event and its atmosphere. Studies exist for the aesthetics of automatic photo book layout [7], photo aesthetics *per se* [6], and video and music playlist generation [1, 3]. However, to the best of our knowledge, no media gallery composition aesthetics studies exist that examine mixing video *and* photo media items.

Temporal Ranking Criteria: This category regards the publication date of media items. If media items are clustered, we can use the youngest media item as cluster representative. Media items can be ranked by recency, as oftentimes more recent items are more interesting in the streaming context of social networks.

9.3 Social Interactions Abstraction Layer

As we have described in Chapter 3, social networks have different paradigms of social interactions. In subsection 6.4.1, we have briefly presented the overall abstraction layer on top of the native data formats of all considered social networks in order to gain a network-agnostic view on the underlying social networks. In this section, we detail the part of the abstraction layer that models the network-specific social interaction patterns. Those interaction patterns ideally are exposed by the social network via specific API calls in order to be considered, which only is the case for a subset of the social networks we deal with. Social interaction data is to some extent the holy grail of social networks, which is the reason why sometimes Web scraping is the last resort when not all data is accessible via APIs, as we have outlined in more detail in subsection 6.4.2. In Table 9.1, we have listed how we abstract the social interactions in question on each social network. In our concrete implementation, we differ unknown values that are returned as `unknown`, *i.e.*, where the information is not exposed, from `0` values, where the value is known to be zero. We briefly recall the social interactions part of the abstraction layer's data format:

`socialInteractions` Container for social interactions

- `likes` Number of times a micropost was liked, or `unknown`
- `shares` Number of times a micropost was shared, or `unknown`
- `comments` Number of comments a micropost received, or `unknown`
- `views` Number of views a micropost reached, or `unknown`

9.4 Merging and Ranking

If a set of media items is sufficiently similar to be clustered under the criteria that were detailed in Chapter 8, we can treat the whole of the cluster as if it were just one media item. Therefore, we need to specify a merging strategy for the associated data of the individual media items in the particular cluster. Listing 9.1 shows the pseudocode of the merging algorithm. During the merging step, we treat unknown values that are represented as `unknown` as `0`. The alternative to this solution would be to exclude

Likes	Shares	Comments	Views
Facebook Like	Facebook Share	Facebook Comments	YouTube Views
Google+ +1	Google+ Share	Google+ Comments	Flickr Views
Instagram Like	Twitter ReTweet	Instagram Comments	Twitpic Views
Flickr Favorite		Twitter manual RT	MobyPicture Views
YouTube Like		Twitter @Replies	
YouTube Favorite		Twitpic Comments	
Twitter Favorite		MobyPicture Comments	
		Flickr Comments	

Table 9.1: Abstract social network interaction paradigms and their underlying native social network counterparts

unknown values from the merging step. However, as in practice a considerable amount of social interaction values are unknwon, we are forced to proceed with the abovementioned simplification. The algorithm accumulates individual social interactions and assigns the accumulated values to the cluster.

```
Input: cluster, cluster of visually similar media items
Output: cluster, cluster with merged social interactions

for mediaItem in cluster
  cluster.likes += isUnknown(mediaItem.likes) ? 0 : mediaItem.likes
  cluster.shares += isUnknown(mediaItem.shares) ? 0 : mediaItem.shares
  cluster.comments += isUnknown(mediaItem.comments) ? 0 : mediaItem.comments
  cluster.views += isUnknown(mediaItem.views) ? 0 : mediaItem.views
end for

return cluster
```

Listing 9.1: Social interactions merging algorithm

9.4.1 Selection of a Cluster's Visual Representative

As outlined in the previous section, similar enough media items are clustered and treated as just one media item by applying the merging algorithm for the social interactions data. Now, we introduce an algorithm for the selection of a cluster's visual representative. Naturally, through the way the clustering algorithm works, the contained media

```
Input: cluster, cluster of visually similar media items
Output: cluster, cluster with visual representative

maxResolution = 0

for mediaItem in cluster

  # Ensure that videos will always be preferred
  if mediaItem.type == 'video' then
    mediaItem.width = mediaItem.height = INFINITY
  end if

  resolution = mediaItem.width * mediaItem.height
  if resolution >= maxResolution then
    maxResolution = resolution
    cluster.representative = mediaItem
  end if
end for

return cluster
```

Listing 9.2: Pseudocode of the cluster visual representative selection algorithm that finds the highest quality media item of a cluster

items are already visually similar based on high-level features. In consequence, we fall back to using *low-level* visual ranking criteria as defined in section 9.2. Listing 9.2 shows the cluster representative selection algorithm, which is based on the low-level feature photo or video *resolution*. The algorithm selects the media item with the highest megapixel resolution as the cluster representative, which is a solid heuristic for the optimal photo or video quality.

9.4.2 Ranking Formula

Up to now, we have shown how media item clusters are formed, how each cluster's social interactions data is accumulated, and how a cluster's representative media item is selected. In this section, we describe a ranking formula to rank a set of media clusters that match a given query. In the ranking formula, we consider several well-defined ranking criteria that were detailed in [11], namely visual, audial, textual, temporal, social, and aesthetic. For a given set of media item clusters, a ranking is calculated as shown in the following formula. The factors *likes*, *shares*, *comments*, and *views* stem from the individual media items as described in section 9.3 and section 9.4. The factor *clusterSize* corresponds to the size of the current cluster. After some experimentation with different event media items sets, the factor *recency* was empirically determined to be calculated as follows. If the youngest media item in the cluster is less than or exactly one day old, the value of recency is 8, for two days it is 4, for three days it is 2, and for each day more, the value is 1. The factor *quality* is a representation of the presence of faces and a media item's photo or video quality.

$$\begin{aligned}&\alpha \times \mathit{likes} + \beta \times \mathit{shares} + \gamma \times \mathit{comments} + \delta \times \mathit{views} +\\ &\epsilon \times \mathit{clusterSize} + \zeta \times \mathit{recency} + \eta \times \mathit{quality}\end{aligned} \tag{9.1}$$

Empirically optimized default values that can be fine-tuned for a concrete media item set were determined as follows: $\alpha = 2$, $\beta = 4$, $\gamma = 8$, $\delta = 1$, $\epsilon = 32$, $\zeta = 2$, and $\eta = 8$. These factors follow the usage patterns of the different actions: viewing happens more often than liking, which in turn happens more often than commenting, *etc.* We describe the evaluation of the ranking formula in the upcoming section.

9.5 Evaluation

Evaluating subjective data like *the* correct ranking for a set of media items, is a challenging task. For different users, there may be different optimal ranking parameter settings. A common subjective evaluation technique is the previously introduced *Mean Opinion Score (MOS)* [2]. Given a subjective evaluation criterion like the correctness of a ranking, MOS provides a meaningful way to judge the overall quality of our approach.

9.5.1 Event Analyses by Social Networks

We have evaluated our approach with the (at time of writing) recent event of the Super Bowl XLVII,[1] which was an American football game between the American Football Conference champion Baltimore Ravens and the National Football Conference champion San Francisco 49ers to decide the National Football League champion for the 2012 season. The Ravens defeated the 49ers by the score of 34–31. This event received broad social media coverage and the social networks Twitter, Instagram, and Facebook all

[1] `http://en.wikipedia.org/wiki/Super_Bowl_XLVII`, accessed July 15, 2013

published blog posts with analyses of the event on the respective networks, whereas the search engine Google published an analysis of trending queries during the match. In the following, we provide summaries of these different analyses, with the expectation to encounter relevant media items for each of the mentioned highlights in our final ranked list of media items stemming from the various social networks.

According to Twitter's analysis,[1] the five moments that generated the most tweets during the game ordered by decreasing number of tweets per minute were the power outage, the 108-yard kickoff return for the Ravens touchdown by Jones, the moment when the clock expired and the Ravens won, Jones catches a 56 yard pass for a Ravens touchdown, and the Gore touchdown for the 49ers. Overall, 24.1 million tweets about the game and halftime show were counted, a number that even leaves aside the advertisements, which in recent years have become a highly expected highlight of the Super Bowl experience. The Twitter article further mentions the performance by superstar artist Beyoncé and a number of Super Bowl advertisements as highlights of the event.

Instagram's analysis[2] mentions that more than three million photos with Super Bowl-related terms in their captions were shared and at peak more than 450 photos about the game were posted every second. During the halftime show, over 200 photos per second were posted about Beyoncé. The blog post further highlights how a TV channel pointed to selected photos and explains that brands ran Instagram campaigns. According to Instagram, people used Instagram both directly at the event venue, but also while watching from home, either by photographing their TV sets, or by photographing each other how they watched the event.

Facebook's analysis[3] mentions as top five most-talked-about moments of the Super Bowl the moment when the Ravens won the Super Bowl, Beyoncé's halftime performance, the power outage in the Superdome, Jacoby Jones' 108-yard kickoff return for a Ravens touchdown, and Joe Flacco's 56-yard pass to Jacoby Jones for a Ravens touchdown. The Super Bowl was nicknamed the Harbaugh Bowl, as both teams' head coaches are named Harbaugh as last name. Facebook also mentions Alicia Keys' performance of the National Anthem as special event.

[1] http://blog.twitter.com/2013/02/the-super-tweets-of-sb47.html, accessed July 15, 2013

[2] http://blog.instagram.com/post/42254883677/sbroundup, accessed July 15, 2013

[3] http://newsroom.fb.com/News/570/Super-Bowl-XLVII-on-Facebook, accessed July 15, 2013

The search engine Google has compiled a list of top trending search terms during the match, with the top ones being the sponsor M&M's, Beyoncé, Baltimore Ravens, San Francisco 49ers, and Colin Kaepernick (quarterback for the San Francisco 49ers). Additional spiking search terms were power outage and Chrysler (driven by an advertisement during the game). Further advertisement-related search terms were for advertisements for M&M's, Mercedes-Benz, Disney's Oz Great and Powerful movie, Lincoln, and Audi.

While (at time of writing) no separate statistics are available for the video hosting platform YouTube, Google's blog post mentions that searches for Gangnam Style were trending on YouTube, along with searches for big game performers in form of the artists Alicia Keys and Beyoncé.

9.5.2 Expected Super Bowl Media Items

Given the differing social networks' own analyses that we have summarized in the previous subsection, we expect to see media items on at least the following topics (in no particular order) in our own ranked media item set.

the power outage
the performances of Beyoncé and Alicia Keys
the advertisements
the match itself from people at the stadium
the Super Bowl watchers around the world

Figure 9.1 and Figure 9.2 show media items for the search bundle *49ers* and *Baltimore Ravens* arranged in two different media gallery styles, loose order and strict order. Search bundles are combined separate searches, *i.e.*, we first searched for *49ers* and then for *Baltimore Ravens* and combined the results as if we had performed just one search. For details on the automated media gallery generation, we refer the reader to the upcoming Chapter 10.

9.5.3 Evaluation Approach

We asked three human raters to agree on a rating for the media items that were retrieved for the two queries 49ers and Baltimore Ravens. We have made the dataset of media items available.[1] We then fine-tuned the weight factors $\alpha, \beta, \gamma, \delta, \epsilon, \zeta$, and η of the ranking formula that was introduced in subsection 9.4.2 until the highest possible agreement

[1] `https://www.dropbox.com/sh/30qwuvphcv49max/ilsaMbSdf6`, accessed July 15, 2013

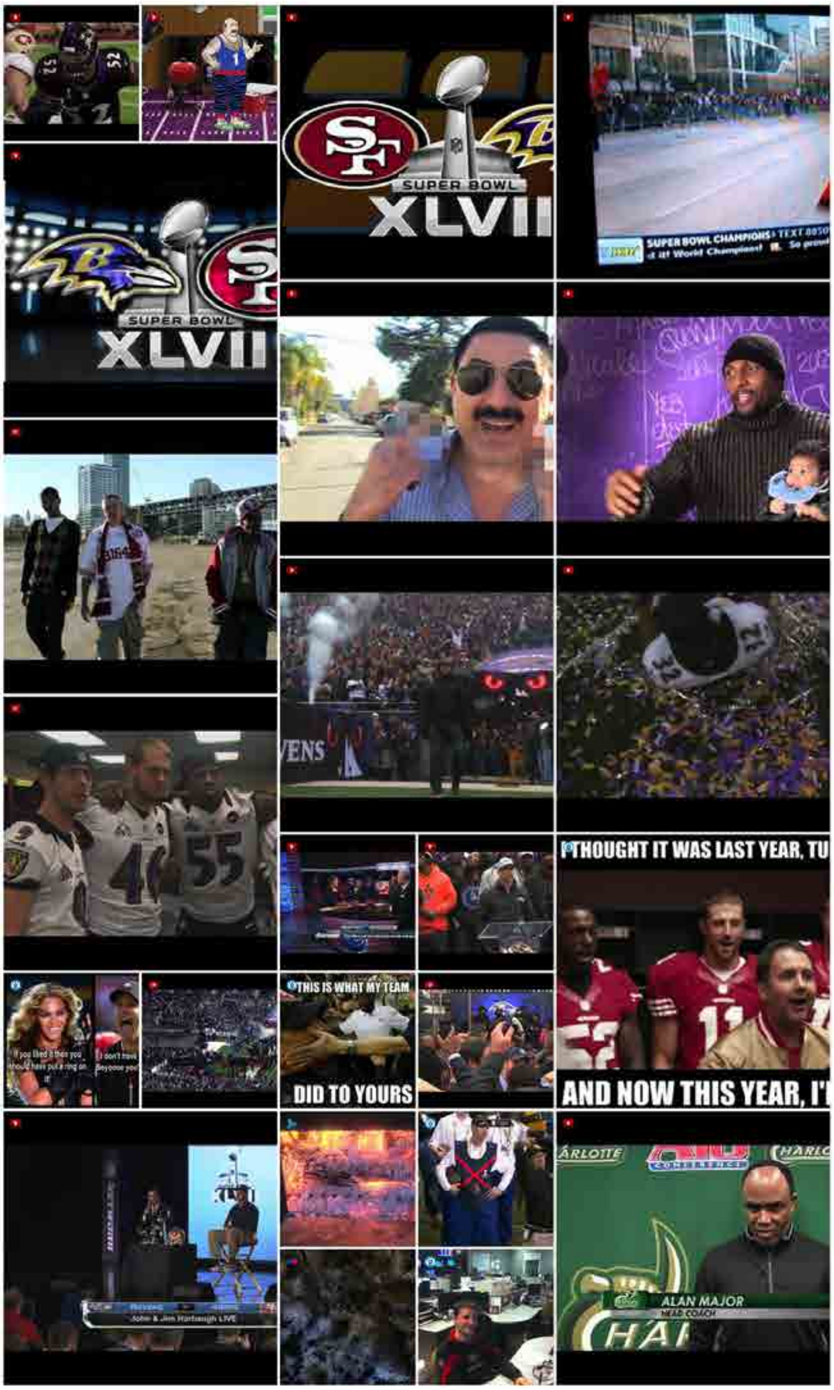


Figure 9.1: Ranked Super Bowl XLVII media gallery in Loose Order, Varying Sizes (LOVS) style

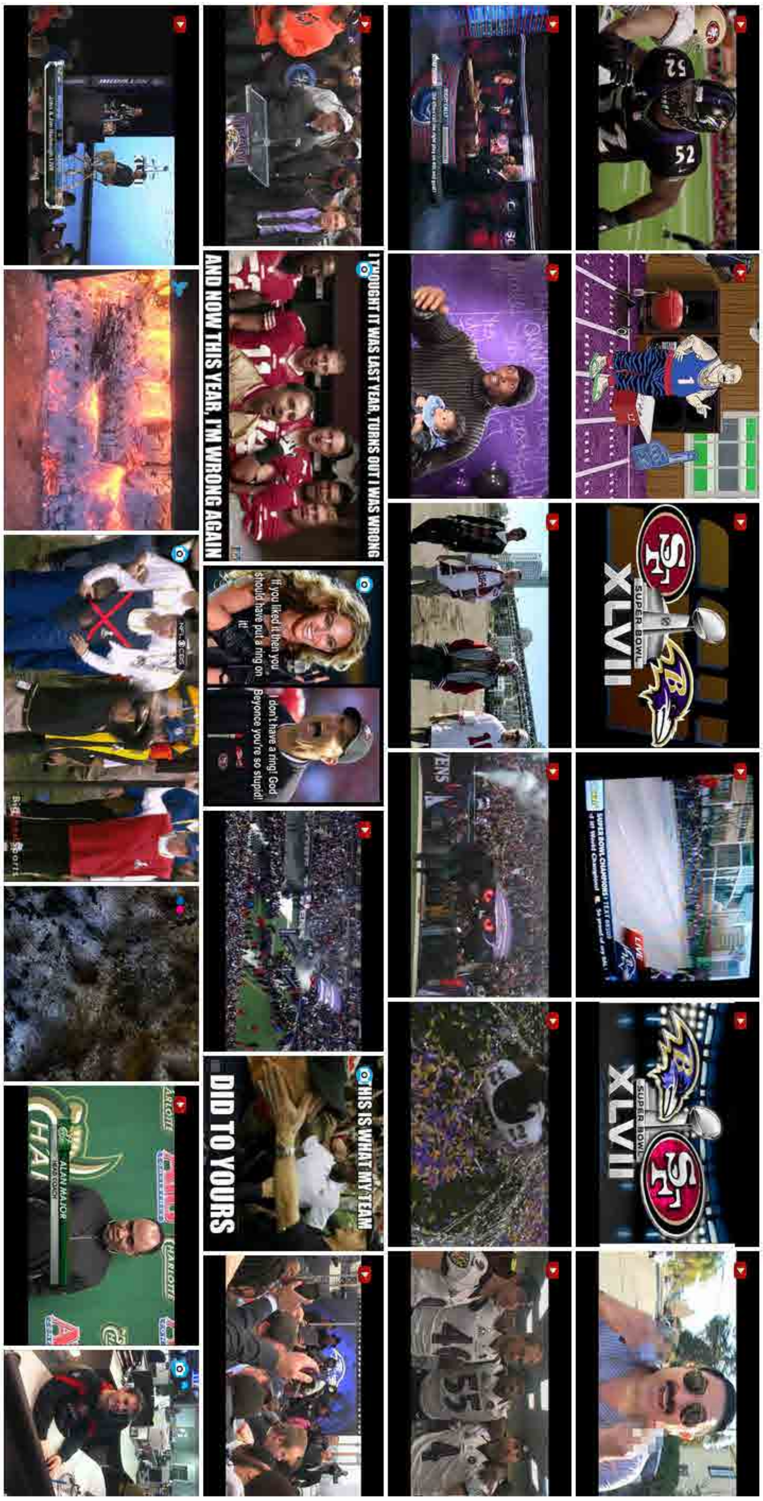
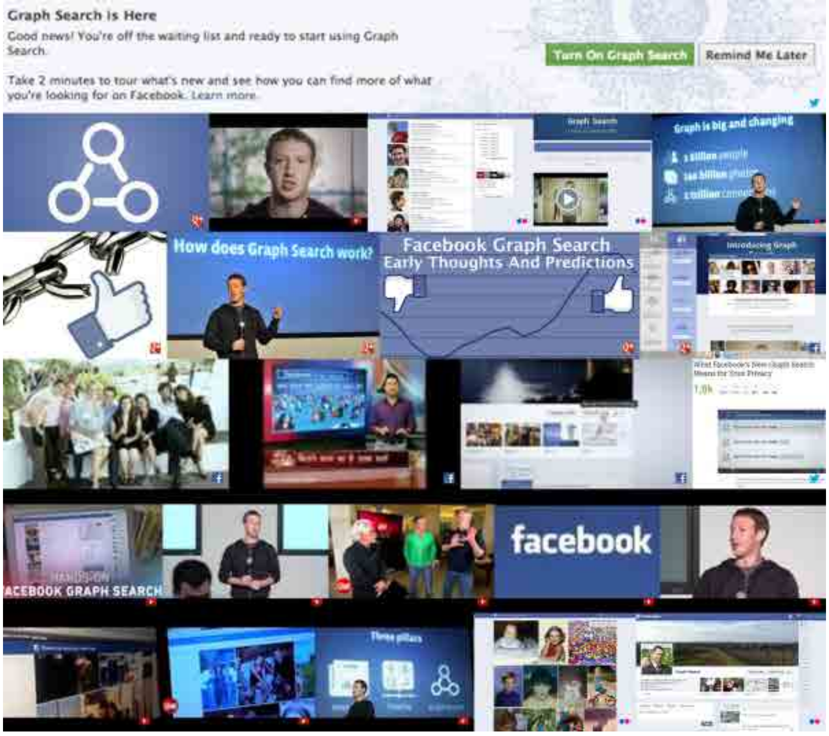

I THOUGHT IT WAS LAST YEAR, TURNS OUT I WAS WRONG
AND NOW THIS YEAR, I'M WRONG AGAIN
If you liked it then you should have put a ring on it!
I don't have a ring! God Beyonce you're so stupid!
THIS IS WHAT MY TEAM DID TO YOURS
SUPER BOWL XLVII
SUPER BOWL XLVII

Figure 9.2: Ranked Super Bowl XLVII media gallery in Strict Order, Equal Sizes (SOES) style

between the human-generated ranking and the system-generated ranking was reached. Afterwards, we tested these empirically determined weight factors of the ranking formula with different events and asked the evaluators to what extent on a MOS scale from 1–5 they agreed with each of the top-10 ranked media items. Rating the ranking of *all* media items is barely possible, which is why we limit ourselves to rating the top-10 returned results, a technique that is also known as *pooling* in the Information Retrieval community [4]. Screenshots of some of the events we tested with and the particular top-ranked media items can be seen in Figure 9.5, Figure 9.4, and Figure 9.3.



Graph Search is Here

Good news! You're off the waiting list and ready to start using Graph Search.

Take 2 minutes to tour what's new and see how you can find more of what you're looking for on Facebook. Learn more.

Figure 9.3: Ranked media gallery for the Facebook Graph Search launch event on January 15, 2013

9.5.4 Evaluation Results

In this subsection, we present the human raters' results in the form of MOS scores of the events that we tested our ranking formula with. As can be seen in Table 9.2 and as the combined MOS of 3.7 (variance 0.8) suggests, all selected top-10 media items were overall considered relevant by the human raters. We arranged for post-test

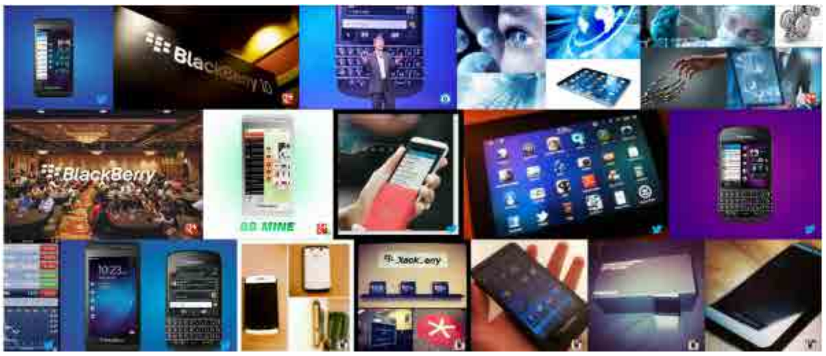


Figure 9.4: Ranked media gallery for the BlackBerry 10 launch event on January 30, 2013

Rank Event	1	2	3	4	5	6	7	8	9	10	avg	var
Facebook	3.8	3.3	4.9	4.2	3.9	5.0	3.7	4.8	4.8	3.1	**3.6**	**0.4**
BlackBerry	4.9	4.8	5.0	3.2	5.0	4.4	4.1	3.8	4.5	2.7	**3.8**	**0.8**
Qualcomm	4.9	4.7	4.9	5.0	2.4	4.1	3.1	5.0	2.1	3.7	**3.6**	**1.0**
Super Bowl	5.0	4.1	5.0	3.2	5.0	2.8	4.1	4.6	3.3	4.0	**3.9**	**0.9**

Table 9.2: Mean Opinion Scores (MOS) for the top-10 ranked media items of four events (overall MOS: 3.7, variance: 0.8)

conversations with each human rater in order to understand their motivations for their ratings. After talking to the human raters it became clear that lower scores were mostly caused by outliers in the media item set. Further discussions with the human raters revealed that if at a first glance the context to the event in question was missing, the raters heavily downgraded the corresponding media items. We note that in order to test the visual ranking aspect in isolation, raters were exclusively shown media items without any micropost context. This approach made it hard for them to recognize connections between *remote* media items to events *on-site*. Examples of remote media items are media items of Super Bowl watchers around the world or photo montages of online news media (see Facebook Graph Search launch, BlackBerry 10 launch). Another reason for outliers according to the raters were too small media item previews of videos that only made sense when seen in motion.

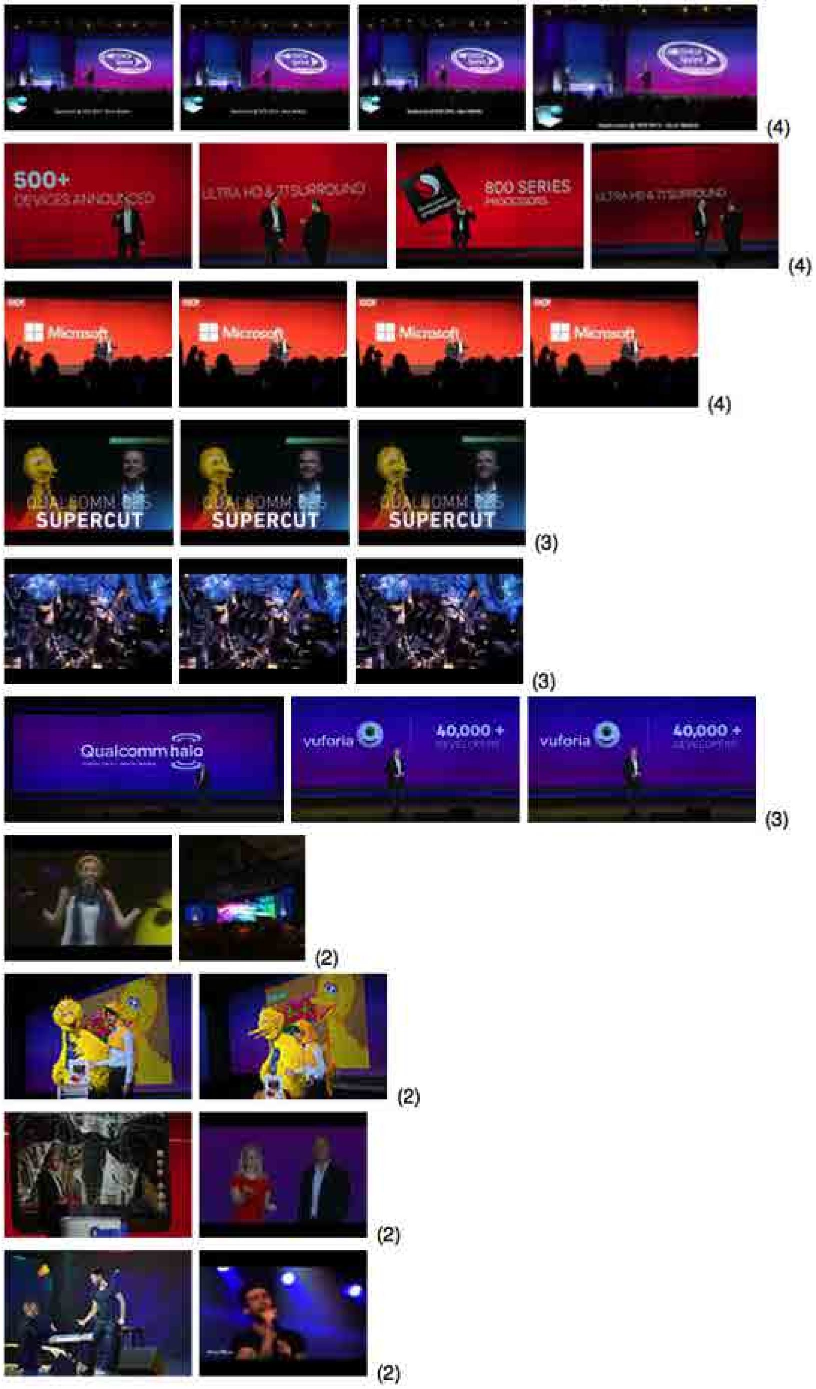
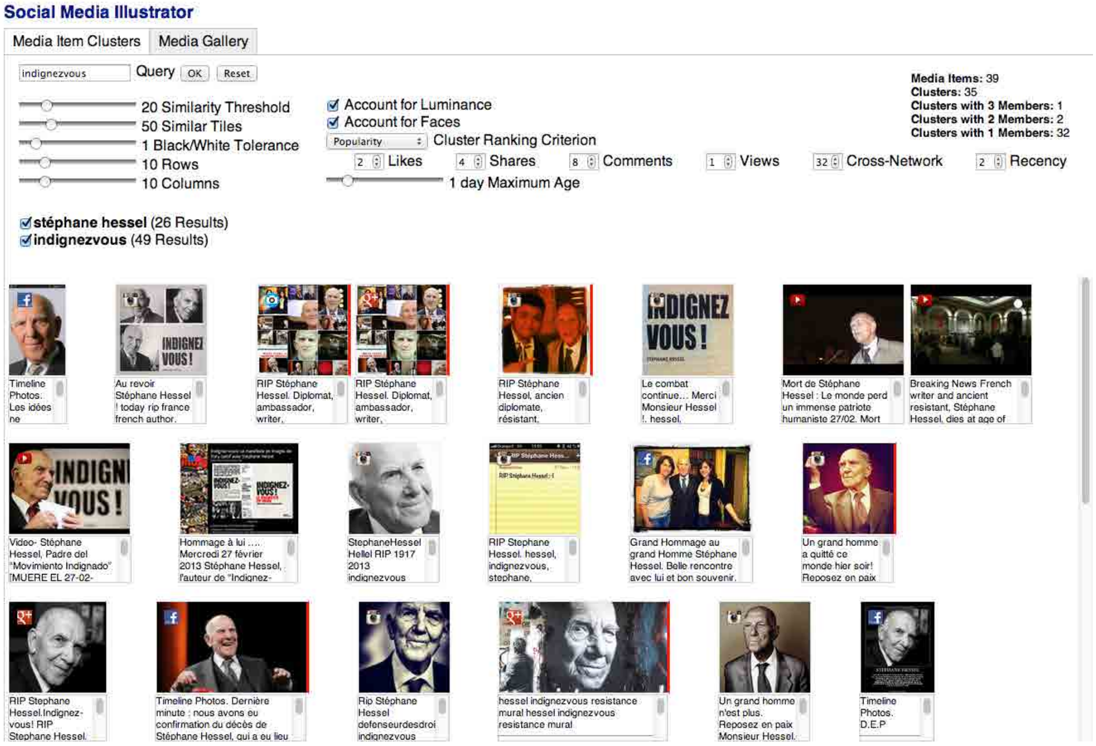


Figure 9.5: Ranked media items for the Qualcomm CES 2013 keynote event on January 8, 2013 (raw cluster view)



indignezvous

20 Similarity Threshold
50 Similar Tiles
1 Black/White Tolerance
10 Rows
10 Columns

☑ Account for Luminance
☑ Account for Faces
Popularity Cluster Ranking Criterion
2 Likes 4 Shares 8 Comments 1 Views 32 Cross-Network 2 Recency
1 day Maximum Age

Media Items: 39
Clusters: 35
Clusters with 3 Members: 1
Clusters with 2 Members: 2
Clusters with 1 Members: 32

☑ **stéphane hessel** (26 Results)
☑ **indignezvous** (49 Results)

Timeline Photos: Les idées ne

Au revoir Stéphane Hessel ! today rip france french author.

RIP Stéphane Hessel. Diplomat, ambassador, writer,

RIP Stéphane Hessel. Diplomat, ambassador, writer,

RIP Stéphane Hessel, ancien diplomate, résistant,

Le combat continue... Merci Monsieur Hessel !. hessel,

Mort de Stéphane Hessel : Le monde perd un immense patriote humaniste 27/02. Mort

Breaking News French writer and ancient resistant, Stéphane Hessel, dies at age of

Video- Stéphane Hessel, Padre del "Movimiento Indignado" ¡MUERE EL 27-02-

Hommage à lui .... Mercredi 27 février 2013 Stéphane Hessel, l'auteur de "Indignez-

StephaneHessel Hellel RIP 1917 2013 indignezvous

RIP Stephane Hessel. hessel, indignezvous, stephane,

Grand Hommage au grand Homme Stéphane Hessel. Belle rencontre avec lui et bon souvenir.

Un grand homme a quitté ce monde hier soir! Reposez en paix

RIP Stephane Hessel.Indignez-vous! RIP Stephane Hessel.

Timeline Photos. Dernière minute : nous avons eu confirmation du décès de Stéphane Hessel, qui a eu lieu

Rip Stéphane Hessel defenseurdesdroi indignezvous

hessel indignezvous resistance mural hessel indignezvous resistance mural

Un grand homme n'est plus. Reposez en paix Monsieur Hessel.

Timeline Photos. D.E.P

Figure 9.6: Media item ranking in our application with adjustable weight factors for the ranking formula

9.6 Conclusions

In this chapter, we have detailed criteria that can be considered for the creation of a ranking. We have shown how we abstract social interactions on social networks and introduced an algorithm to merge social interactions when media items are clustered. Additionally, we have detailed an algorithm for the selection of a media item cluster's visual representative. Finally, we have used the previously introduced aspects for the definition of a ranking formula whose weight factors were empirically determined with a representative sports event and successfully evaluated in practice with three other events. For the task of ranking media items, there is no single one correct solution, but only optimizations under given constraints. Albeit a maximum agreement between human raters is aimed for, individual users will always have different ranking needs. With our ranking formula proposition, we want to help such individual users with a rationally traceable and useful default ranking that can be adapted easily to special needs or media item sets. We have implemented this ranking formula in a stand-alone application,which will be described in more detail in section 10.7. With this application, users can easily modify the weight factors to see their effects on-the fly. Besides all possible customization, the application still proposes reasonable default values for each individual parameter that have been demonstrated to work well for the majority of cases. Figure 9.6 shows a screenshot of this application that effectively helps users see the media items needles first and the haystack last.

Chapter Notes

This chapter is partly based on the following publications.

- Thomas Steiner. "A meteoroid on steroids: ranking media items stemming from multiple social networks". In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW '13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 31–34. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2487798`.

- Vuk Milicic, Giuseppe Rizzo, José Luis Redondo Garcia, Raphaël Troncy, and Thomas Steiner. "Live topic generation from event streams". In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW '13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 285–288. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2487924`.

- Thomas Steiner, Seth van Hooland, Ruben Verborgh, Joseph Tennis, and Rik Van de Walle. "Identifying VHS Recording Artifacts in the Age of Online Video Platforms". In: *Proceedings of the 1st international Workshop on Search and Exploration of X-rated Information.* Feb. 2013. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2013/identifying-vhs-recording-sexi2013.pdf`.

References

[1] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, et al. “The YouTube Video Recommendation System”. In: *Proceedings of the 4th ACM Conference on Recommender Systems*. RecSys ’10. Barcelona, Spain: ACM, 2010, pp. 293–296.

[2] International Telecommunication Union, Telecommunication Standardization Sector. *ITU-T Recommendation P.800: Methods for Subjective Determination of Transmission Quality*. `http://www.itu.int/rec/T-REC-P.800-199608-I/en`, accessed July 15, 2013. Aug. 1998.

[3] Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer. “Combining Audio-based Similarity with Web-based Data to Accelerate Automatic Music Playlist Generation”. In: *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*. MIR ’06. Santa Barbara, California, USA: ACM, 2006, pp. 147–154.

[4] Tie-Yan Liu. “Learning to Rank for Information Retrieval”. In: *Found. Trends Inf. Retr.* 3.3 (Mar. 2009), pp. 225–331. ISSN: 1554-0669.

[5] Vuk Milicic, Giuseppe Rizzo, José Luis Redondo Garcia, Raphaël Troncy, and Thomas Steiner. “Live topic generation from event streams”. In: *Proceedings of the 22nd international conference on World Wide Web companion*. WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 285–288. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2487924`.

[6] Pere Obrador, Michele Saad, Poonam Suryanarayan, and Nuria Oliver. “Towards Category-Based Aesthetic Models of Photographs”. In: *Proceedings of the 18th International Conference on Advances in Multimedia Modeling – Volume Part I (MMM 2012)*. 2012, pp. 63–76.

[7] Philipp Sandhaus, Mohammad Rabbath, and Susanne Boll. “Employing Aesthetic Principles for Automatic Photo Book Layout”. In: *Proceedings of the 17th International Conference on Advances in Multimedia Modeling – Volume Part I (MMM 2011)*. 2011, pp. 84–95.

[8] Thomas Steiner. "A meteoroid on steroids: ranking media items stemming from multiple social networks". In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW '13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 31–34. ISBN: 978-1-4503-2038-2. URL: http://dl.acm.org/citation.cfm?id=2487788.2487798.

[9] Thomas Steiner, Seth van Hooland, Ruben Verborgh, Joseph Tennis, and Rik Van de Walle. "Identifying VHS Recording Artifacts in the Age of Online Video Platforms". In: *Proceedings of the 1st international Workshop on Search and Exploration of X-rated Information.* Feb. 2013. URL: http://www.lsi.upc.edu/~tsteiner/papers/2013/identifying-vhs-recording-sexi2013.pdf.

[10] Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Facebook Microposts via a Mash-up API and Tracking its Data Provenance". In: *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on.* Oct. 2011, pp. 342–345. URL: http://research.google.com/pubs/archive/37426.pdf.

[11] Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, and Rik Van de Walle. "Defining Aesthetic Principles for Automatic Media Gallery Layout for Visual and Audial Event Summarization based on Social Networks". In: *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on.* July 2012, pp. 27–28. URL: http://www.lsi.upc.edu/~tsteiner/papers/2012/defining-aesthetic-principles-for-automatic-media-gallery-layout-qomex2012.pdf.

10

Media Item Compilation

10.1 Introduction

In this chapter, we introduce aesthetic principles for the automated generation of media galleries based on media items retrieved from social networks that—after a ranking and pruning step—can serve to authentically summarize events and their atmosphere from a visual and an audial standpoint. Mobile devices such as smartphones, together with social networks, enable people to create, share, and consume media items like videos or photos. These devices accompany their owners almost everywhere and are thus omnipresent at all sorts of events. At such events—given a stable network connection—part of the event-related media items are published on social networks both as the event happens or afterwards, once a stable network connection has been established again. Ranked media items stemming from multiple social networks can serve to create authentic media galleries that illustrate events and their atmosphere. A key feature for this task is the semantic enrichment of media items and associated microposts and the extraction of *visual*, *audial*, *textual*, and *social* features. Based on this set of features, additional *aesthetic* features can be defined and exploited to obtain appealing and harmonic media galleries.

10.2 Related Work

While enormous efforts have been made to extract *visual*, *audial*, *textual*, and *social* features from media items and microposts on social networks in *isolation*, to the best of our knowledge, remarkably less initiatives concern the extraction and the application of all those features *in combination* for *all* types of media items, including microposts. In [24], Sandhaus *et al.* consider visual and aesthetic features for the automated creation of photo books. Obrador *et al.* use visual and aesthetic features for a category-based approach to automatically assess the aesthetic appeal of photographs [22]. In [19], Knees *et al.* use audial and textual features for the automatic generation of music playlists. Choudhury *et al.* show in [11] how social and textual features can be used to achieve precise detection results of named entities and significant events in sports-related microposts. In [12], Davidson *et al.* show how visual, textual, and social features can be used for personalized video recommendations. A service called Storify [1, 15] lets users manually combine microposts, photos, videos, and other elements onto one page for the purpose of storytelling or summarizing an event and share stories permanently on the Web. Finally, social networks present photos and videos often in grid-like galleries,[1] sometimes scaled based on the amount of comments. When unique media items have been collected, the remaining task is to summarize events by selecting the most relevant media items or media fragments. Fabro and Böszörményi [13] detail the summarization and presentation of events from content retrieved from social media. Nowadays, many domain-specific methods already exhibit good accuracy, for example, in the sports domain [20, 21]. However, the challenge is to find content-agnostic methods. Methods that exploit semantic information (*e.g.*, [10]) will likely provide high-quality results in the near future, but today's most relevant summaries are still produced by manual—at best semi-automated—user interaction [23].

10.3 Media Gallery Aesthetics

Definition: A media gallery is a compilation of photos or videos retrieved from social networks that are related to a given event. Given a set $M_{start} = \{m_1, ..., m_n\}$ of media items related to a certain event, a ranking formula f, and a ranking threshold t, the resulting subset $M_{final} \subset M_{start}$ is the result after the application of f to M_{start}: $f(M_{start}) = M_{ranked}$ and pruning the ranked set M_{ranked} to only include members whose rank is greater than t, with the resulting set named M_{final}. Each media item m_i can either be an instance of video or photo. For each point t_x on a timeline T, the state of the media gallery at t_x is defined for each media item m_i as a set S_x of n tuples

[1] `http://twitpic.com/904yka/full`, accessed July 15, 2013

$s_{x,i}$, where $s_{x,i} = \langle$*left*, *top*, *width*, *height*, *alpha*, *z-index*, *animation*, *start*, *playing*, *volume*$\rangle$. The first six properties are defined as in CSS [4], the *animation* property allows for the definition of CSS transitions as defined in [18] and CSS transformations as defined in [16], the *start* property defines the start time in a video. The property *playing* describes whether a video is currently paused or playing. Finally, the property *volume* describes the volume level of a video. A schematic media gallery at t_x can be seen in Figure 10.1.

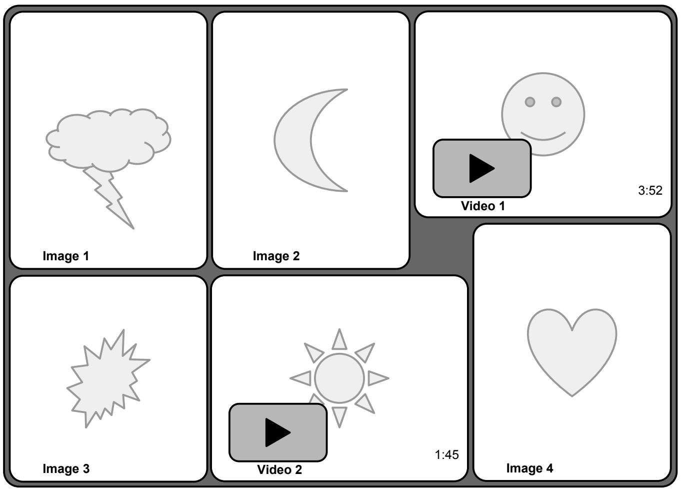


Figure 10.1: Schematic media gallery with four photos and two videos

Audial aesthetics: Audial aesthetics thus consist of aspects like volume level normalization, avoiding multiple videos playing music in parallel, smooth transitions, *etc.* We remark that through selective mixing of audio tracks of event-related videos, "noise clouds" that are very characteristic for an event's atmosphere can be observed. We support this by allowing users to play more than one video at a time.

Visual aesthetics: Visual aesthetics are determined by the composition, *i.e.*, the relation of the number of photos *vs.* the number of videos *globally*, *per coherent scene*, and per *point in time*. In order to avoid cognitive overload of viewers, the number of visible (moving) media items at a time should be limited. We will treat this topic in more detail in subsection 10.5.1. Depending on the event, a consistent or a contrast-rich overall appearance and transitions of items may be desired.

10.4 Motivation for Automated Media Gallery Generation

Media galleries (see Figure 10.2 and Figure 10.3 for examples) help users consume significant amounts of media items in an ideally pleasing and aesthetic way. These media items may—or, more commonly: may not—be ordered, besides an intrinsic chronologic order. In the context of our work on summarizing events based on microposts and media items stemming from multiple social networks, we have created methods to first *extract* event-related media items from multiple social networks, second, to *deduplicate* near- and exact-duplicate media items, third, to *cluster* them by visual similarity, and finally, to *rank* the resulting media item clusters according to well-defined ranking criteria. In this chapter, we treat the challenge of *compiling* ranked media item clusters in media galleries in ways such that the ranking-implied order is at least loosely respected. In the previous section, we have defined aesthetic principles for automated media gallery layout [26], which we now apply to media gallery styles. The task of media gallery compilation is different from the widely researched task of photo book generation, as media galleries can contain both photos and videos. Different types of media gallery layouts are possible, two of which we have implemented and evaluated via two different user studies: one with, and one without detailed user comments.

10.5 Media Gallery Styles

Media galleries—in contrast to free-form digital media collages—necessarily display media items in a grid-like way. The crucial question is thus whether the media items' aspect ratios should be respected, or whether they should be cropped to square, or other aspect ratios (*e.g.*, 4:3 or 16:9). Respecting the aspect ratio has the advantage that media items do not need to be cropped, which may affect important contained information like, *e.g.*, contained faces or text. However, due to the unpredictable media item formats, compiling media galleries that do not look frayed is harder. The advantage of cropping is that media gallery layout is easier, as the media item formats are predictably the same, at the cost of having to decide where to crop. Different algorithms (*e.g.*, [27]) beyond this chapter's scope exist to aid this decision. Common cropping heuristics include maximizing the number of included faces, focusing on the detected center of the media item, or cropping at detected background areas.

Terminology: We use the following terminology. A media gallery is called *balanced* if its shape is rectangular, *hole-free* if there are no gaps from missing media items, and *order-respecting* if media items appear in insertion order. Optimal media galleries fulfill all three conditions.

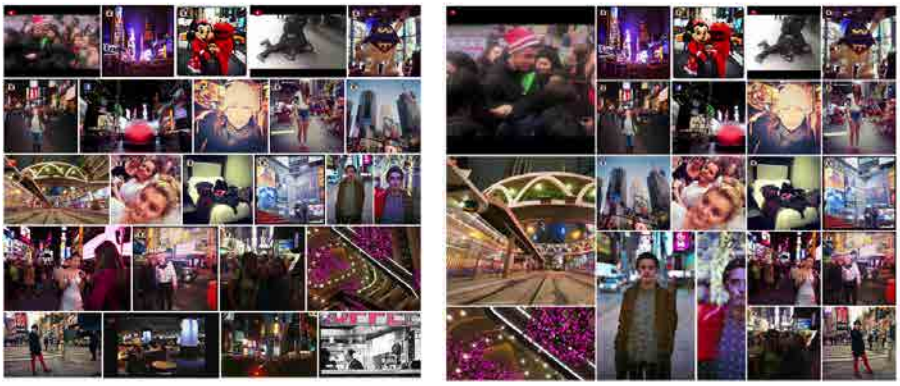
Non-Order-Respecting Styles: An interesting technique for arranging media items is dividing. Paper sizes that follow the ISO 216 standard[1] are the most common everyday examples of the dividing principle: every media item with an aspect ratio of $\sqrt{2}$ can be divided into two media items with the same aspect ratio [9]. This works for portrait and landscape orientations, however, is not necessarily order-respecting. Two other non-order-respecting techniques are (i) working with pre-defined placeholder patterns (small and big squares, portrait and landscape rectangles) and then filling the placeholder shapes with media items [6] or (ii) working with columns of pre-defined widths and then iteratively inserting in the smallest column [8].

As outlined in section 10.4, we require (at least loosely) order-respecting media galleries for the ranking step to make sense. In the upcoming two paragraphs, we will introduce two techniques for the generation of such media galleries.

Strict Order, Equal Size (SOES): A media gallery style that we call *Strict Order, Equal Size (SOES)*, which strictly respects the ranking-implied order is presented in [5]. Examples can be seen in Figure 10.2a and Figure 10.3a. The algorithm works by resizing all media items in a row to the same height and adjusting the widths in a way that the aspect ratios are maintained. A row is filled until a maximum row height is reached, then a new row (with potentially different height) starts, *etc.* This media gallery style is strictly order-respecting, hole-free, and can be balanced by adjusting the number of media items in +1 steps.

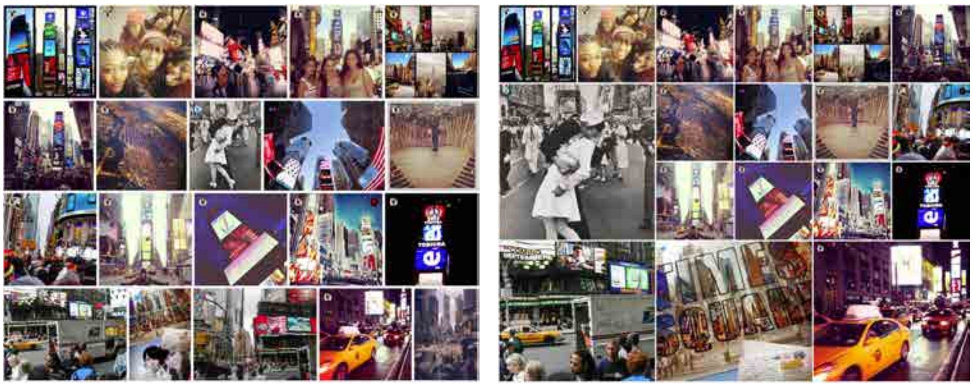
Loose Order, Varying Size (LOVS): Examples of a media gallery style that we call *Loose Order, Varying Size (LOVS)* can be seen in Figure 10.2b and Figure 10.3b, with the details explained in [7]. The algorithm works by cropping all images to a square aspect ratio, which allows for organizing media items in a way such that one big square

[1] `http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=36631`, accessed July 15, 2013



(a) *SOES*, Survey A **(b)** *LOVS*, Survey A

Figure 10.2: Survey A: Media galleries visualizing a gathering at Times Square, New York on February 8, 2013



(a) *SOES*, Survey B **(b)** *LOVS*, Survey B

Figure 10.3: Survey B: Media galleries visualizing a gathering at Times Square, New York on February 8, 2013

always contains two horizontal blocks, each with two pairs of small squares. The media gallery is then formed by iteratively filling big or small squares until a square is full and then adding the square to the smallest column. This media gallery style allows any media item to become big, while still being loosely order-respecting and always hole-free. Balancing the gallery is slightly harder, as depending on the shape both small *and* big media items may be required.

10.5.1 Discussion and Evaluation

The main motivation for the *Loose Order, Varying Size* style is that certain media items can be featured more prominently by making them big, while still loosely respecting the ranking-implied order. Examples of to-be-featured media items can be videos, media items with faces, media items available in High-Density quality, or media items with interesting details [27]. Users may want to decide what media items to feature, albeit we aim for an automated solution. Evaluating subjective data like *the* correct presentation form for a set of media items is a challenging task. For different users and different media galleries, there may be different optimal settings. Again, we use the Mean Opinion Score (MOS, [17]) for our evaluation, as previously motivated.

User Studies: We have conducted two types of surveys. Survey A via multiple social networks, where we simply asked people to "Like" and/or comment on their favorite style of media gallery and Survey B that was distributed via email to a company-internal "miscellaneous" mailing list, where we asked people to rate media galleries via MOS, with optional comments. Survey A and Survey B used different media items in the media galleries, as to have some measure in how far content has an impact.

Survey A—Via Social Networks: For Survey A on the social networks Twitter, Facebook, and Google+, we had overall 16 participants (7 female, 8 male, 1 unknown). 7 users liked *SOES* more, whereas 9 users liked *LOVS* more. Interestingly, no user commented on why they liked *SOES* more. Users who commented on why they liked *LOVS* more mentioned they liked the additional structure and tidiness, the fact that some media items were bigger, the fact that it was easier to identify individual media items, and the fact that important media items were highlighted.

Survey B—Via Email: For Survey B via email with MOS ratings, we had 19 participants (6 female, 13 male). The majority of users who liked *LOVS* more mentioned that the different sizes gave the eye focal points and orientation, whereas one user explicitly disliked this guidance. Users liked the harmony and the structure. Two users mentioned that small media items were proportionally too small. Regarding *SOES*, users reported they felt overloaded and did not know where to start. Some users said the layout was boring and that, while they liked the outer framing, they were confused by the irregular inner grid. The MOS for *SOES* was 2.39 (variance 0.68), the MOS for *LOVS* was 4.17 (variance 0.47). The data of Survey B is available.[1] For privacy reasons, we have posted the media galleries as non-public microposts on social networks, which is why we are unable to share the data of Survey A.

10.5.2 Maintaining Provenance Information with Downloadable Media Galleries

Media galleries consist of individual media items, each with its specific pieces of provenance data like creator, originating social network, *etc.* In the context of the application that we have developed in the context of this thesis and that will be described in full detail in section 10.7, this provenance data is maintained in the form of HTML hyperlinks back to the originating social networks. However, when media galleries get downloaded in form of one static image dump, this is no longer the case. In consequence, we generate a caption-like image legend that gets added to each downloaded media gallery. An example of a media gallery with provenance data generated in the context of the 2013 Taksim Gezi Park protests[2] can be seen in Figure 10.4.

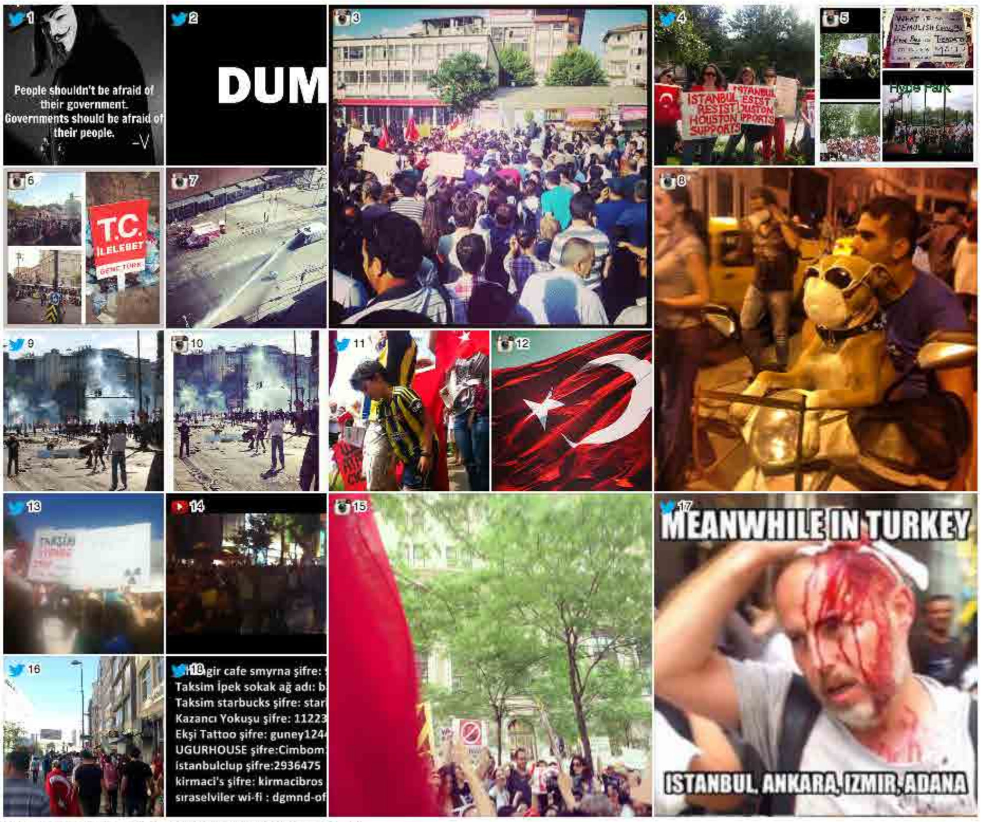
10.6 Interactive Media Galleries

Traditional slideshows: Up to now, we have presented different algorithms to generate media galleries of different styles and static media galleries, including a way to preserve provenance data when media galleries get downloaded as one image. Such media galleries are useful in the context of static media, such as newspapers or embedded in (online or offline) news articles. A first step towards more interactive media galleries are so-called slideshows. An exemplary slideshow, courtesy of the BBC's Online division, can be seen in Figure 10.5.

Media gallery paradigm slideshows: We have opted to extend the traditional slideshow model by adhering to the media gallery paradigm of our two preferred styles Loose Order, Varying Size (*LOVS*) and Strict Order, Equal Size (*SOES*). Given a media gallery in either *LOVS* or *SOES* style, each media item can be focused, *i.e.*, be put

[1] `http://bit.ly/media-gallery-survey`, accessed July 15, 2013

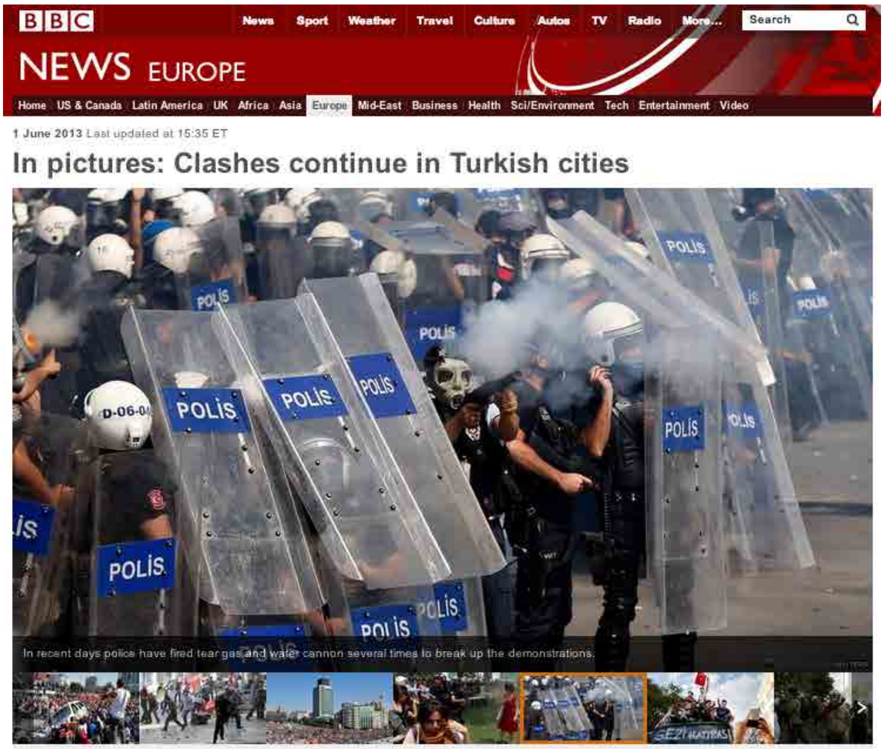
[2] `http://en.wikipedia.org/wiki/2013_Taksim_Gezi_Park_protests`, accessed July 15, 2013


1
People shouldn't be afraid of their government. Governments should be afraid of their people. -V
2
DUM
3
4
ISTANBUL RESIST HOUSTON SUPPORTS
5
Hyde Park
6
T.C. İLELEBET GENÇ TÜRK
7
8
9
10
11
12
13
14
15
16
17
MEANWHILE IN TURKEY
ISTANBUL, ANKARA, IZMIR, ADANA
18
gir cafe smyrna şifre:
Taksim İpek sokak ağ adı: b
Taksim starbucks şifre: star
Kazancı Yokuşu şifre: 11223
Ekşi Tattoo şifre: guney124
UGURHOUSE şifre:Cimbom
istanbulclup şifre:2936475
kirmaci's şifre: kirmacibros
sıraselviler wi-fi : dgmnd-of

[1] Source: http://twitter.com/bnazorhon/status/340706614942248960
[2] Source: http://twitter.com/atokmakchiev/status/340587069703335937
[3] Source: http://instagram.com/p/aBSxffDXbF/
[4] Source: http://twitter.com/zeynepgabrali/status/340852794137395201
[5] Source: http://instagram.com/p/aBEL2FtmzF/
[6] Source: http://instagram.com/p/aBosQflvsq/
[7] Source: http://instagram.com/p/aBmpcSpnn3/
[8] Source: http://instagram.com/p/aBPuV9rrjn/
[9] Source: http://twitter.com/fulyacimen/status/340846160413597698
[10] Source: http://instagram.com/p/aBUWnRFpgG/
[11] Source: http://twitter.com/zeynepgabrali/status/340860450864500736
[12] Source: http://instagram.com/p/aBoI3llQAy/
[13] Source: http://twitter.com/DenizhanKurban/status/340865987400323072
[14] Source: http://www.youtube.com/watch?v=9SA21GDtj2o&feature=youtube_gdata_player
[15] Source: http://instagram.com/p/aBxHFeQIBn/
[16] Source: http://twitter.com/atokmakchiev/status/340587069703335937
[17] Source: http://twitter.com/zmdursun/status/340910004385230848
[18] Source: http://twitter.com/DogaSzyz/status/340558456958377985

Figure 10.4: Downloadable media gallery with provenance data

prominently in the foreground. When a media item is focused, it smoothly transitions from its original location in the media gallery to the center, while in parallel it is zoomed to double its size. All other media items that are currently not focused are faded out in a black-and-white variant and blurred, in order to put the maximum emphasis to the currently focused media item. Figure 10.6 shows three steps of the described transitions and Figure 10.7 shows the final state after the transition. A short screencast of the whole animation is available online at the URL `https://vine.co/v/bT7eiwjE6DQ` (accessed July 15, 2013).

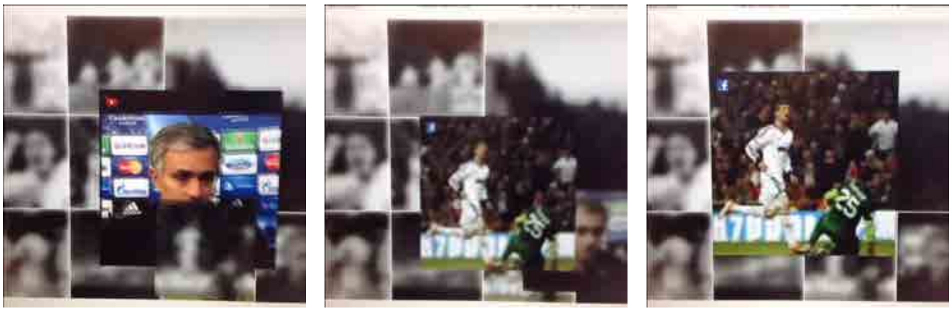


1 June 2013 Last updated at 15:35 ET

In pictures: Clashes continue in Turkish cities

In recent days police have fired tear gas and water cannon several times to break up the demonstrations.

Figure 10.5: Media gallery in form of a slideshow (Source and copyright: BBC Online `http://www.bbc.co.uk/news/world-europe-22740038`, accessed July 15, 2013)



(a) Animation step 1 **(b)** Animation step 2 **(c)** Animation step 3

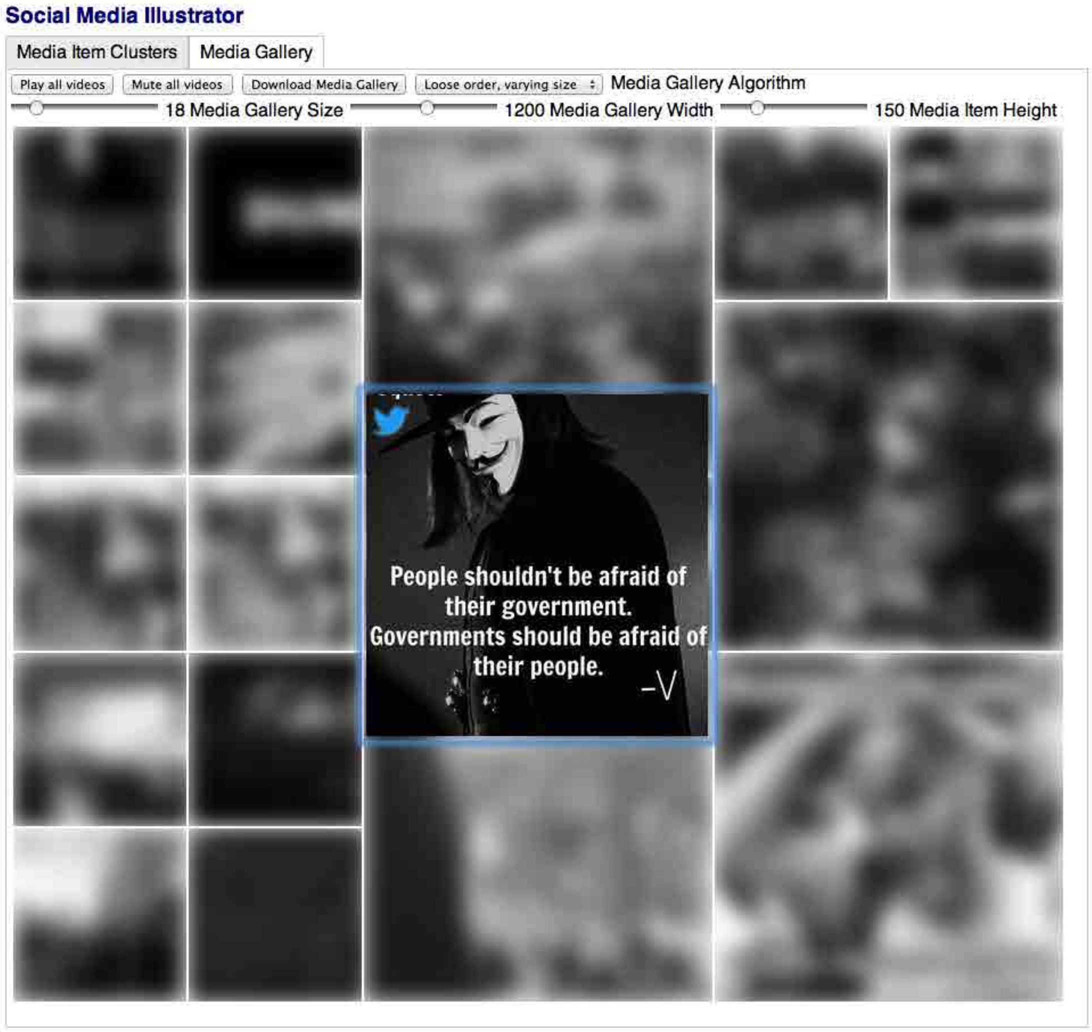
Figure 10.6: Three animation steps of interactive media gallery

Audial media galleries and text-to-speech synthesis: With our media galleries, we go yet another step further and add an audio component to the interactive slideshow. As media items of type video typically already have an audio track, it is thus straightforward to play the video once it is focused in the slideshow. In contrast, photos do not have audible information associated with them. We can, however, use the (potentially machine-translated, see subsection 4.3.2) textual information of any of the media items in the particular media item's media cluster (see section 8.3) and via speech synthesis create an audial experience. This follows the hypothesis that visually similar media items also share similar textual descriptions. We use the set of extracted and disambiguated named entities (see subsection 4.3.1) combined with the insights gained from part-of-speech tagging (see subsection 4.3.3) to select the one textual description from the entire set of textual descriptions in each cluster that *(i)* maximizes the number of named entities and that *(ii)* has the most diverse set of words identified as nouns, verbs, and adjectives. Following this heuristic, we effectively avoid choosing a textual description that only consists of a list of tags, as is often the case with, *e.g.*, Instagram (see subsection 3.2.6). We convert the textual description of a media item to audial information with the help of a text-to-speech system. We use the eSpeak [14] speech synthesizer that was described earlier in subsection 8.5.4.

10.7 Social Media Illustrator

Media galleries can be generated following the steps described in the previous chapters, starting with micropost annotation (Chapter 4), followed by event detection (Chapter 5), continuing with media item extraction (Chapter 6), over to media item deduplication and clustering (Chapter 7 and Chapter 8), media item ranking (Chapter 9) and finally media item compilation (Chapter 10). We have developed an application called *Social Media Illustrator* for the automated generation of media galleries that visually and audially summarize events based on media items like videos and photos from multiple social networks. The application is publicly available at `http://social-media-illustrator.herokuapp.com/` (accessed July 15, 2013). *Social Media Illustrator* implements all the abovementioned steps and is a start-to-end solution tailored to both non-expert and expert users. Figure 10.8 shows the start screen of the application. The application has two tabs: *"Media Item Clusters"* and *"Media Gallery"*.

Media Item Extraction: In a first step, the user enters a set of search terms that are likely to reveal media items related to a given event. These search terms can be official event hashtags (*e.g.*, `#VSFashionShow` for the event described in section 8.3.6),


Social Media Illustrator

Media Item Clusters | Media Gallery

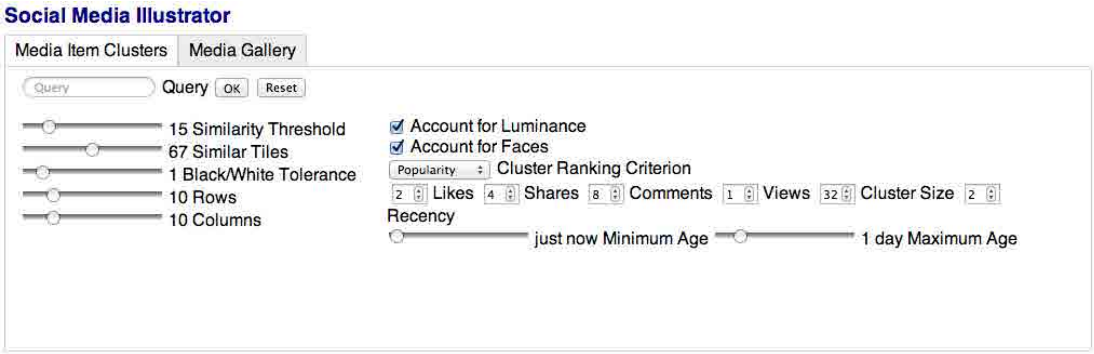
Play all videos | Mute all videos | Download Media Gallery | Loose order, varying size | Media Gallery Algorithm

18 Media Gallery Size | 1200 Media Gallery Width | 150 Media Item Height

People shouldn't be afraid of
their government.
Governments should be afraid of
their people.
-V

Figure 10.7: Media gallery in interactive mode, one media item centered and exclusively focused, all non-focused media items smoothly blurred and transitioned to a black-and-white version

but more commonly a combination of names of the involved actors, event names, event locations, or times [2, 3] like, for example, *Stanley Cup 2013*. Search results for each search term appear in the results panel in the lower part of the graphical user interface in form of a so-called search bundle. Search bundles are combined separate searches that maintain mappings to the individual original search terms, which can be enabled and disabled at will. Undesired media items can be removed from the result bundle by clicking a red cross that appears when hovering over the media item in question.



Social Media Illustrator

Media Item Clusters | Media Gallery

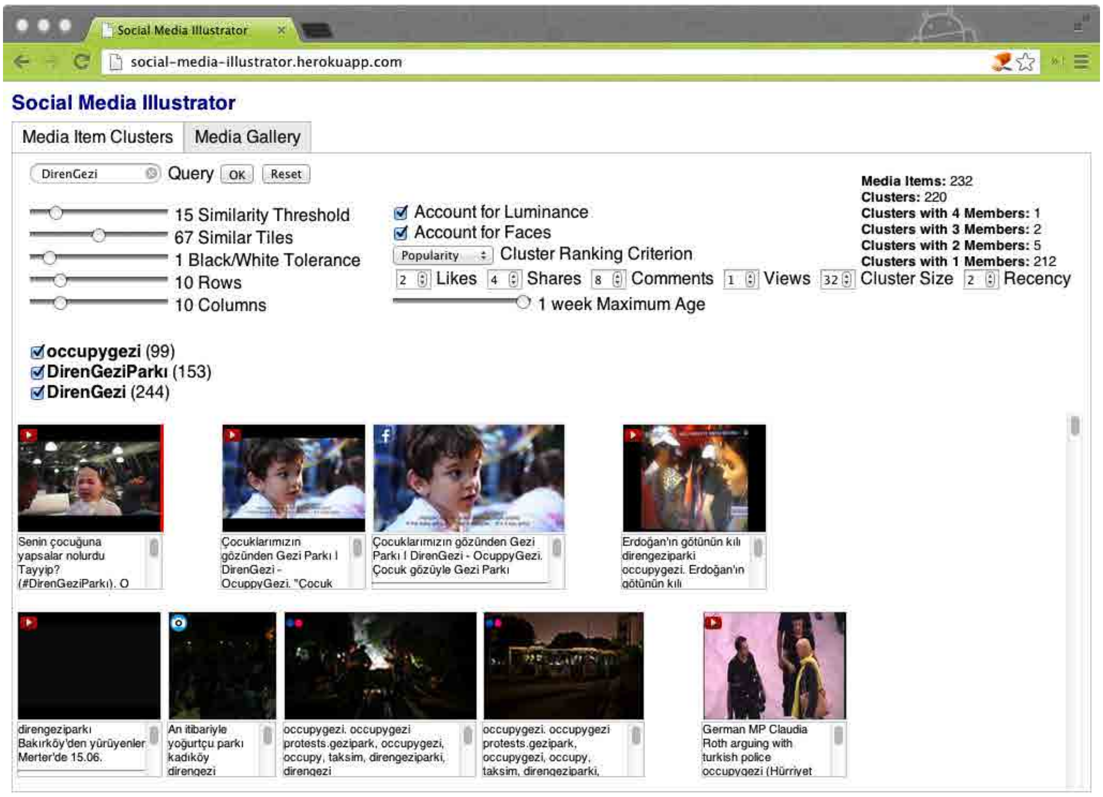
Query Query OK Reset

15 Similarity Threshold
67 Similar Tiles
1 Black/White Tolerance
10 Rows
10 Columns

☑ Account for Luminance
☑ Account for Faces
Popularity Cluster Ranking Criterion
2 Likes 4 Shares 8 Comments 1 Views 32 Cluster Size 2
Recency
just now Minimum Age 1 day Maximum Age

Figure 10.8: *Social Media Illustrator* start screen

Figure 10.9 shows *Social Media Illustrator* with extracted media items stemming from various social networks for three search terms.



Social Media Illustrator ×

social-media-illustrator.herokuapp.com

Social Media Illustrator

Media Item Clusters | Media Gallery

DirenGezi Query OK Reset

15 Similarity Threshold
67 Similar Tiles
1 Black/White Tolerance
10 Rows
10 Columns

☑ Account for Luminance
☑ Account for Faces
Popularity Cluster Ranking Criterion
2 Likes 4 Shares 8 Comments 1 Views 32 Cluster Size 2 Recency
1 week Maximum Age

Media Items: 232
Clusters: 220
Clusters with 4 Members: 1
Clusters with 3 Members: 2
Clusters with 2 Members: 5
Clusters with 1 Members: 212

☑ **occupygezi** (99)
☑ **DirenGeziParkı** (153)
☑ **DirenGezi** (244)

Senin çocuğuna yapsalar nolurdu Tayyip? (#DirenGeziParkı). O

Çocuklarımızın gözünden Gezi Parkı I DirenGezi - OcuppyGezi. "Çocuk

Çocuklarımızın gözünden Gezi Parkı I DirenGezi - OcuppyGezi. Çocuk gözüyle Gezi Parkı

Erdoğan'ın götünün kılı direngeziparki occupygezi. Erdoğan'ın götünün kılı

direngeziparkı Bakırköy'den yürüyenler Merter'de 15.06.

An itibariyle yoğurtçu parkı kadıköy direngezi

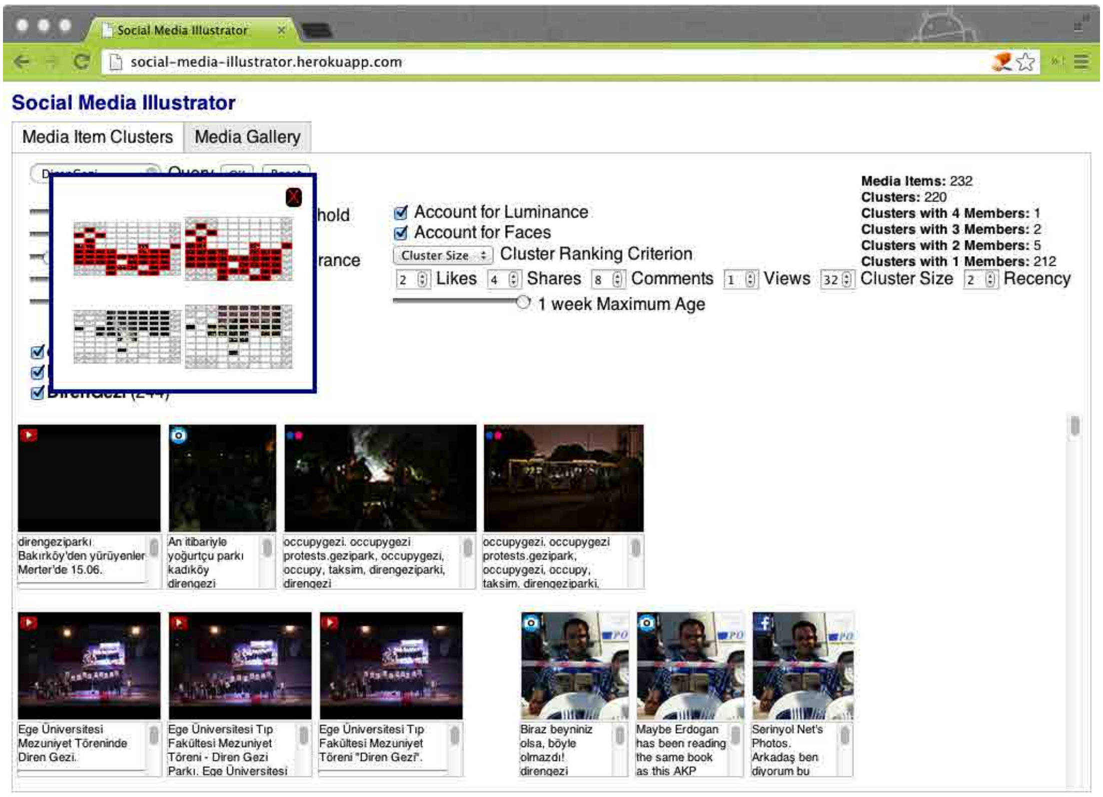
occupygezi. occupygezi protests.gezipark, occupygezi, occupy, taksim, direngeziparki, direngezi

occupygezi. occupygezi protests.gezipark, occupygezi, occupy, taksim, direngeziparki,

German MP Claudia Roth arguing with turkish police occupygezi (Hürriyet

Figure 10.9: Media item extraction with search bundles from three search terms

Media Item Deduplication: Via the parameter settings on the left side, fine-grained control over the deduplicating and matching algorithm is possible. The configurable options are described in subsection 8.5.3. Changes to any of the parameters are reflected in realtime in the results panel below, which facilitates finding the optimal settings for a given result set consisting of result bundles. Figure 10.10 shows the deduplication debug view (see subsection 8.5.3) that is accessible by right-clicking two media items.



☑ Account for Luminance
☑ Account for Faces
Cluster Size Cluster Ranking Criterion
2 Likes 4 Shares 8 Comments 1 Views 32 Cluster Size 2 Recency
1 week Maximum Age

Media Items: 232
Clusters: 220
Clusters with 4 Members: 1
Clusters with 3 Members: 2
Clusters with 2 Members: 5
Clusters with 1 Members: 212

direngeziparkı Bakırköy'den yürüyenler Merter'de 15.06.

An itibariyle yoğurtçu parkı kadıköy direngezi

occupygezi. occupygezi protests.gezipark, occupygezi, occupy, taksim, direngeziparki, direngezi

occupygezi. occupygezi protests.gezipark, occupygezi, occupy, taksim, direngeziparki,

Ege Üniversitesi Mezuniyet Töreninde Diren Gezi.

Ege Üniversitesi Tıp Fakültesi Mezuniyet Töreni - Diren Gezi Parkı. Ege Üniversitesi

Ege Üniversitesi Tıp Fakültesi Mezuniyet Töreni "Diren Gezi".

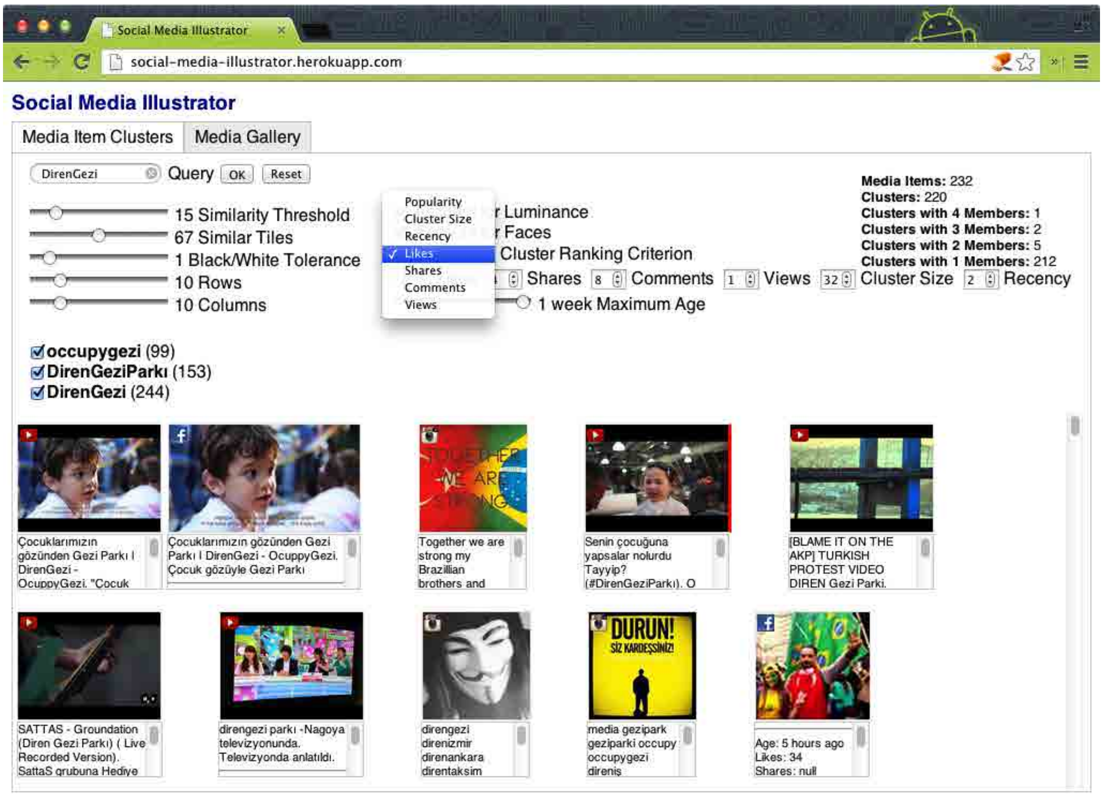
Biraz beyniniz olsa, böyle olmazdı! direngezi

Maybe Erdogan has been reading the same book as this AKP

Serinyol Net's Photos. Arkadaş ben diyorum bu

Figure 10.10: Media item deduplication debug view

Media Item Ranking: The parameter settings on the right side allow for interactively changing the ranking factors. The user can select the main ranking criterion like recency or popularity, and modify the weight factors for the different ranking features in the ranking formula, as detailed in subsection 9.4.2. Again all changes are reflected in realtime. A slider control allows for selecting the maximum and minimum age of the considered media items in order to restrict the result set to a given time period. Figure 10.11 shows the select box where the ranking formula can be selected.



Social Media Illustrator ×
social-media-illustrator.herokuapp.com

Social Media Illustrator

Media Item Clusters | Media Gallery

DirenGezi Query OK Reset

15 Similarity Threshold
67 Similar Tiles
1 Black/White Tolerance
10 Rows
10 Columns

Popularity
Cluster Size
Recency
✓ Likes
Shares
Comments
Views

r Luminance
r Faces
Cluster Ranking Criterion
4 Shares 8 Comments 1 Views 32 Cluster Size 2 Recency
1 week Maximum Age

Media Items: 232
Clusters: 220
Clusters with 4 Members: 1
Clusters with 3 Members: 2
Clusters with 2 Members: 5
Clusters with 1 Members: 212

☑ **occupygezi** (99)
☑ **DirenGeziParkı** (153)
☑ **DirenGezi** (244)

Çocuklarımızın gözünden Gezi Parkı I DirenGezi - OcuppyGezi. "Çocuk

Çocuklarımızın gözünden Gezi Parkı I DirenGezi - OcuppyGezi. Çocuk gözüyle Gezi Parkı

Together we are strong my Brazillian brothers and

Senin çocuğuna yapsalar nolurdu Tayyip? (#DirenGeziParkı). O

[BLAME IT ON THE AKP] TURKISH PROTEST VIDEO DIREN Gezi Parki.

SATTAS - Groundation (Diren Gezi Parkı) (Live Recorded Version). SattaS grubuna Hediye

direngezi parkı -Nagoya televizyonunda. Televizyonda anlatıldı.

direngezi direnizmir direnankara direntaksim

media gezipark geziparki occupy occupygezi direniş

Age: 5 hours ago
Likes: 34
Shares: null

Figure 10.11: Media item ranking with configurable ranking formula

Micropost Annotation: Once the user is happy with her selection and ranking of media items, the next step is micropost annotation, which—in contrast to the sequence suggested by the chapter order—only happens at this stage, based on only the final set of media items. This is due to the fact that calling the external named entity extraction and disambiguation services described in subsection 4.3.1 is very expensive and time-consuming, especially if an intermediate machine translation step is required. This step happens transparently in the background, while the media gallery is being compiled and is invisible to the user.

Media Item Compilation: The final step of media item compilation can be initiated by clicking on the *"Media Gallery"* tab. A configurable number of media items are compiled either in Loose Order, Varying Size (LOVS) style, or Strict Order, Equal Size (SOES), which was detailed in section 10.5. The number of considered media items, the width of the overall media gallery, and the width of individual media items can be customized, with the changes being reflected on-the-fly. Figure 10.12 shows a media gallery in the LOVS style. By clicking on one of the media items, the media gallery enters the interactive mode, as can be seen in Figure 10.13. Navigation from one media item to the next is possible via the arrow keys.

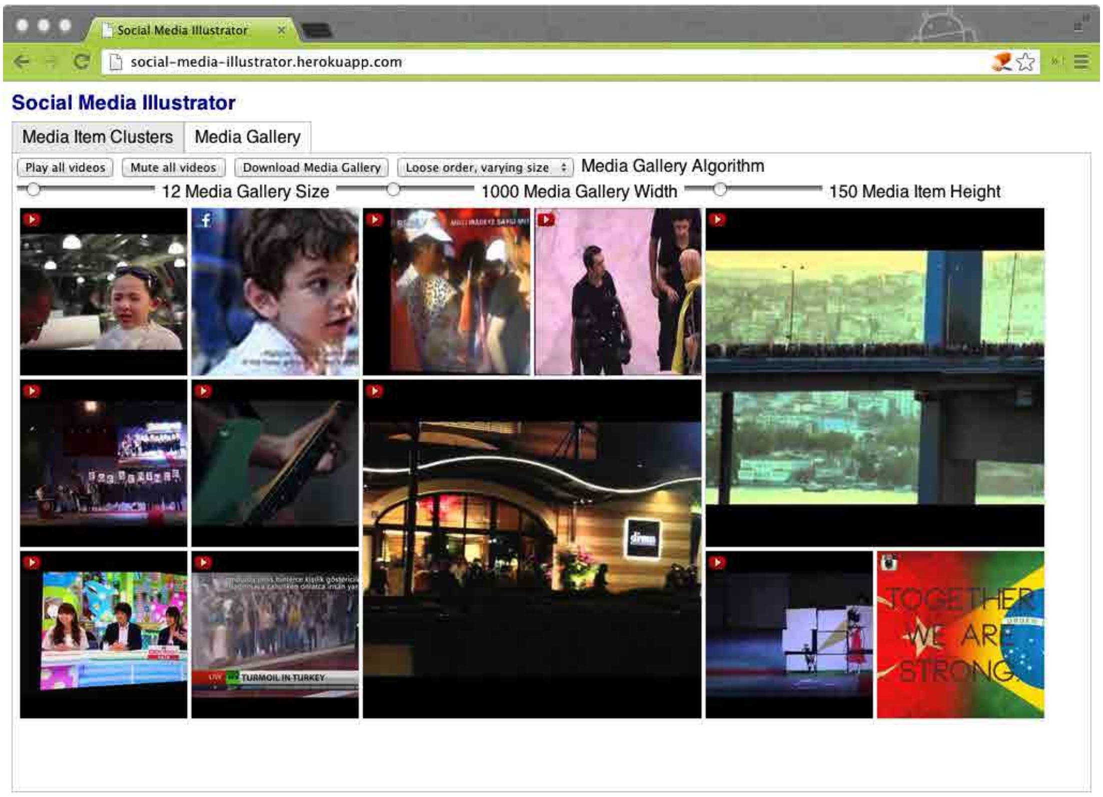
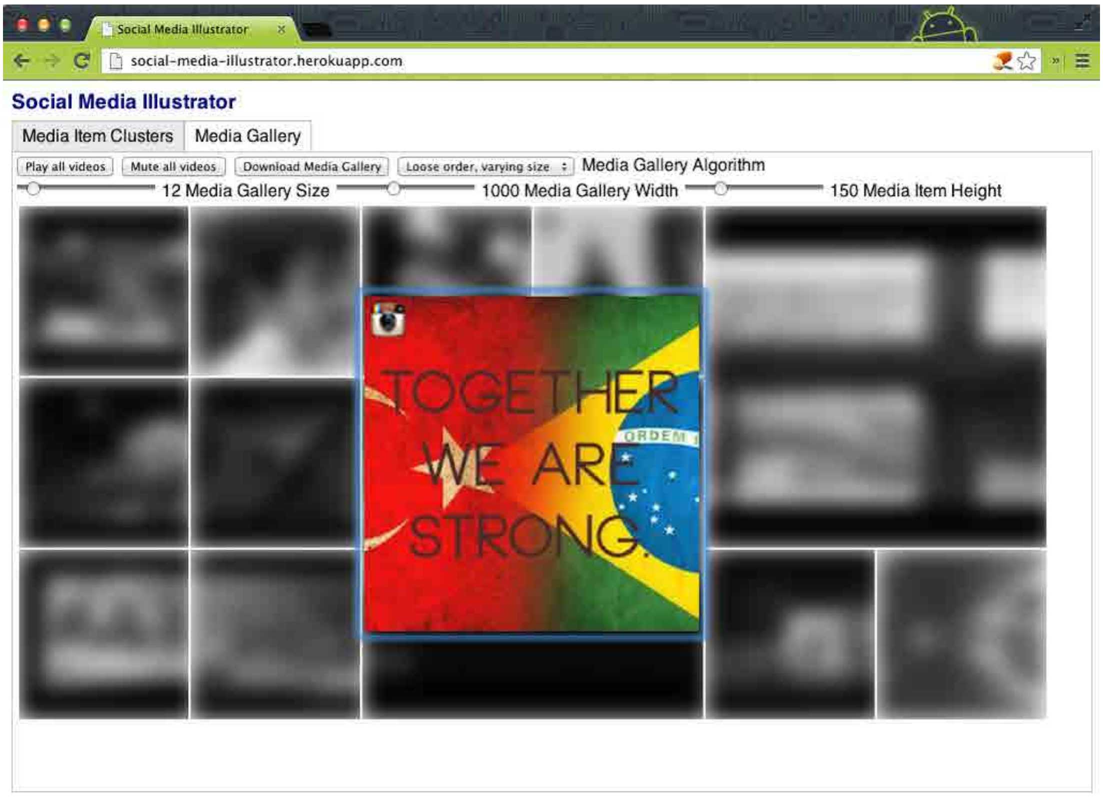


Figure 10.12: Media gallery generation based on the Loose Order, Varying Sizes style

10.8 Conclusions

In this chapter, we have defined factors that determine and influence media gallery aesthetics. After an overview of related work and a motivation, we have examined different algorithms for the automated generation of media galleries. While some of the described algorithms do not fulfill our requirements with regard to respecting the ranking-implied order, two of them—namely *LOVS* and *SOES*—do fulfill them and are in consequence considered. We have created an application that auto-generates the two media gallery styles *SOES* and *LOVS*, and evaluated users' perceived quality with



Loose order, varying size Media Gallery Algorithm

12 Media Gallery Size

1000 Media Gallery Width

150 Media Item Height

Figure 10.13: Media gallery in interactive mode

two separate surveys. The trend is that users prefer *LOVS*. Future work will be on evaluating more media gallery styles and advanced heuristics for media item cropping tailored to social network media items. As outlined earlier, such social media items do not necessarily share the same properties as common photos and videos. Social media cropping algorithms need to respect the characteristics of social media that were described in subsection 8.3.1.

We have learned that eyes need focal points to spot the needles in the media gallery haystack. Interactivity in form of visual *eye candy* as well as audial information are helpful factors to create the impression of a consistent *event summarization* that makes forget the fact that it was generated by combining potentially many social network users' contributions. Interactive media galleries help users process potentially large amounts of data in an entertaining way. Nevertheless—and besides all desired media gallery unity and consistency—we need to ensure that the individual social network user's contributions are still traceable in the combined media gallery. This is given for all use cases, offline and online. Concluding, with our publicly accessible application *Social Media Illustrator* we have contributed a valuable social media tool that allows non-expert users to create event-summarizing media galleries at ease.

Chapter Notes

This chapter is partly based on the following publication.

- Thomas Steiner and Christopher Chedeau. "To crop, or not to crop: compiling online media galleries". In: *Proceedings of the 22nd international conference on World Wide Web companion*. WWW '13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 201–202. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2487890`.

References

[1] Berke Atasoy and Jean-Bernard Martens. “STORIFY: A Tool to Assist Design Teams in Envisioning and Discussing User Experience”. In: *CHI ’11 Extended Abstracts on Human Factors in Computing Systems*. CHI EA ’11. Vancouver, BC, Canada: ACM, 2011, pp. 2263–2268. ISBN: 978-1-4503-0268-5. DOI: `10.1145/1979742.1979905`. URL: `http://doi.acm.org/10.1145/1979742.1979905`.

[2] Hila Becker, Dan Iter, Mor Naaman, and Luis Gravano. “Identifying Content for Planned Events Across Social Media Sites”. In: *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. WSDM ’12. ACM, 2012, pp. 533–542.

[3] Hila Becker, Mor Naaman, and Luis Gravano. “Learning Similarity Metrics for Event Identification in Social Media”. In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. WSDM ’10. ACM, 2010, pp. 291–300.

[4] Bert Bos, Tantek Çelik, Ian Hickson, and Håkon Wium Lie. *Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification*. Recommendation. W3C, 2011.

[5] Christopher Chedeau. *Image Layout Algorithm – Google+*. `http://blog.vjeux.com/2012/image/image-layout-algorithm-google.html`, accessed July 15, 2013. July 2012.

[6] Christopher Chedeau. *Image Layout Algorithm – 500px*. `http://blog.vjeux.com/2012/image/image-layout-algorithm-500px.html`, accessed July 15, 2013. Sept. 2012.

[7] Christopher Chedeau. *Image Layout Algorithm – Facebook*. `http://blog.vjeux.com/2012/image/image-layout-algorithm-facebook.html`, accessed July 15, 2013. Aug. 2012.

[8] Christopher Chedeau. *Image Layout Algorithm – Lightbox*. `http://blog.vjeux.com/2012/image/image-layout-algorithm-lightbox.html`, accessed July 15, 2013. July 2012.

[9] Christopher Chedeau. *Image Layout Algorithm – Lightbox Android*. `http://blog.vjeux.com/2012/image/image-layout-algorithm-lightbox-android.html`, accessed July 15, 2013. July 2012.

[10] Bo-Wei Chen, Jia-Ching Wang, and Jhing-Fa Wang. “A Novel Video Summarization Based on Mining the Story-Structure and Semantic Relations Among Concept Entities”. In: *Trans. Multi.* 11.2 (Feb. 2009), pp. 295–312. ISSN: 1520-9210.

[11] Smitashree Choudhury and John Breslin. “Extracting Semantic Entities and Events from Sports Tweets”. In: *Proceedings of the ESWC2011 Workshop on ‘Making Sense of Microposts’: Big things come in small packages.* Ed. by Matthew Rowe, Milan Stankovic, Aba-Sah Dadzie, and Mariann Hardey. 2011, pp. 22–32.

[12] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, et al. “The YouTube Video Recommendation System”. In: *Proceedings of the 4^{th} ACM Conference on Recommender Systems.* RecSys ’10. Barcelona, Spain: ACM, 2010, pp. 293–296.

[13] Manfred Del Fabro and Laszlo Böszörmenyi. “Summarization and Presentation of Real-Life Events Using Community-Contributed Content”. In: *Proceedings of the 18th International Conference on Advances in Multimedia Modeling.* MMM’12. Klagenfurt, Austria: Springer-Verlag, 2012, pp. 630–632. ISBN: 978-3-642-27354-4.

[14] J. Duddington. *eSpeak Text to Speech.* `http://espeak.sourceforge.net/`, accessed July 15, 2013. 2012.

[15] Kelly Fincham. “Review: Storify (2011)”. In: *Journal of Media Literacy Education* 3.1 (2011).

[16] Simon Fraser, Dean Jackson, Edward O’Connor, Dirk Schulze, Aryeh Gregor, David Hyatt, et al. *CSS Transforms.* Working Draft. `http://www.w3.org/TR/css3-transforms/`, accessed July 15, 2013. W3C, 2012.

[17] International Telecommunication Union, Telecommunication Standardization Sector. *ITU-T Recommendation P.800: Methods for Subjective Determination of Transmission Quality.* `http://www.itu.int/rec/T-REC-P.800-199608-I/en`, accessed July 15, 2013. Aug. 1998.

[18] Dean Jackson, David Hyatt, Chris Marrin, and L. David Baron. *CSS Transitions Module Level 3.* Working Draft. `http://www.w3.org/TR/css3-transitions/`, accessed July 15, 2013. W3C, 2013.

[19] Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer. “Combining Audio-based Similarity with Web-based Data to Accelerate Automatic Music Playlist Generation”. In: *Proceedings of the 8^{th} ACM International Workshop on Multimedia Information Retrieval.* MIR ’06. Santa Barbara, California, USA: ACM, 2006, pp. 147–154.

[20] Baoxin Li and M. Ibrahim Sezan. “Event Detection and Summarization in American Football Broadcast Video”. In: *Storage and Retrieval for Media Databases.* Ed. by Minerva M. Yeung, Chung-Sheng Li, and Rainer Lienhart. Vol. 4676. SPIE Proceedings. SPIE, 2002, pp. 202–213.

[21] Baoxin Li and M. Ibrahim Sezan. “Event Detection and Summarization in Sports Video”. In: *Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL’01).* CBAIVL ’01. Washington, DC, USA: IEEE Computer Society, 2001.

[22] Pere Obrador, Michele Saad, Poonam Suryanarayan, and Nuria Oliver. “Towards Category-Based Aesthetic Models of Photographs”. In: *Proceedings of the 18th International Conference on Advances in Multimedia Modeling – Volume Part I (MMM 2012).* 2012, pp. 63–76.

[23] Dan R. Olsen and Brandon Moon. “Video Summarization Based on User Interaction”. In: *Proceedings of the 9th International Interactive Conference on Interactive Television.* EuroITV ’11. Lisbon, Portugal: ACM, 2011, pp. 115–122. ISBN: 978-1-4503-0602-7.

[24] Philipp Sandhaus, Mohammad Rabbath, and Susanne Boll. “Employing Aesthetic Principles for Automatic Photo Book Layout”. In: *Proceedings of the 17th International Conference on Advances in Multimedia Modeling – Volume Part I (MMM 2011).* 2011, pp. 84–95.

[25] Thomas Steiner and Christopher Chedeau. “To crop, or not to crop: compiling online media galleries”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 201–202. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2487890`.

[26] Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, and Rik Van de Walle. “Defining Aesthetic Principles for Automatic Media Gallery Layout for Visual and Audial Event Summarization based on Social Networks”. In: *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on.* July 2012, pp. 27–28. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/defining-aesthetic-principles-for-automatic-media-gallery-layout-qomex2012.pdf`.

[27] Bongwon Suh, Haibin Ling, Benjamin B. Bederson, and David W. Jacobs. "Automatic Thumbnail Cropping and its Effectiveness". In: *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*. Vancouver, Canada, 2003. ISBN: 1-58113-636-6.

11

Conclusions and Future Work

11.1 Conclusions

In this thesis, we have tackled the challenge of event summarization from a multimedia angle by leveraging social networks. In the first part of the thesis, we have presented the Semantic Web, its technologies, and its applications like Linked Data. In continuation, we have discussed social networks and their features and have classified them based on their level of media item support. We have applied semantic methods for limiting ambiguity that is intrinsic to language and that especially affects short textual microposts, which oftentimes lack context. As this task required the orchestration of several services in combination, we have shown how provenance information can be preserved to make the involved steps traceable. We have shown how Wikipedia edits that we capture in realtime can be clustered by language for the task of detecting breaking news event candidates. An accompanying application called *Wikipedia Live Monitor* was developed and publicly released in the same context. We have examined how media items can be extracted from various social networks and how the different underlying social network messages and interactions can be aligned by an abstraction layer. Further, we have introduced an algorithm to detect camera shots in streaming video and annotated them semantically with media fragments URIs. This task is a required step for video deduplication, which we have tackled together with photo deduplication. We have determined social-network-specific reasons for the occurrence of exact-duplicate and near-duplicate content on social networks. Based on these findings, we have developed and algorithm that is tailored to detect such occurrences of exact-duplicate and near-duplicate content.

This algorithm has multiple matching conditions, which makes it hard for a non-expert user to determine why or why not media items have been matched. Driven by this observation, we have researched if speech synthesis together with dynamic media fragment creation can help non-expert users understand the algorithm's output, which in consequence allowed us to significantly improve it. We have introduced a ranking formula for social media clusters that, based on a social interaction abstraction layer, ranks clusters by popularity and from each cluster selects the visually most appealing cluster representative media item. Afterwards, we have discussed and evaluated several media gallery compilation algorithms that fulfill media gallery aesthetics criteria, which we have defined. Finally, we have created an interactive, speech-synthesis-supported visualization format on top of our media gallery compilation algorithms.

Based on a very open event definition from WordNet—*"something that happens at a given place and time"*—we have identified the need to visually and audially summarize events of personal or public interest. The increasingly important role of first-hand eye-witness social media data at recent events like the Boston Marathon bombings 2013[1] or the Occupy Gezi movement 2013 in Turkey[2] drastically confirm the observation that there is a strong demand for social-network-based event summarization. Since the beginning, we have openly shared our progress in form of screenshots for the different tasks of the application (`http://twitpic.com/tag/TomsPhD`, accessed July 15, 2013) and have also documented our progress in general (`http://tomayac.com/tweets/search?q=%23TomsPhD`, accessed July 15, 2013). This has allowed us to get early feedback on ideas and visualizations already at the design phase. The way interactive mode works in media galleries that was detailed in section 10.6 is a result of this early-stage feedback loop. An earlier iteration of interactive mode is documented in the screenshot available at `http://twitpic.com/c94j02` (accessed July 15, 2013), where the background media items were already blurred, but not yet transformed into black-and-white. Especially with media items of type photo, through steadily monitoring (at time of writing) current events, we have learned that social media galleries need to be specifically tailored to handle all sorts of photo formats, as screenshots in non-standard aspect ratios are more common than expected. Some examples are documented in a screenshot available at `http://twitpic.com/bvjmgc`

[1] `http://en.wikipedia.org/wiki/Boston_Marathon_bombings`, accessed July 15, 2013

[2] `http://en.wikipedia.org/wiki/2013_Taksim_Gezi_Park_protests`, accessed July 15, 2013

(accessed July 15, 2013). We were also surprised by the diversity of the media items related to a given event. Where media galleries of traditional sources like news agencies feature strictly event-related media items, media galleries created by our approach also feature media items of humorous nature like memes,[1] photo montages, or parody images. On the one hand, this can be attributed to the less formal requirements regarding the political correctness of media galleries created through our approach. On the other hand, it can be traced back to the insouciant naivety of non-professional photographers and cinematographers that are responsible for the majority of social network media items on the other. A good example is documented in the screenshot available at `http://twitpic.com/bvwz7x` (accessed July 15, 2013).

Our initial research question was the following. *"Can user-customizable media galleries that summarize given events be created solely based on textual and multimedia data from social networks?"* Concluding, the answer is clearly *yes*. Media galleries based on social network multimedia data are faster to generate, more authentic and concise, more flexible and customizable, more comprehensive and diverse, and finally oftentimes more interesting to consume than traditional media galleries. For any serious use case, human final inspection will always be required, even in the long-term, as we will outline in the next subsection. With our research and the applications *Social Media Illustrator* and *Wikipedia Live Monitor*, we have contributed valuable tools, methods, design ideas, and algorithms to facilitate and automate the otherwise tedious task of manually generating media galleries. This allows users of these applications to focus on tasks where humans excel, like, *e.g.*, interpreting the effect of events on society, putting events in relation to each other, or identifying reasons that caused them.

11.2 Future Work

11.2.1 Media Item Verification

An aspect that we have left aside so far is the *verification* of the *credibility* and *authenticity* of both sources and media items themselves that get shared on social networks. The growing importance and usage of truly first-hand eyewitness social media in traditional news media—or even social media being *the* actual news, as to some extent

[1] An Internet meme is an idea, style, or action that spreads, often as mimicry, from person to person via the Internet, as with imitating the concept

it was the case with the Boston Marathon bombings, makes social networks also an increasingly popular focus of *intentionally false information.* The distribution of false information in form of media items can have all sorts of motivations, ranging from (sometimes fun) hoaxes[1] to political propaganda to simple human errors. The list is far from being complete. Occurrences of false information can include media items stemming from unrelated events being published as event-related, manipulation of photos or videos to modify, add, or remove persons or objects in media items, or wrong statements in the accompanying microposts. Manual approaches to recognize false information can include carefully checking the account publishing history of the originating source (does the social network user seem legit?), comparison of depicted scenes with independent media, *e.g.*, satellite imagery or street panoramas (does the depicted scene look the same elsewhere?), verifying weather conditions (were the known weather conditions at the time when the event happened the same as the depicted ones?), analyzing media items for known patterns of pixel manipulation (are traces of, for example, image editing software usage visible?), or finally, verifying media item metadata (do the data in the Exif block look valid?). Research on media item verification is ongoing, an example is [3] by Gupta *et al.* The first commercial companies are beginning to offer media item verification as a service, *e.g.*, Storyful (`http://storyful.com/`). The Managing Editor of the company, Markham Nolan, has covered the topic of media item verification in a TED Talk, which is available online.[2] A future research direction can be to automate this entirely manual process, albeit, having a human in the loop will—all potential automation aside—still be required and desirable for many use cases.

11.2.2 Evaluation of Subjective Data

Examples of Subjective Data: In many of the tasks from the previous chapters we had to deal with subjective data and how to evaluate it. In contrast to *objectivity* (where people see things from a standpoint *free* from human perception and its influences, human cultural interventions, past experiences, and expectation of the result), the contrasting term *subjectivity* is used to refer to the condition of being a *subject*,

[1] A hoax is a deliberately fabricated falsehood made to masquerade as truth

[2] `http://www.ted.com/talks/markham_nolan_how_to_separate_fact_and_fiction_online.html`, accessed July 15, 2013

under the influence of the subject's perspective, experiences, feelings, beliefs, and desires [4]. In the following, we list some of the subjective things that were evaluated in this thesis. As a first example, there are media gallery aesthetics and media gallery usefulness (Chapter 10), where the subjective decision is whether or not generated media galleries are aesthetically pleasing and at the same time useful for getting an understanding of the summarized event. Further examples are the subjective decisions on a media item set's ranking (Chapter 9), its clustering and deduplication (Chapter 8, Chapter 7), and the set itself (Chapter 6). In addition to that, there is also event detection, where the subjective decision is whether or not a given detected breaking news event candidate is indeed newsworthy (Chapter 5) and for whom. Finally, there are the extracted and disambiguated named entities from the accompanying microposts for media items (Chapter 4).

Subjective Data Evaluation Strategies: Common strategies for the evaluation of subjective data were examined by Brabb and Morrison in [1]. In the multimedia context, the main evaluation strategies are the use of Likert scales [6] and the Mean Opinion Score [5], which we have chosen for our evaluation purposes, as we were mainly interested in the perceived quality of our tasks. What all these evaluation strategies have in common is that they create a potentially artificial test feeling or lab environment situation, where users tend to not act naturally.

Multi-Armed Bandits Experiments: One-armed bandits are slot machines with potentially varying expected payout. Multi-armed bandit experiments are hypothetical experiments where, when faced with several one-armed bandits, the objective is to determine the most profitable one. The compromise or tension with such experiments is to greedily decide on bandits that have performed well in the past and taking the risk of trying new ones. Highly developed mathematical models exist [8] to optimize this problem. Compared to more classical A/B tests, where two variants are tested against each other, multi-armed bandit experiments are statistically just as valid and can oftentimes return results earlier, as already during the experiment more focus is gradually put on well-performing variants.

In the online retail industry, multi-armed bandit experiments have found broad adoption as they are easy to set up and efficient in finding actionable results. They are used extensively, *e.g.*, for the optimization of conversions for things like online purchases, newsletter sign-ups, or click-throughs. Typical experiment factors are heading texts, button colors and shapes, as well as page layout variants. Multi-armed bandits experiments are in consequence standard features of common off-the-shelf Web analytics software like, for example, Google Analytics (`http://google.com/analytics`).

Our hypothesis is that multi-armed bandit experiments can be used to evaluate the sort of subjective data we generate with our application *Social Media Illustrator*, given that we properly define our optimization criteria. Unlike with, *e.g.*, online purchases, where the optimization criterion are conversions,[1] with media galleries there is no direct optimization criterion. However, our assumption is that we can use indirect criteria like interaction with the media gallery as outlined in section 10.6, the rationale being that if a media gallery is interesting, the user will interact with it. Common Web analytics software is capable of tracking mouse and keyboard events that occur when users interact with Web pages. By exploiting this fact, we can attach event listeners to media galleries and report these events to the Web analytics software running on a remote server. By varying media gallery styles and parameters like the width, number of contained items, item size, *etc.*, we can then over time determine promising candidates. The proposed evaluation approach is less expensive and more scalable than user studies, albeit user studies may still outperform the approach with regard to discovering aspects that were not part of a multi-armed bandit experiment and thus never tested, but that a study participant may have noted in a free-form question. In the long-term, the proposed approach can thus be the seed for *more targeted user studies.*

11.2.3 Application Domains

Embedded Media Galleries: A final future research direction is finding new application domains. With our examples so far, we have mainly focused on the (online) journalist use case. We envision interactive media galleries taking the place of static photo galleries (Figure 10.5) or embedded videos on online editions of news websites or also Web portals.

[1] The amount of people who do not just put items in the virtual shopping cart, but who then proceed to and successfully complete the checkout process.

Data Journalism: Data journalism [2] is a form of journalism that reflects the increased role of numerical data in the production and distribution of information in the digital era. It touches on the fields of design, computer science, and statistics. Interactive media galleries and the cross-network search capabilities enabled by our application *Social Media Illustrator* can greatly facilitate the data journalism task of researching news stories and exploring multimedia data.

Event Pages of Social Networks: Some social networks like Facebook or Google+ offer their users the creation of events where event-related media items can be manually or automatically uploaded when event attendees *check in* to an event. Naturally, interactive media galleries embedded on social network sites themselves will only feature media items from the social network in question and not include foreign ones.

Disaster Response: Disasters like earthquakes, floods, plane crashes, *etc.* cause great damage or even loss of lives. Disaster response includes measures to mitigate the effects of a disastrous event in order to avoid further loss of lives or property. Social networks and especially social media play an increasing role in disaster response [9, 10]. Our work has already sparked initial interest [7] in the disaster response community. We will focus future work primarily on this aspect.

11.2.4 Commercial Activity in Social-Network-Based Event Summarization

The research fields of event summarization and event archiving based on social network multimedia data have resulted in interesting business creations in recent months. Albeit similarities to our work exist, there are still many differences in the details.

Mahaya: The company Mahaya has launched a beta-version of a commercial automatic event archiving tool called Seen (`http://beta.seen.co/`), which, based on manually entered event metadata like event name, location, and dates, uses a necessarily provided Twitter hashtag to create a complete and permanent archive of all event-related tweets, media items, and slide decks. Based on term frequency and co-occurrence analyses, the event is split into subevents and each subevent's hot topics are tried to be detected. The application's main data source is Twitter, links to certain media hosting platforms are followed. At time of writing, Seen does not yet deduplicate and cluster similar media items, albeit the tool is being actively worked on. A screenshot of Seen can be found in Figure 11.1.

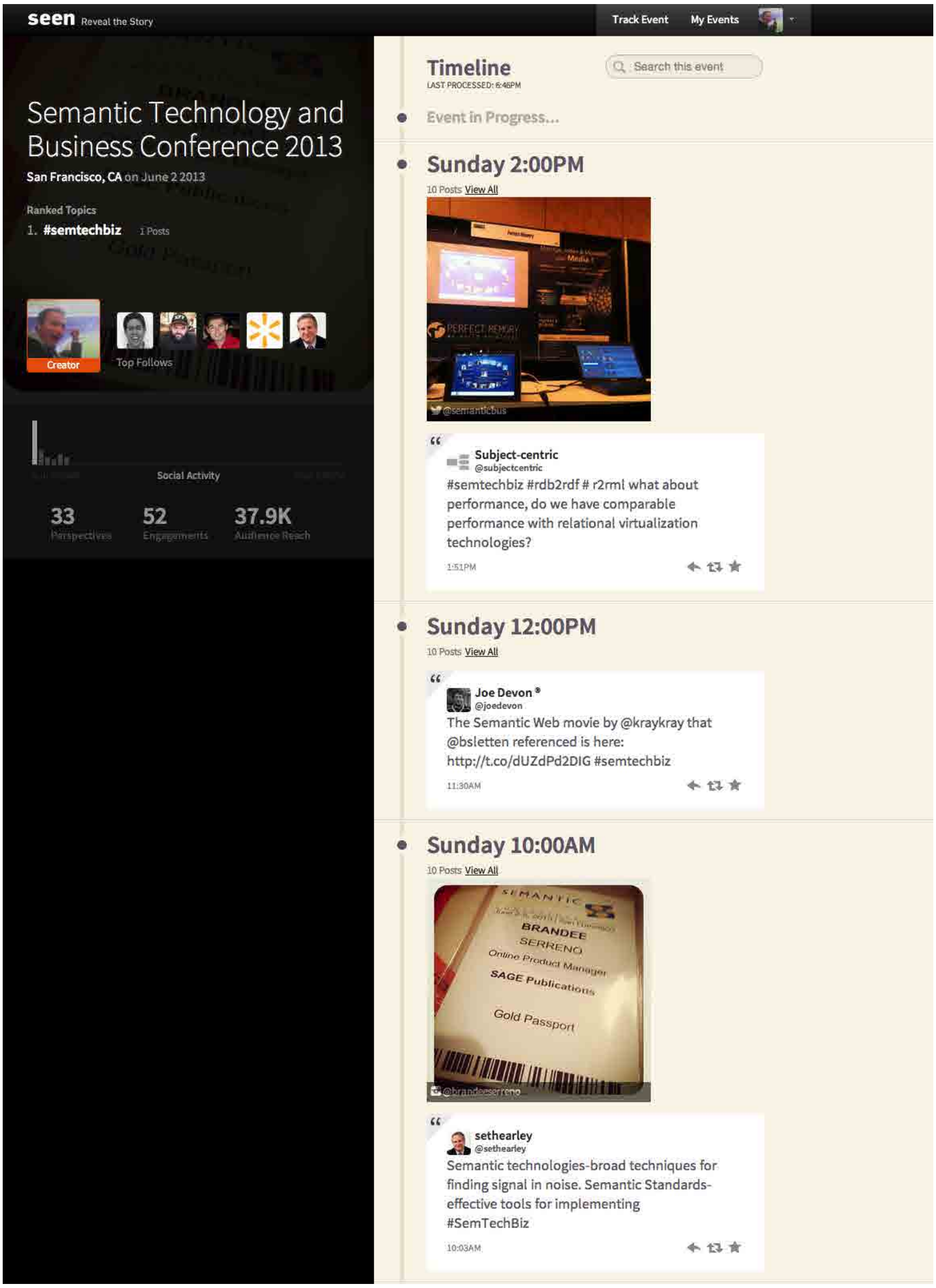
Eventifier: Eventifier `http://eventifier.co/`) is a commercial tool that facilitates the automated permanent archiving of events in form of event-related photos, videos, tweets, slide decks, and event contributors. Similar to Mahaya's product Seen, the application's main data source is Twitter. The manually entered official event hashtag and potentially existing official Twitter account serve to encounter event-related content. At time of writing, Eventifier does not yet deduplicate and cluster similar media items, however, this feature is said to be implemented. A screenshot of Eventifier can be seen in Figure 11.2.

Storify: The commercial tool Storify (`http://storify.com/`) allows for the manual compilation of event-related media items, articles, and microposts to generated permanently available stories that—depending on the level of human curation—can efficiently summarize an event. Storify does not deduplicate and cluster similar media items. A screenshot of Storify can be seen in Figure 11.3.

Media Finder: Media Finder (`http://mediafinder.eurecom.fr/`) is an academic non-commercial tool that has advanced named entity centric clustering capabilities based on extracted named entities in microposts. Media items can be clustered by topic, named entity, named entity type, and micropost instance. We have contributed the application's media extraction component, in consequence the covered social networks are exactly as described in Chapter 6. A screenshot of Media Finder can be seen in Figure 11.4.

11.2.5 Comparison of Tools

In the previous paragraphs, we have characterized commercial and non-commercial academic tools for the tasks of event summarization and event archiving. Table 11.1 shows how these tools compare against our own application *Social Media Illustrator*, which for reference is depicted again in Figure 11.5. What sets our application apart are its interactive media galleries that, together with speech synthesis as outlined in



Semantic Technology and Business Conference 2013

San Francisco, CA on June 2 2013

Ranked Topics

1. #semtechbiz 1 Posts

Timeline

LAST PROCESSED: 6:46PM

Event in Progress...

Sunday 2:00PM

10 Posts View All

Subject-centric
@subjectcentric
#semtechbiz #rdb2rdf # r2rml what about performance, do we have comparable performance with relational virtualization technologies?
1:51PM

Sunday 12:00PM

10 Posts View All

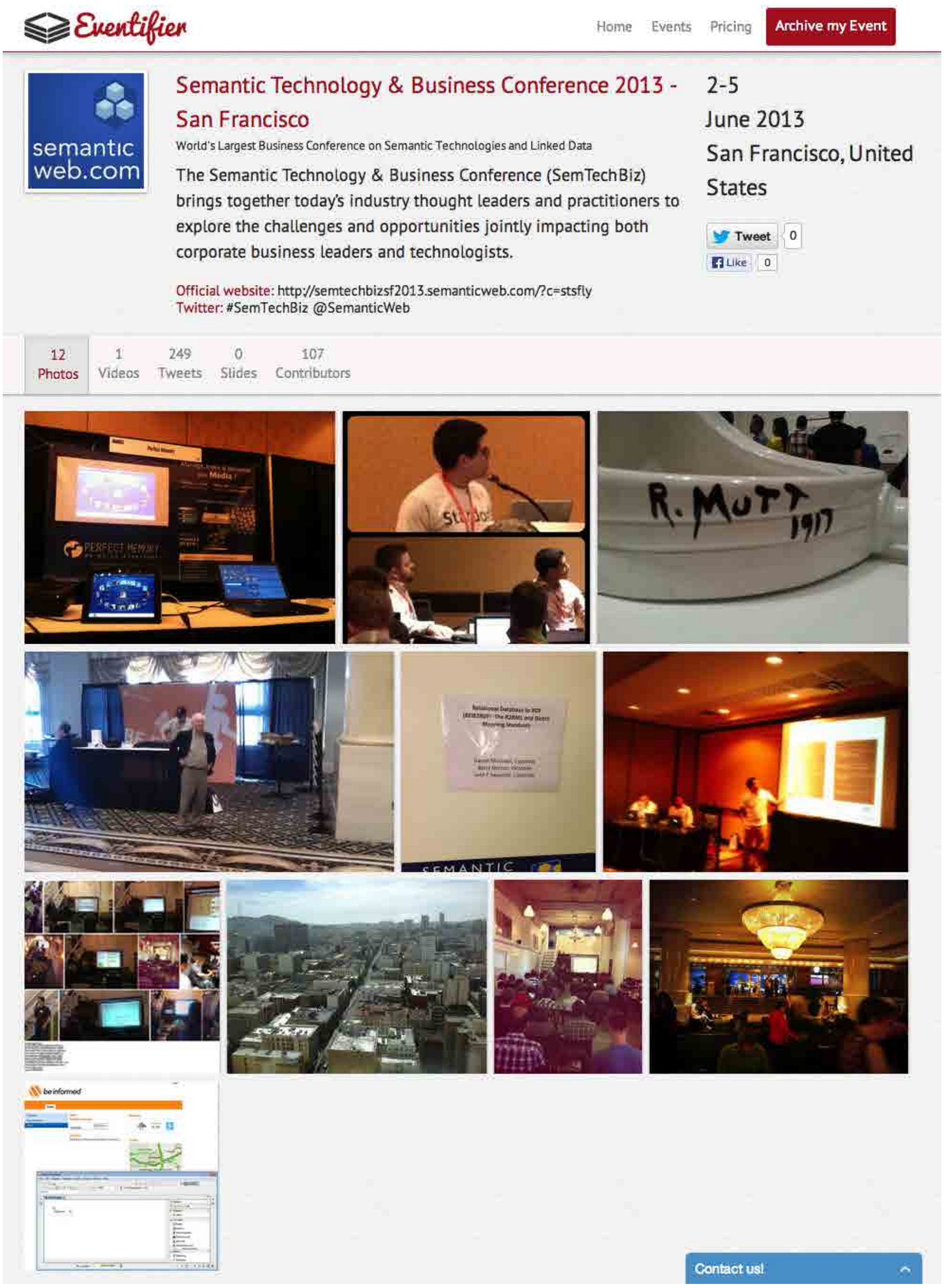
Joe Devon
@joedevon
The Semantic Web movie by @kraykray that @bsletten referenced is here: http://t.co/dUZdPd2DIG #semtechbiz
11:30AM

Sunday 10:00AM

10 Posts View All

sethearley
@sethearley
Semantic technologies-broad techniques for finding signal in noise. Semantic Standards-effective tools for implementing #SemTechBiz
10:03AM

Figure 11.1: Mahaya's commercial automatic event archiving tool called Seen (`http://beta.seen.co/`)



Semantic Technology & Business Conference 2013 - San Francisco

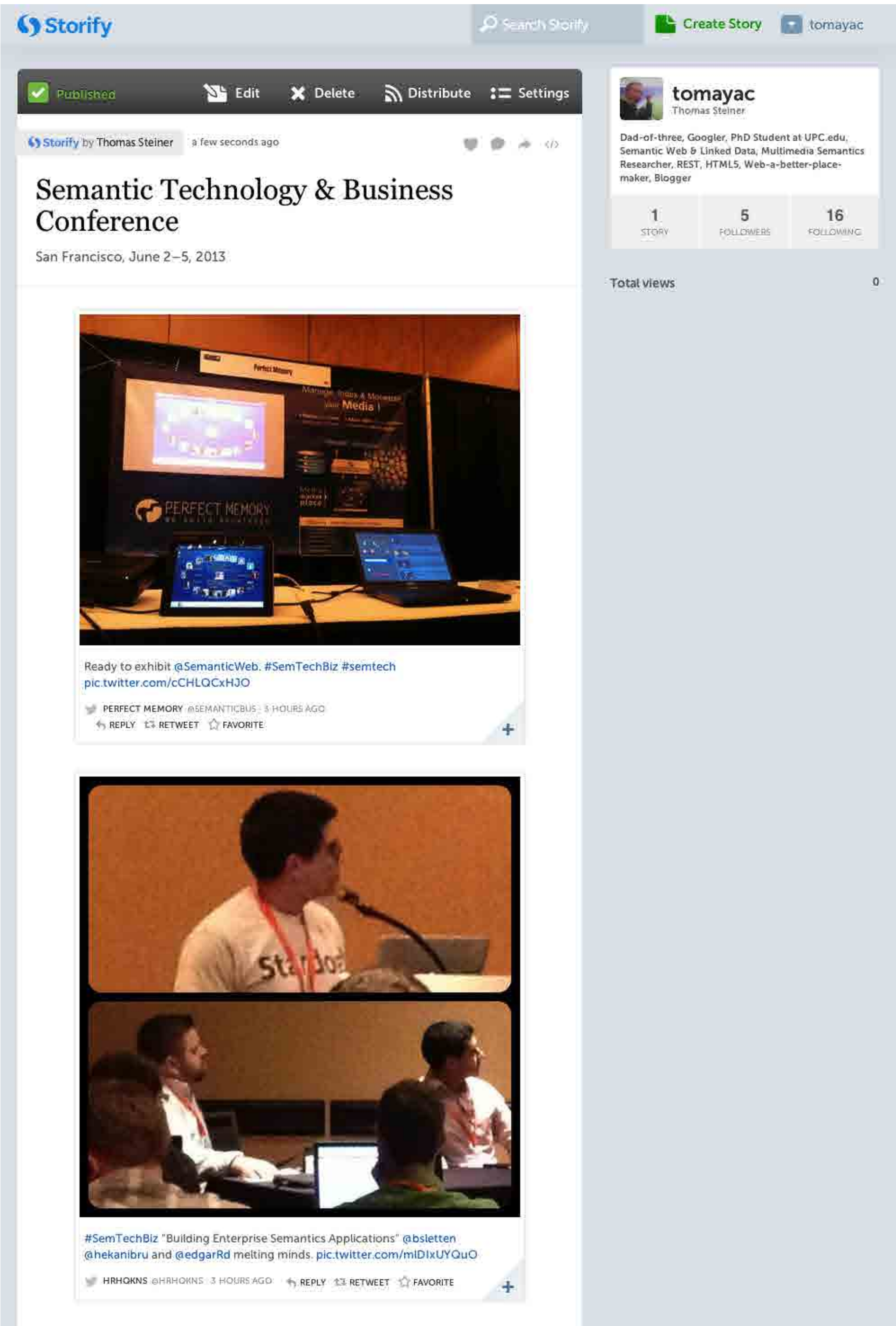
World's Largest Business Conference on Semantic Technologies and Linked Data

The Semantic Technology & Business Conference (SemTechBiz) brings together today's industry thought leaders and practitioners to explore the challenges and opportunities jointly impacting both corporate business leaders and technologists.

Official website: http://semtechbizsf2013.semanticweb.com/?c=stsfly
Twitter: #SemTechBiz @SemanticWeb

2-5 June 2013
San Francisco, United States

Figure 11.2: Automated Twitter-centered commercial event archiving tool Eventifier (`http://eventifier.co/`)



tomayac
Thomas Steiner

Dad-of-three, Googler, PhD Student at UPC.edu, Semantic Web & Linked Data, Multimedia Semantics Researcher, REST, HTML5, Web-a-better-place-maker, Blogger

1	5	16
STORY	FOLLOWERS	FOLLOWING

Total views 0

Semantic Technology & Business Conference

San Francisco, June 2–5, 2013

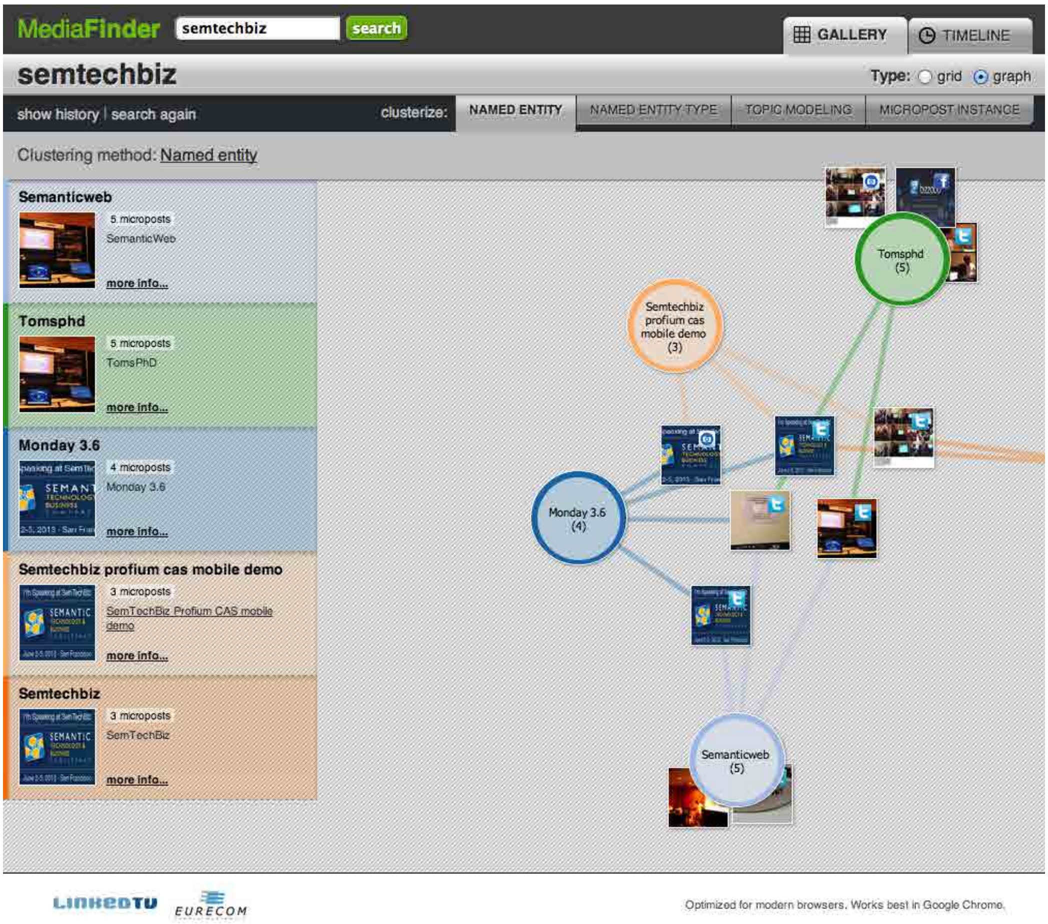
Ready to exhibit @SemanticWeb. #SemTechBiz #semtech pic.twitter.com/cCHLQCxHJO

PERFECT MEMORY @SEMANTICBUS · 3 HOURS AGO

#SemTechBiz "Building Enterprise Semantics Applications" @bsletten @hekanibru and @edgarRd melting minds. pic.twitter.com/mlDIxUYQuO

HRHQKNS @HRHQKNS · 3 HOURS AGO

Figure 11.3: Manually assisted multi-network commercial event archiving tool Storify (`http://storify.com/`)



MediaFinder semtechbiz search

semtechbiz

Type: ○ grid ● graph

show history | search again

clusterize: NAMED ENTITY | NAMED ENTITY TYPE | TOPIC MODELING | MICROPOST INSTANCE

Clustering method: Named entity

Semanticweb
5 microposts
SemanticWeb
more info...

Tomsphd
5 microposts
TomsPhD
more info...

Monday 3.6
4 microposts
Monday 3.6
more info...

Semtechbiz profium cas mobile demo
3 microposts
SemTechBiz Profium CAS mobile demo
more info...

Semtechbiz
3 microposts
SemTechBiz
more info...

Optimized for modern browsers. Works best in Google Chrome.

Figure 11.4: Academic multi-network event summarization tool Media Finder (`http://mediafinder.eurecom.fr/`)

section 10.6 allow for novel kinds of experiences when it comes to user-customizable visual *and* audial event summary consumption in both passive and active user-controlled ways with full provenance-aware download support.

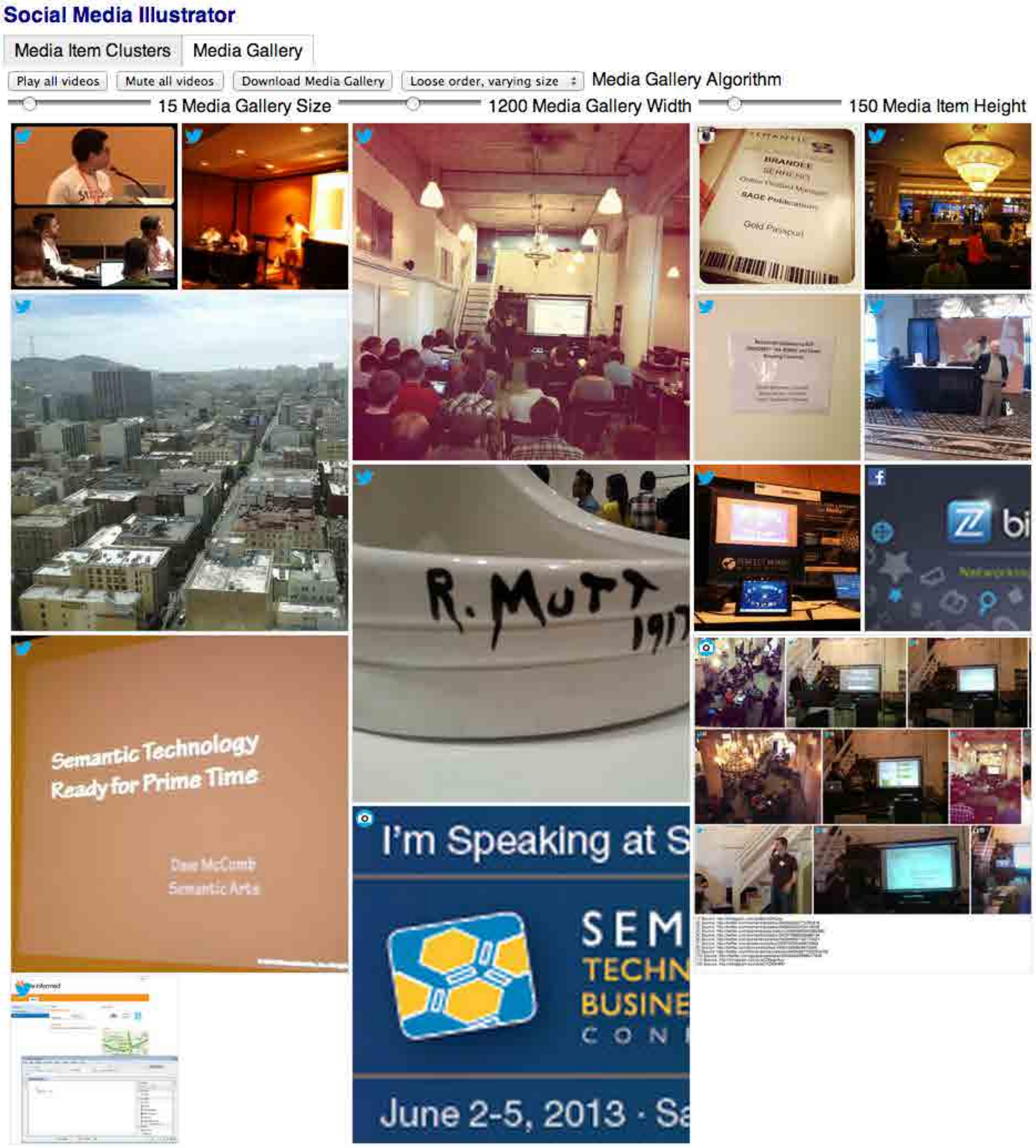
11.3 Closing Words

In this thesis, we have touched on multiple areas of research. Some of the encountered problems and challenges, for example, named entity extraction and disambiguation for short and oftentimes sloppily-authored microposts or also video deduplication, certainly deserve a thesis of their own. We have opted for a pragmatic approach in such cases and have not shied back from either using third-party tools or working with approximations or heuristics, which work well enough for our use case. From the beginning, we have envisioned an application that would facilitate the tedious work of compiling media galleries manually. This application, *Social Media Illustrator*, forms part of the deliverables of the thesis. We have separated the task of building this application in several actionable steps and have contributed scientific publications for each of them. The chapters of this thesis follow these steps loosely. As a reminder, the concrete steps were the following: (i) micropost annotation, (ii) event detection, (iii) media item extraction, (iv) media item deduplication, (v) media item ranking, and (vi) media item compilation. At the end of this thesis, we are now in the position to first accurately detect events and second, to visually and audially summarize them in an optionally fully-automated or semi-automated manner, so the circle has been closed.

We were ourselves surprised by the broad range of possible future use cases, ranging from end users reviving the atmosphere of a concert, to data journalists researching political events of potentially global interest, to finally disaster relief workers coordinating their efforts based on information derived from our applications. We are excited to improve, extend, and adapt *Social Media Illustrator* and *Wikipedia Live Monitor* in the future with concrete ongoing research opportunities that were outlined earlier in this chapter. This thesis marks the end of this doctorate, but it certainly does not mark the end of this work.

"The Software shall be used for Good, not Evil."[1] —Douglas Crockford

[1] `https://raw.github.com/douglascrockford/JSLint/master/jslint.js`, accessed July 15, 2013



Loose order, varying size Media Gallery Algorithm

15 Media Gallery Size

1200 Media Gallery Width

150 Media Item Height

Figure 11.5: Our own academic event summarization tool *Social Media Illustrator* (`http://social-media-illustrator.herokuapp.com/`)

Tool	**Seen**	**Eventifier**	**Storify**	**Media Finder**	**Social Media Illustrator**
Data source	Twitter	Twitter	Multiple	Multiple	Multiple
Main task	Archiving	Archiving	Summarization	Summarization	Summarization
Operation mode	Automated	Automated	Manual	Semi-automated	Semi-automated
Linkable	Yes	Yes	Yes	Yes	No
Downloadable	No	No	No	No	Yes
Customizable	No	No	Yes	Yes	Yes
Interactive	No	No	No	No	Yes
Commercial	Yes	Yes	Yes	No	No

Table 11.1: Comparison of commercial and non-commercial event archiving and summarization tools

References

[1] George J Brabb and Edmund D Morrison. “The evaluation of subjective information”. In: *Journal of Marketing Research* (1964), pp. 40–44.

[2] J. Gray, L. Chambers, and L. Bounegru. *The Data Journalism Handbook*. O’Reilly Media, 2012. ISBN: 9781449330026.

[3] Aditi Gupta, Hemank Lamba, Ponnurangam Kumaraguru, and Anupam Joshi. “Faking Sandy: Characterizing and Identifying Fake Images on Twitter During Hurricane Sandy”. In: *Proceedings of the 22nd International Conference Companion on World Wide Web*. WWW ’13. Rio de Janeiro, Brazil, May 2013, pp. 729–736.

[4] T. Honderich. *The Oxford Companion to Philosophy*. Oxc Series. Nhà xuất bản Văn hóa thông tin, 2005. ISBN: 9780199264797.

[5] International Telecommunication Union, Telecommunication Standardization Sector. *ITU-T Recommendation P.800: Methods for Subjective Determination of Transmission Quality*. `http://www.itu.int/rec/T-REC-P.800-199608-I/en`, accessed July 15, 2013. Aug. 1998.

[6] R. Likert. “A technique for the measurement of attitudes.” In: *Archives of Psychology* 22.140 (1932), pp. 1–55.

[7] Patrick Meier. *Web App Tracks Breaking News Using Wikipedia Edits*. `http://irevolution.net/2013/04/23/breaking-news-using-wikipedia-edits/`, accessed July 15, 2013. 2013.

[8] Steven L. Scott. “A modern Bayesian look at the multi-armed bandit”. In: *Appl. Stoch. Model. Bus. Ind.* 26.6 (Nov. 2010), pp. 639–658. ISSN: 1524-1904.

[9] Irina Shklovski, Leysia Palen, and Jeannette Sutton. “Finding community through information and communication technology in disaster response”. In: *Proceedings of the 2008 ACM conference on Computer supported cooperative work*. CSCW ’08. San Diego, CA, USA: ACM, 2008, pp. 127–136. ISBN: 978-1-60558-007-4.

[10] Jeannette Sutton, Leysia Palen, and Irina Shklovski. “Backchannels on the front lines: Emergent uses of social media in the 2007 southern California wildfires”. In: *Proceedings of the 5th International ISCRAM Conference*. Washington, DC. 2008, pp. 624–632.

12

Appendices

Curriculum Vitæ

Personal Data

Thomas Steiner
Bäckerbreitergang 12, 20355 Hamburg, Germany

Born December 17, 1981 in Freudenstadt

Education

2013–2014	*Postdoctoral Researcher*, Université Claude Bernard Lyon 1, Lyon (France).
2010–2014	*PhD candidate* in Computing, Universitat Politècnica de Catalunya, Barcelona (Spain).
2005–2007	Double degree *Master of Computer Science*, Karlsruhe Institute of Technology (Germany) and ENSIMAG Grenoble (France). Thesis: Thomas Steiner. "Automatic Multi Language Program Library Generation for REST APIs". MA thesis. Karlsruhe Institute of Technology, 2007. URL: `http://www.iks.kit.edu/fileadmin/User/calmet/stdip/DA-Steiner.pdf`
2002–2005	*Bachelor of Computer Science*, Karlsruhe Institute of Technology (Germany).

2001 *Final secondary-school examinations*, Isolde-Kurz-Gymnasium Reutlingen (Germany).

Professional Experience

2013–present *Customer Solutions Engineer* at Google Germany GmbH, Hamburg.

2010–2013 *Research Engineer*, Google Germany GmbH, Hamburg.

Worked on the EU project *I-SEARCH*, which created a novel unified framework for multimedia and multimodal content indexing, sharing, search, and retrieval. I-SEARCH was the first multimodal search engine able to handle multimedia (text, 2D image, sketch, video, 3D objects, audio), multimodal content (gestures, face expressions), combined with real-world information (GPS, time, weather).

2007–2010 *Customer Solutions Engineer* at Google Germany GmbH, Hamburg.

Worked in the Google Technical Services (gTech) team, which serves as the primary point of contact for Google's global Sales, Business Development, and Partnerships teams to support the sales organization across all products.

Publications

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, Erik Mannens, and Rik Van de Walle. "Clustering Media Items Stemming from Multiple Social Networks". In: *The Computer Journal* (2013). DOI: `10.1093/comjnl/bxt147`. eprint: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.full.pdf+html`. URL: `http://comjnl.oxfordjournals.org/content/early/2013/12/29/comjnl.bxt147.abstract`.

- Ruben Verborgh, Michael Hausenblas, Thomas Steiner, Erik Mannens, and Rik Van de Walle. “Distributed Affordance: An Open-World Assumption for Hypermedia”. In: *Proceedings of the Fourth International Workshop on RESTful Design*. May 2013. URL: `http://distributedaffordance.org/publications/ws-rest2013.pdf`.

- Thomas Steiner, Seth van Hooland, Ruben Verborgh, Joseph Tennis, and Rik Van de Walle. “Identifying VHS Recording Artifacts in the Age of Online Video Platforms”. In: *Proceedings of the 1st international Workshop on Search and Exploration of X-rated Information*. Feb. 2013. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2013/identifying-vhs-recording-sexi2013.pdf`.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. “Near-duplicate Photo Deduplication in Event Media Shared on Social Networks”. In: *Proceedings of the International Conference on Advanced IT, Engineering and Management*. Feb. 2013, pp. 187–188.

- Thomas Steiner. “A meteoroid on steroids: ranking media items stemming from multiple social networks”. In: *Proceedings of the 22nd international conference on World Wide Web companion*. WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 31–34. ISBN: 978-1-4503-2038-2. URL: `http://dl.acm.org/citation.cfm?id=2487788.2487798`.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. “Adding Meaning to Social Network Microposts via Multiple Named Entity Disambiguation APIs and Tracking Their Data Provenance”. In: *International Journal of Computer Information Systems and Industrial Management* 5 (2013), pp. 69–78. URL: `http://www.mirlabs.org/ijcisim/regular_papers_2013/Paper82.pdf`.

- Thomas Steiner. “Enriching Unstructured Media Content About Events to Enable Semi-Automated Summaries, Compilations, and Improved Search by Leveraging Social Networks”. PhD thesis. Universitat Politècnica de Catalunya, 2013.

- Vuk Milicic, Giuseppe Rizzo, José Luis Redondo Garcia, Raphaël Troncy, and Thomas Steiner. “Live topic generation from event streams”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 285–288. ISBN: 978-1-4503-2038-2. URL: http://dl.acm.org/citation.cfm?id=2487788.2487924.

- Thomas Steiner, Seth van Hooland, and Ed Summers. “MJ no more: using concurrent wikipedia edit spikes with social network plausibility checks for breaking news detection”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 791–794. ISBN: 978-1-4503-2038-2. URL: http://dl.acm.org/citation.cfm?id=2487788.2488049.

- Seth van Hooland, Max De Wilde, Ruben Verborgh, Thomas Steiner, and Rik Van de Walle. “Named-Entity Recognition: A Gateway Drug for Cultural Heritage Collections to the Linked Data Cloud?” In: *Literary and Linguistic Computing* (2013). URL: http://freeyourmetadata.org/publications/named-entity-recognition.pdf.

- Ruben Verborgh, Thomas Steiner, Erik Mannens, Rik Van de Walle, and Joaquim Gabarró Vallés. “Proof-based Automated Web API Composition and Integration”. In: *Proceedings of the International Conference on Advanced IT, Engineering and Management* (2013), pp. 181–182.

- Ruben Verborgh, Andreas Harth, Maria Maleshkova, Steffen Stadtmüller, Thomas Steiner, Mohsen Taheriyan, et al. “Semantic Description of REST APIs”. In: *rest: Advanced Research Topics and Practical Applications* (2013).

- Thomas Steiner and Christopher Chedeau. “To crop, or not to crop: compiling online media galleries”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* WWW ’13 Companion. Rio de Janeiro, Brazil: International World Wide Web Conferences Steering Committee, 2013, pp. 201–202. ISBN: 978-1-4503-2038-2. URL: http://dl.acm.org/citation.cfm?id=2487788.2487890.

- Thomas Steiner, Ruben Verborgh, Raphael Troncy, Joaquim Gabarro, and Rik Van de Walle. "Adding Realtime Coverage to the Google Knowledge Graph". In: *Proceedings of the ISWC 2012 Posters & Demonstrations Track, Boston, USA, November 11–15, 2012*. Ed. by Birte Glimm and David Huynh. Vol. 914. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2012. URL: `http://ceur-ws.org/Vol-914/paper_2.pdf`.

- Ruben Verborgh, Vincent Haerinck, Thomas Steiner, Davy Van Deursen, Sofie Van Hoecke, Jos De Roo, et al. "Functional Composition of Sensor Web APIs ". In: *Proceedings of the 5th International Workshop on Semantic Sensor Networks, A Workshop of the 11th International Semantic Web Conference 2012 (ISWC 2012), Boston, Massachusetts, USA, November 12, 2012*. Ed. by Cory Henson, Kerry Taylor, and Oscar Corcho. Vol. 904. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2012, pp. 65–80. URL: `http://ceur-ws.org/Vol-904/paper6.pdf`.

- Thomas Steiner and Stefan Mirea. *SEKI@home, a Generic Approach for Crowdsourcing Knowledge Extraction from Arbitrary Web Pages*. Nov. 2012. URL: `http://challenge.semanticweb.org/2012/submissions/swc2012_submission_28.pdf`.

- Thomas Steiner and Stefan Mirea. "SEKI@home, or Crowdsourcing an Open Knowledge Graph". In: *Proceedings of the 1st International Workshop on Knowledge Extraction & Consolidation from Social Media, in conjunction with the 11th International Semantic Web Conference (ISWC 2012), Boston, USA, November 12, 2012*. Ed. by Diana Maynard, Stefan Dietze, Wim Peters, and Jonathon Hare. Vol. 895. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2012. URL: `http://ceur-ws.org/Vol-895/paper2.pdf`.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarro, and Rik Van de Walle. "Defining Aesthetic Principles for Automatic Media Gallery Layout for Visual and Audial Event Summarization based on Social Networks". In: *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on.* July 2012, pp. 27–28. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/defining-aesthetic-principles-for-automatic-media-gallery-layout-qomex2012.pdf`.

- Houda Khrouf, Ghislain Atemezing, Giuseppe Rizzo, Raphaël Troncy, and Thomas Steiner. “Aggregating Social Media for Enhancing Conference Experience”. In: *Real-Time Analysis And Mining of Social Streams, Papers from the 2012 ICWSM Workshop.* Ed. by Arkaitz Zubiaga, Maarten de Rijke, Markus Strohmaier, and Mor Naaman. AAAI Technical Report WS-12–02. June 2012. URL: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/download/4779/5086.

- Ruben Verborgh, Thomas Steiner, Rik Van de Walle, and Joaquim Gabarró Vallés. “The Missing Links – How the Description Format RESTdesc Applies the Linked Data Vision to Connect Hypermedia APIs”. In: *Proceedings of the First Linked APIs Workshop at the Ninth Extended Semantic Web Conference.* May 2012. URL: http://lapis2012.linkedservices.org/papers/3.pdf.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, Michael Hausenblas, Raphaël Troncy, and Rik Van de Walle. *Enabling on-the-fly Video Shot Detection on YouTube.* Apr. 2012.

- Thomas Steiner and Ruben Verborgh. *Fixing the Web One Page at a Time, or Actually Implementing xkcd #37.* Apr. 2012. URL: http://www2012.org/proceedings/nocompanion/DevTrack_032.pdf.

- Ruben Verborgh, Thomas Steiner, Joaquim Gabarró Vallés, Erik Mannens, and Rik Van de Walle. “A Social Description Revolution—Describing Web APIs’ Social Parameters with RESTdesc”. In: *Proceedings of the AAAI 2012 Spring Symposia.* Mar. 2012. URL: http://www.aaai.org/ocs/index.php/SSS/SSS12/paper/viewFile/4283/4665.

- Ruben Verborgh, Thomas Steiner, Davy Deursen, Jos Roo, RikVan de Walle, and Joaquim Gabarró Vallés. “Capturing the functionality of Web services with functional descriptions”. In: *Multimedia Tools and Applications* (2012), pp. 1–23. ISSN: 1380-7501. URL: http://rd.springer.com/content/pdf/10.1007/s11042-012-1004-5.

- Houda Khrouf, Ghislain Atemezing, Thomas Steiner, Giuseppe Rizzo, and Raphaël Troncy. *Confomaton: A Conference Enhancer with Social Media from the Cloud.* 2012. URL: `http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_343.pdf`.

- Jonas Etzold, Arnaud Brousseau, Paul Grimm, and Thomas Steiner. "Context-Aware Querying for Multimodal Search Engines". In: *Advances in Multimedia Modeling.* Ed. by Klaus Schoeffmann, Bernard Merialdo, Alexander G. Hauptmann, et al. Vol. 7131. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 728–739. ISBN: 978-3-642-27354-4. URL: `http://research.google.com/pubs/archive/37423.pdf`.

- Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Sam Coppens, Joaquim Gabarró Vallés, and Rik Van de Walle. "Functional Descriptions as the Bridge between Hypermedia APIs and the Semantic Web". In: *Proceedings of the Third International Workshop on RESTful Design.* WS-REST '12. Lyon, France: ACM, 2012, pp. 33–40. ISBN: 978-1-4503-1190-8. URL: `http://ws-rest.org/2012/proc/a5-9-verborgh.pdf`.

- Thomas Steiner, Lorenzo Sutton, Sabine Spiller, Marilena Lazzaro, Francesco Nucci, Vincenzo Croce, et al. "I-SEARCH: A Multimodal Search Engine based on Rich Unified Content Description (RUCoD)". in: *Proceedings of the 21^{st} International Conference Companion on World Wide Web.* WWW '12 Companion. Lyon, France: ACM, 2012, pp. 291–294. ISBN: 978-1-4503-1230-1. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/isearch-multimodal-search-www2012.pdf`.

- Apostolos Axenopoulos, Petros Daras, Sotiris Malassiotis, Vincenzo Croce, Marilena Lazzaro, Jonas Etzold, et al. "I-SEARCH: A Unified Framework for Multimodal Search and Retrieval". In: *The Future Internet.* Ed. by Federico Álvarez, Frances Cleary, Petros Daras, John Domingue, Alex Galis, Ana Garcia, et al. Vol. 7281. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 130–141. ISBN: 978-3-642-30240-4. URL: `http://www.lsi.upc.edu/~tsteiner/papers/2012/isearch-a-unified-framework-for-multimodal-search-and-retrieval.pdf`.

- R. Troncy, E. Mannens, S. Pfeiffer, D. Van Deursen, M. Hausenblas, P. Jägenstedt, et al. *Media Fragments URI 1.0 (basic).* Recommendation. http://www.w3.org/TR/media-frags/, accessed July 15, 2013. W3C, 2012.

- Thomas Steiner, Marilena Lazzaro, Francesco Nucci, Vincenzo Croce, Lorenzo Sutton, Alberto Massari, et al. "One Size Does Not Fit All: Multimodal Search on Mobile and Desktop Devices with the I-SEARCH Search Engine". In: *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval.* ICMR '12. Hong Kong, China: ACM, 2012, 58:1–58:2. ISBN: 978-1-4503-1329-2. URL: http://www.lsi.upc.edu/~tsteiner/papers/2012/one-size-does-not-fit-all-icmr2012.pdf.

- Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Sam Coppens, Erik Mannens, Rik Van de Walle, et al. *RESTdesc—A Functionality-Centered Approach to Semantic Service Description and Composition.* 2012. URL: http://2012.eswc-conferences.org/sites/default/files/eswc2012_submission_302.pdf.

- Giuseppe Rizzo, Thomas Steiner, Raphaël Troncy, Ruben Verborgh, José Luis Redondo García, and Rik Van de Walle. "What Fresh Media Are You Looking For?: Retrieving Media Items From Multiple Social Networks". In: *Proceedings of the 2012 International Workshop on Socially-aware Multimedia.* SAM '12. Nara, Japan: ACM, 2012, pp. 15–20. ISBN: 978-1-4503-1586-9. URL: http://www.eurecom.fr/~troncy/Publications/Troncy-saw12.pdf.

- Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Jos De Roo, Rik Van de Walle, and Joaquim Gabarró Vallés. "Description and Interaction of RESTful Services for Automatic Discovery and Execution". In: *Proceedings of the FTRA 2011 International Workshop on Advanced Future Multimedia Services.* Dec. 2011. URL: https://biblio.ugent.be/publication/2003291/file/2003308.pdf.

- Thomas Steiner, Ruben Verborgh, Joaquim Gabarró Vallés, and Rik Van de Walle. "Adding Meaning to Facebook Microposts via a Mash-up API and Tracking its Data Provenance". In: *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on.* Oct. 2011, pp. 342–345. URL: http://research.google.com/pubs/archive/37426.pdf.

- Thomas Steiner, Ruben Verborgh, and Michael Hausenblas. “Crowdsourcing Event Detection in YouTube Videos”. In: *Proceedings of the Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011), Workshop in conjunction with the 10th International Semantic Web Conference 2011 (ISWC 2011), Bonn, Germany, October 23, 2011.* Ed. by Marieke van Erp, Willem Robert van Hage, Laura Hollink, Anthony Jameson, and Raphaël Troncy. Vol. 779. CEUR Workshop Proceedings ISSN 1613-0073. Oct. 2011, pp. 58–67. URL: `http://ceur-ws.org/Vol-779/derive2011_submission_8.pdf`.

- Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Rik Van de Walle, and Joaquim Gabarró Vallés. “Efficient Runtime Service Discovery and Consumption with Hyperlinked RESTdesc”. In: *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on.* Oct. 2011, pp. 373–379. URL: `http://research.google.com/pubs/archive/37427.pdf`.

- Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Sam Coppens, Erik Mannens, Rik Van de Walle, et al. “Integrating Data and Services through Functional Semantic Service Descriptions”. In: *Proceedings of the W3C Workshop on Data and Services Integration.* Oct. 2011. URL: `http://www.w3.org/2011/10/integration-workshop/p/integration-ws-mmlab.pdf`.

- Thomas Steiner, Arnaud Brousseau, and Raphaël Troncy. *A Tweet Consumers’ Look At Twitter Trends.* May 2011. URL: `http://research.hypios.com/msm2011/posters/steiner.pdf`.

- Thomas Steiner. “DC Proposal: Enriching Unstructured Media Content About Events to Enable Semi-Automated Summaries, Compilations, and Improved Search by Leveraging Social Networks”. In: *Proceedings of the 10th International Conference on The Semantic Web – Volume Part II.* ISWC’ 11. Bonn, Germany: Springer-Verlag, 2011, pp. 365–372. ISBN: 978-3-642-25092-7. URL: `http://iswc2011.semanticweb.org/fileadmin/iswc/Papers/DC_Proposals/70320369.pdf`.

- Thomas Steiner and Jan Algermissen. “Fulfilling the Hypermedia Constraint via HTTP OPTIONS, the HTTP Vocabulary in RDF, and Link Headers”. In: *Proceedings of the Second International Workshop on RESTful Design.* WS-REST ’11. Hyderabad, India: ACM, 2011, pp. 11–14. ISBN: 978-1-4503-0623-2. URL: `http://ws-rest.org/2011/proc/a3-steiner.pdf`.

- Maria Alduan, Federico Álvarez, Jan Bouwen, Gonzalo Camarillo, Pablo Cesar, Pedros Daras, et al. *Future Media Internet Architecture Reference Model (v1. 0).* 2011. URL: `http://www.coast-fp7.eu/public/FMIA_Reference_Architecture.pdf`.

- Petros Daras, Apostolos Axenopoulos, Vasileios Darlagiannis, Dimitrios Tzovaras, Xavier Le Bourdon, Laurent Joyeux, et al. “Introducing a Unified Framework for Content Object Description”. In: *Multimedia Intelligence and Security* 2.3 (2011), pp. 351–375. ISSN: 2042–3470. URL: `http://www.iti.gr/iti/files/document/work/IJMIS0203-0409%20DARAS.pdf`.

- Thomas Steiner, Raphaël Troncy, and Michael Hausenblas. “How Google is using Linked Data Today and Vision For Tomorrow”. In: *Proceedings of the Workshop on Linked Data in the Future Internet at the Future Internet Assembly, Ghent 16–17 Dec 2010.* Ed. by Sören Auer, Stefan Decker, and Manfred Hauswirth. Vol. 700. CEUR Workshop Proceedings ISSN 1613-0073. Dec. 2010. URL: `http://CEUR-WS.org/Vol-700/Paper5.pdf`.

- Thomas Steiner. “SemWebVid – Making Video a First Class Semantic Web Citizen and a First Class Web Bourgeois”. In: *Proceedings of the ISWC 2010 Posters & Demonstrations Track: Collected Abstracts, Shanghai, China, November 9, 2010.* Ed. by Axel Polleres and Huajun Chen. Vol. 658. CEUR Workshop Proceedings ISSN 1613-0073. Nov. 2010, pp. 97–100. URL: `http://ceur-ws.org/Vol-658/paper469.pdf`.

- Thomas Steiner and Michael Hausenblas. *SemWebVid – Making Video a First Class Semantic Web Citizen and a First Class Web Bourgeois, Submission to the Open Track of the Semantic Web Challenge 2010.* Nov. 2010. URL: `http://challenge.semanticweb.org/submissions/swc2010_submission_12.pdf`.

Declaration

I herewith declare that I have produced this document without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This document has not previously been presented in identical or similar form to any other examination board.

The thesis work was conducted from February 2010 to July 2013 under the supervision of Joaquim Gabarró Vallés (Universitat Politècnica de Catalunya, Barcelona, Spain) and Michael Hausenblas (Digital Enterprise Research Institute, Galway, Ireland and MapR Technologies, San Jose, CA, USA).

Barcelona, in February 2014