Social Web Data Analytics: Relevance, Redundancy, Diversity

Ke Tao

Social Web Data Analytics: Relevance, Redundancy, Diversity

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op dinsdag 9 december 2014 om 10:00 uur

door **Ke TAO**

Master of Engineering in Computer Science and Technology,
National University of Defense Technology, China,
geboren te Beijing, China.

Dit proefschrift is goedgekeurd door de promotoren:
Prof.dr.ir. G.J.P.M. Houben
Copromotor: Dr. C. Hauff

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof.dr.ir. G.J.P.M. Houben	Technische Universiteit Delft, promotor
Dr. C. Hauff	Technische Universiteit Delft, co-promotor
Prof.dr. A. Hanjalic	Technische Universiteit Delft
Prof.dr.ir. W. Kraaij	Radbound Universiteit Nijmegen
Prof.dr. J. Lin	University of Maryland
Prof.dr. M. Strohmaier	GESIS Leibniz Institut für Sozialwissenschaften
Dr. F. Abel	XING AG
Prof.dr. E. Visser	Technische Universiteit Delft, reservelid

SIKS Dissertation Series No. 2014-46

The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

This work is supported by the China Scholarship Council.

Published and distributed by: Ke Tao
E-mail: tao.ke@me.com

ISBN: 978-94-6186-396-6

Keywords: Social Web, Data Analytics, Information Retrieval, Twitter

Cover images: (front) Traces, Grand Central Station, New York, the United States, July 3rd, 2012; (back) Model, Miniatur Wunderland, Hamburg, Germany, November 22nd, 2013.
Cover images & design by: Ke Tao

Printed and bound in the Netherlands by CPI Wörmann Print Service.

Dedicated to my beloved parents
献给我亲爱的爸妈

Acknowledgments

Finally, I started writing this section, which came at the last chronologically. And probably, this will be the texts that have the widest audience in this book of more than 200 pages. Since the first day that I landed in the Netherlands, it will have been 1,528 days until the defence of this thesis. During this process, many people have helped me, experienced with me, shared with me, and finally I become myself as of today. Herewith, I would like to express my appreciation to all of you and list some of your names as well as our stories briefly.

First of all, I would like to express my gratitude to my promotor Prof.dr.ir. Geert-Jan Houben. In early 2010, I made my decision to come to our Web Information Systems group because of your patience and sense of responsibility that I felt from the Skype interviews that seemed not having an end. Throughout the four years, I have been receiving advices and encouragements from you on both work and life. Not long after my arrival in this country, which is about 8000 km away from home, I received your wishes of my first birthday in the Netherlands, at around 5 a.m. via an email. I always feel being lucky to have the chance to pursue my PhD study under your supervision.

I have to admit that I owe a lot to my two daily supervisors: my co-pomotor Dr. Claudia Hauff and my advisor Dr. Fabian Abel. This thesis would not have been possible without the help from you. Claudia, thanks for your detailed feedback for every piece of text and I am quite impressed by the work that you did when you were about to give birth to your lovely daughter Zoe. Fabian, you are the best office mate that I have ever had. Frequently, I cherish the memory of hearing the announcement of closing the building and running out before 10:30 p.m. Thanks for the advices and instructions that you have given me during our numerous meetings.

Besides my promotor and two daily-supervisors, I would like to thank other committee members of my thesis defence: Prof.dr. Alan Hanjalic, Prof.dr.ir. Wessel Kraaij, Prof.dr. Jimmy Lin, Prof.dr. Markus Strohmaier, and Prof.dr. Eelco Visser. Thanks a lot for your time spent on this thesis and your valuable feedback.

In the Web Information Systems group, I have met a lot of learned and energetic researchers: Dr. Jan Hidders, Dr. Laura Hollink, Dr. Alessandro Bozzon, Dr. Stefano Bocconi, Dr. Qi Gao (高琦), Dr. Beibei Hu (胡蓓蓓), Jie Yang (杨杰), Jasper Oosterman, Dr. Ilknur Çelik, Richard Stronkman, and Dr. Damir Juric. I have learned a lot from you during the lunches, coffee breaks, and outings that we had together. I sincerely hope that some of the current members can take over my position in WISCof[1] and keep our mug full of notes and coins. Apart from the work done in WIS group, I have also collaborated with a number of outstanding researchers: Dr. Guido Wachsmuth, Dr. Elaheh Momenim, Ujwal Gadiraju. I feel really honored to work with you and get our papers published. CuiTing Chen (陈翠婷), Tiago Espinha (赵飞龙), Alberto González, and Èric Piel, I feel fortunate for having you on the same floor and a lot of inspiring and interesting conversations between us in the coffee room.

Soon after I started working in TU Delft, I noticed that the supporting staffs are extremely helpful and allow me to focus on my research work. Therefore, I would like to thank Paulo Anita, Stephen van der Laan for their excellent ICT supports, Esther van Rooijen, Tamara Brusik, Franca Post, Rina Abbriata, and Ilse Oonk, who provide us with considerate administrative supports.

ZhiJie Ren (任之劼), we have known each other for 15 years and it was you who further convinced me to choose TU Delft for pursuing my PhD study. You made me feel not far from home. And more realistically, it feels great to live in your apartment situated in the city center of Delft. Of course, I would like to thank your parents, who visited us from time to time and treated us for various delicious food for so many times.

Many thanks for my Chinese friends who often get together with me for a drink or a meal in the Netherlands: Jian Fu (付剑), XiaoYu Zhang (张晓禹), Li Mo (莫雳), HaiQiang Wang (王海强), Zhou Zhou (周舟), SiQi Shen (沈思淇), JianBin Fang (方建滨), WenYan Li (李文砚), Xin Wang (王鑫), Qiaole Zhao (赵俏功), Xin Wang (王昕), and YuHui Peng (彭玉慧). The

[1] http://wis.ewi.tudelft.nl/wiscof/, accessed November 7th, 2014.

preparation of every meal makes my cooking skills better and I will definitely miss the time we spent together. Particularly, I would like to give special thanks to YongJia Li (李泳佳) for being my paranymph.

I think I have made much more trips than I had before coming to the Netherlands. For many of those trips, I have been offered a number of free guided tours and couches in many places all over the world. Dawei Feng (冯大为), Chengkun Wu (吴诚堃), Mingtang Deng (邓明堂), and Yabing Liu (刘亚冰), thank you for being so helpful and generous!

Furthermore, I have been keeping in touch with the teachers and mates from my alma mater, National University of Defence Technology in Changsha, China. Prof.dr. YiJie Wang (王意洁), Prof.dr. Ting Wang (王挺), Dr. Yue Liu (刘越), thanks for your support during my application to TU Delft. Lei Li (李磊), JinGang Xie (解金刚), Kai Zhang (张凯), and Hui Song (宋辉), thank you for having me whenever I went back to BeiJing. ShiCe Ni (倪时策) and RongChun Li (李荣春), thanks a lot of taking care of the trivial administrations in China so that I can fully focus on my research work.

Thanks to the 4-year life in BeiJing No.8 Middle School, it gives me the chance to know a lot of talented people. Taoyu Li (李洮禹), Fan Yang (杨帆), Bo Qin (秦博), WangYi Liu (刘往一). I have been enjoying to exchange a lot with you. MengDi Wang (王梦迪), ZongXi Li (李宗溪), HaoSheng Cui (崔浩生), ShiMeng Cheng (程诗萌), and Tong Meng (孟瞳), thank you for your treats in the States!

Special thanks go to those who remind myself about the weakness that I might have. The most special one is Cheryl Guan (关丞), whom I met during her business trip on the last stage of my 4-year journey. She constantly kept alerting me on some mistakes that I might make, though according to a theory that I, as a scientific researcher, could not believe in. Moreover, thanks for planning to travel for thousands of miles to attend my thesis defence and become my paranymph.

Ying Zhu (朱颖), you might noticed that the opening of this thesis actually refers to my attempt for reaching you after the earthquake in 2011. Thank you so much for the beautiful postcards you have sent me from Japan!

XiaoXing Li (李晓星), Kai Zhang (张开), JinLi Qiu (邱劲励), thank you very much for having me in the production team for our podcast "JiangY-

ouWeiBo" (酱油微播)[2]. JinLi, I really appreciate your help with designing the cover of this thesis.

Finally, I want to thank my family and other relatives in China, especially my mother and my aunt for travelling so far to attend my thesis defence. Thank you for your support and I love all of you.

最后我要感谢我的父母以及其他在国内的亲人们，尤其是不远万里来参加我论文答辩意识的妈妈和小姨。感谢你们多年来对我的支持，我爱你们。

Ke Tao
November 2014
Delft, the Netherlands

[2]http://jywave.com/, accessed November 7th, 2014.

Foreword

In the last four years of my PhD study, I have been working on solving the following problems: how can we fulfil various information needs by conducting analytics with Social Web data and how can we build a system to make the construction of such analytics simpler?

Given the main requirement of fulfilling information needs by using the Social Web, I have studied different aspects, including relevance, redundancy, and diversity of Twitter data by conducting different analytical tasks in the context of information retrieval. My initial idea was to investigate how the semantics in Social Web data can help in meeting this requirement. The storyline behind my work is described as follows.

Motivated by the task of the TREC Microblog Track that was first introduced in 2011, we exploit the usage of background knowledge for a *query expansion framework* by referring the semantic links to Linked Open Data Cloud and news articles[1]. Then we further propose our *relevance estimation framework* to predict the relevance of tweets to a given topic, taking the results from the previous work as one feature of which the importance can be analyzed. Hence, the framework[2] not only considers the retrieval score given by the classical language model, for both the original queries and the expanded version derived from the aforementioned work, but also the features that do not depend on the given queries, such as syntactic characteristics, semantics, and contextual information. The extensive evaluation

[1]Published as: WISTUD at TREC 2011 Microblog Track: Exploiting Background Knowledge from DBpedia and News Articles for Search on Twitter. By K. Tao, F. Abel, C. Hauff. In Proceedings of The Twentieth Text REtrieval Conference (TREC'12), Gaithersburg, Maryland, 2011

[2]Published as: What makes a tweet relevant for a topic? By K. Tao, F. Abel, C. Hauff, G.J. Houben. In Proceedings of the workshop on Making Sense of Microposts (MSM2012), workshop at the 21st World Wide Web Conference 2012 (WWW'12), Lyon, France

with a standard corpus leads us to interesting findings in the relevance of tweets to a given topic, of which we make use to improve the retrieval effectiveness. Moreover, we put our findings into practice and propose *Twinder*, which is a search engine for Twitter streams[3]. This search engine serves as a playground to conduct further analytical research on Twitter search.

Having noticed the occurrences of duplicate content in microblogging search results even after filtering out retweets, we are motivated to further investigate redundancy in Twitter data. We introduce a *framework for near-duplicate detection on Twitter*[4]. We infer a model for duplication levels between tweet pairs based on a case-study of microblog search results. Then we develop a framework which can utilize the machine learning algorithms to automatically identify the near-duplicate pairs and their level of duplication with the features that we construct by applying syntactic, semantic, and contextual analyses. The evaluations on representative dataset results show that, with our effective strategies, the redundancy in search results can be reduced by around 50%. Again, we integrate the outcomes into Twinder and improve the quality of search results[5].

Based on the analysis of redundancy in Twitter data, we further aim at diversifying microblog search results and analyzing the impact of reducing duplicates on diversity. However, the lack of an existing corpus for diversification research on microblog search makes it harder for us to do so. Hence, we present a *methodology of building such a corpus*[6]. *The corpus* is made available to public for further research. A comprehensive analysis of the corpus shows its suitability for the research on search result diversification. Moreover, we evaluate the diversity of search results derived from the application of our duplicate detection framework. Again, we find a redundancy decrease achieved by applying our de-duplicate strategies. Meanwhile, we discover the importance of the features for the topic types, e.g. long-term versus short-term topics and topic recency, in diversification.

[3]Published as: Twinder: A Search Engine for Twitter Streams. By K. Tao, F. Abel, C. Hauff, G.J. Houben. In Proceedings of the 12th International Conference on Web Engineering (ICWE'12), Berlin, Germany, 2012

[4]Published as: Groundhog Day: Near-Duplicate Detection on Twitter. By K. Tao, F. Abel, C. Hauff, G.J. Houben, U. Gadiraju. In Proceedings of the 22nd International World Wide Web Conference (WWW'13), Rio de Janeiro, Brazil, 2013

[5]Published as: Twinder: Enhancing Twitter Search. By K. Tao, Fabian Abel, Claudia Hauff, Geert-Jan Houben, Ujwal Gadiraju. In PROMISE Winter School 2013: Bridging between Information Retrieval and Databases. Springer, 2013

[6]Published as: Building a Microblog Corpus for Search Result Diversification. By K. Tao, C. Hauff, G.J. Houben. In Proceedings of 9th Asia Information Retrieval Societies Conference (AIRS'13), Singapore, 2013

Based on summarizing both the research that we have done with Twitter data and the survey of analytics with information from the same source[7], we distill our research methodologies into a set of common tools for conducting Social Web data analytics on Twitter. Therefore, we propose the *Twitter Analytical Platform* that allows application developers, scientists, etc. to understand Twitter data in their own perspectives[8]. This platform, which can be customized by using *Twitter Analysis Language*, implements the functions of data acquisition, manipulation, enrichment, aggregation, as well as integration with machine learning capabilities. We show the validity of the platform with the successful implementation of above three analytical tasks in Twitter Analysis Language.

Besides research on general scientific problems, we also look into the real-life challenges in order to see how Twitter Analytical Platform can support the application in production. For instance, it is a non-trivial challenge to fulfil the information need during a real-world incident. Therefore, we introduce *Twitcident*[9], a system that relies on Twitter Analytical Platform to automatically filter relevant information about a real-world incident from Twitter streams and make the information accessible and findable in the given context of the incident. Consequently, the processed data given by our platform provides support for the applications, including faceted search and visualized analytics that allow people and emergency services to retrieve particular information fragments as well as overview and analyze the current situation as reported on Twitter. The large-scale evaluation proves that the semantic enrichment offered by our platform leads to major and significant improvements of both the filtering and the search performance.

The additional information, including datasets, experimental code, and demonstrations, on this PhD thesis is available online at http://ktao.github.io/phd/.

[7]Published as: Information Retrieval for Twitter Data. By K. Tao, C. Hauff, F. Abel, G.J. Houben. Book Chapter In Twitter and Society. Peter Lang, 2013.

[8]Published as: Facilitating Twitter Data Analytics: Platform, Language, and Functionality. By K. Tao, C. Hauff, G.J. Houben, F. Abel, G. Wachsmuth. In Proceedings of 2014 IEEE International Conference on Big Data (IEEE BigData'14), Washington DC, USA, 2014

[9]Published as: i) Twitcident: Fighting Fire with Information from Social Web Streams. By F. Abel, C. Hauff, G.J. Houben, R. Stronkman, K. Tao. In Companion Proceedings of International Conference on World Wide Web (WWW'12), Lyon, France, 2012; and ii) Semantics + Filtering + Search = Twitcident. Exploring Information in Social Web Streams. By F. Abel, C. Hauff, G.J. Houben, R. Stronkman, K. Tao. In Proceedings of International Conference on Hypertext and Social Media (Hypertext'12), Milwaukee, USA, 2012

Contents

Chapter 1

Introduction

1.1 Motivation

As early as in the first year of my PhD study, an earthquake[1] struck Japan and I got the news from my mobile phone in the morning. People who called from all over the world to Japan did not get through due to the failure of telephone networks [81]. However, the messages confirming the safety of their loved ones were sent across the globe via the Social Web[2]. It was found that the volume of messages sent through the Social Web and Twitter in particular reached 5,000 tweets per seconds for several times. It motivated researchers to make use of this data to investigate information diffusion on the Social Web and deploy the applications inspired by this research for the general public. One of the first applications in this area is an earthquake early warning system based on Twitter data, which is able to provide warnings 2 minutes faster than traditional warning systems [51]. Besides the Internet, special thanks should go to Sir Tim Berners-Lee. In 1989, he proposed a system that aims at making the information sharing between scientists working at CERN more effective [16]. This system, which is known as the World Wide Web or now the Web, was realized not only in a small academic circle, but at a global scope. Thus, more and more hypertext documents get interlinked in the World Wide Web so that one can navigate between them via hyperlinks with a Web browser.

During the past decades, Web technologies have tremendously changed

[1] http://earthquake.usgs.gov/earthquakes/eqinthenews/2011/usc0001xgp/, accessed July 30th, 2014

[2] https://blog.twitter.com/2011/global-pulse, accessed July 30th, 2014

the mechanisms of information exchange between individuals or groups of people. Furthermore, the development of a second generation of Web applications, Web 2.0 applications, has allowed for content authoring by every single user that is connected to the Internet. Part of these applications constitute the Social Web, which enables people to engage with each other at a relatively low threshold and motivate individual users to participate in the sharing of information. The amount of messages shared on Social Web platforms, via systems like Twitter, Sina Weibo, Facebook, or YouTube, is so large that details about people's daily lives in all extent are shared via easily composed snippets in different media forms, including text, locations, images, or videos. Taking Twitter as an example, the number of posts published per day typically exceeds several hundred million[3] while the number of monthly active users has reached 255 million[4]. During the 2014 FIFA World Cup[5], 32.1 million tweets were posted during the final match while the peak volume reached 618,725 tweets per minute when Germany won the championship. Such huge amount of data becomes a source for people to exploit its values in different scenarios. Researchers have attempted, with Social Web data, to warn people of an earthquake wave ahead of traditional systems, to profile users [3, 104], to analyze the travelling patterns of people [11, 37], and even to predict the results of political elections [121]. Furthermore, many profitable and valuable ideas have been implemented to support decision making for business cases [35] and public interests [5]. These works can be considered as conducting analytics with data from the Social Web, i.e. finding meaningful patterns of knowledge in Social Web data to fulfill information needs or provide the "knowledge to act". Due to the characteristics of Social Web data, it is a non-trivial challenge to conduct data analytics to fulfill those specific information needs.

Among the numerous Web applications that can provide information relevant for a specific information need, the most commonly used service is the Web search engine, as represented by Google [24]. With the exponential growth of contents on the Web [75], such systems make use of information retrieval techniques to facilitate information discovery processes on the Web with simple keyword queries. Building such systems involves multiple phases, including crawling [31, 38], indexing [165], searching [131], etc., and prompts researchers to make the information finding more efficient and

[3] https://blog.twitter.com/2013/new-tweets-per-second-record-and-how, accessed July 30th, 2014

[4] https://investor.twitterinc.com/releasedetail.cfm?ReleaseID=843245, accessed July 30th, 2014

[5] https://blog.twitter.com/2014/the-roar-of-the-crowd-for-the-worldcupfinal, accessed July 30th, 2014

effective with diverse research efforts, such as evaluating the importance of documents [124], personalization [79], diversification [135], or adapting to human browsing preferences [56]. Given the substantial amount of information generated on Social Web applications, the new challenges, which are introduced by the incapability of existing solutions for adapting to user-generated contents [174], lead to an urgent need for investigating the techniques in effectively retrieving user-generated contents.

Besides the tools for satisfying general information demands, e.g. a search engine, applications for specific domains can also benefit from Social Web data analytics to provide relevant information to the concerned parties. For example, marketing researchers consider the Social Web as a new communication channel between businesses and consumers [105], which allows for exploiting the *conversations* to timely harness the responses from consumers and apply countermeasures [179]. In addition to business cases, systems have been developed for public interests such as monitoring disease [143, 147], disseminating early warning of natural disasters [133], or post-analyzing the multimedia contents during mass events [125]. Building such kinds of systems not only shows the social value of these short messages but also poses scientific and engineering challenges to the researchers.

1.2 Objectives

In this thesis we investigate how to build a system to make the construction of Social Web data analytics simpler and how to utilize the analytical results can be utilized to fulfill manifold information needs. Based on reviewing the existing use cases of Social Web data analytics, we introduce a systematic platform solution, which includes both general purpose tools and domain-specific use cases. Twitter, as one of the most influential Social Web applications, has been selected as the main target of study in this thesis because of its pervasiveness throughout the world.

In the context of *information retrieval* on *Twitter*, we conduct analytics on microblog search from three different aspects, including *relevance*, *redundancy*, and *diversity*, with the aim of applying the results in a microblog search engine to improve the effectiveness of results. These objectives cannot be achieved by directly applying the existing methods for the Web due to the characteristics of short messages and search behavior on Twitter. Inspired by the task of real-time search on Twitter, we first propose a framework to combine various features as evidence to predict whether a tweet is relevant to a given query (*relevance*). Then we investigate duplicate content in mi-

croblog search results and propose different strategies to detect it. Next, we explore the various aspects conveyed by Twitter messages on a certain topic to gain a deep understanding of diversity in microblog posts, which not only means novelty – avoiding redundancy – but also the ability to resolve under-specified information needs. Finally, we put these results into practice and build a search engine for Twitter streams, to show not only the effectiveness of our analytical results but also the applicability of our platform solution.

Having defined and implemented our platform, we focus on the domain of crisis management and present a system (built on top of our platform) that supports stakeholders from public sectors and the general public during emergency or sensitive circumstances. The systems allows interested users to from an opinion about the important occurrences during an event, e.g. a festival or public holiday, a thunderstorm, a large fire, by filtering, enriching, and analyzing Twitter streams. Here, Twitter users act as so-called *social sensors* [133], providing a near real-time coverage of an event. Aggregating the individual users' tweets in a meaningful way can provide actionable insights for interested parties such as the police, the city council, the regional government, etc.

In summary, this thesis makes the following research contributions.

- **Social Web Data Analytical Platform.** We introduce the *Twitter Analytical Platform* (TAP) for conducting Twitter data analytics based on a survey of existing typical user cases. The platform provides a set of analysis tools that can be used to construct analytical workflows with a domain-specific language. Based on this platform, we build *Twinder*, a prototype search engine for Twitter streams, which serves as the target to which our analytical results for microblogging-based search can be applied.

- **Relevance Estimation for Microblog Search.** We propose a framework to expand the microblogging-based search queries with external knowledge. Then we combine it into another framework for extracting the features which are potentially predicative for estimating the relevance of a microblog post to a given topic. Finally, this enables us to analyze their importance and evaluate their impact on the retrieval effectiveness.

- **Near-Duplicate Detection for Microblog Search.** Based on the analysis of duplicate content in microblog search results, we set up a framework for extracting the syntactical elements, semantics, and

contextual characteristics and evaluate their effects on both detecting duplicates and determining the severity of the duplication.

- **Diversity Analytics of Microblog Search Results.** We present a methodology for building a corpus for diversification of microblog search and analyze the diversity characteristics in microblog search results.
- **Information Exploration System for Social Web Streams.** We apply our analytical platform and utilize the provided tools to build *Twitcident*, which is an information exploration system for Social Web streams and evaluate the efficiency of information seeking with a faceted search framework.

1.3 Research Questions

Social Web applications stimulate the prosperous development of social aspects in the Web 2.0 era and drive the popular participation of end-users to generate and exchange resources on the Web. Thanks to the ease of authoring microposts, the microblogging platforms like Twitter and Sina Weibo have become highly influential Social Web applications. Investigations and explorations based on data from these microblogging sites have become so active that more and more value and research possibilities behind the microblog posts have been identified. We have noticed the potential of microblogging messages in fulfilling information needs under either a general setting or more restrictive circumstances [26, 35]. Novel knowledge can be derived from the large volume of microblogging posts with predictive analytics supports. Lin et al. [100] detailed the efforts spent at Twitter to provide such kind of analytics. It relies on machine learning methods implemented in Pig (a higher-level language for the Hadoop platform) [122] to achieve scalability. This research field is naturally interdisciplinary [72] and there is an urgent need of engaging researchers from various domains, especially those who do not have a computer science background. Therefore, an easy to use, generic solution for typical applications to the problem of Social Web data analytics is valuable for the research community.

In the following we will make the contributions indicated in Section 1.1 concrete by listing the research questions that will be answered in this thesis.

- **Social Web Data Analytical Platform.** With structured data stored in relational database management systems, data analytics rely-

ing on a series of collection, extraction, and analysis technologies have been considered as a data-centric approach to provide business intelligence [34, 35, 168, 178]. The Social Web promotes content generation by end-users and thus brings new possibilities for conducting analytics for a wider range of application scenarios, including social-political patterns [97], discussions about celebrities, professional activities [95], and other activities in daily life. However, the characteristics of Social Web contents lead to the need for new analysis tools and orchestration frameworks. Therefore, we provide our solution to this problem by answering the following research questions.

  - What are the characteristics of Social Web data, which make analytics a non-trivial challenge?
  - What are the common core procedures across Social Web data analytics?
  - How can we accommodate essential procedures for Social Web data analytics in a scalable platform?
  - How can we efficiently build workflows for Social Web data analytics?

  In Chapter 2, the solutions to these questions are presented by a systematic solution to Twitter data analytics that allows for reusing a set of common analysis tools by programming in a domain specific language. Furthermore, we will show the efficiency of this solution by building a prototype search engine for Twitter streams, which can be enhanced by the analytical results from Chapters 3-5.

- **Relevance Estimation for Microblog Search.** Taking *information retrieval on Twitter* as the main context of research for providing relevant information, the most fundamental problem is to estimate the relevance of the tweets to the given topic. The classical approach is to apply mature information retrieval methodologies, e.g. relevance-based language modelling [77, 92], to retrieve a list of documents. In our solution, we take this method as part of the evidences available for relevance estimation and take advantage of the knowledge and predicative factors from various sources.

  - How can we enrich search queries on Twitter with background knowledge in order to better understand the meaning behind them?
  - Which micropost features allow us to best predict a micropost's relevance to a query?

  - How can we put our analytical findings into our prototype Twinder so that the overall retrieval effectiveness of the system improves?

  In Chapter 3 we will answer these questions and propose a query expansion framework and a framework that combines various features, including results derived from query expansion, to predict the relevance of tweets to the given queries. With a publicly available corpus, we present our analytical results to evaluate the importance of these features. Moreover, we will describe how we applied these results into Twinder, which is a search engine for Twitter streams proposed in this thesis, to improve the retrieval effectiveness.

- **Near-Duplicate Detection for Microblog Search.** According to previous quantitative investigations of the Twittersphere [91], 85% of the tweets are related to news. Trending topics are being discussed by Twitter users and it is reasonable to assume that there is considerable duplicate contents even when excluding retweets. The duplicate messages will decrease the novelty of search results and degrade the efficiency of seeking relevant information [18].

  - How much duplicate content exists in typical microblog search results?
  - How can we automatically detect the duplicate content along with the duplication level?
  - How does removing or aggregating duplicate contents affect the quality of the search results with respect to diversity?

  These questions will be answered in Chapter 4 by presenting a study of duplicate contents in Twitter search results and by proposing a near-duplicate detection framework for Twitter of which the effectiveness will be evaluated with a representative corpus.

- **Diversity Analytics of Microblog Search Results.** Decreasing redundancy in the search results makes space for more novel search results but does not necessarily mean diversity in a more general sense [40]. As of yet, there does not exist a microblog corpus for conducting research on search result diversification.

  - How can we build a microblog corpus for search result diversification?
  - How suitable is the corpus that we created for research on search result diversification?

  - To what extent can we achieve diversity by applying the developed de-duplication strategies?

  Chapter 5 provides answers to these questions and presents our efforts in building a microblog corpus for search result diversification. We then conduct comprehensive analyses to gain an understanding of diversity in microblog messages that are relevant to general topics.

- **Information Exploration System for Social Web Streams.** During crisis situations such as large fires, storms or other types of incidents, people nowadays report and discuss their observations, experiences and opinions in their Social Web streams. Recent studies show that data from the Social Web and particularly Twitter helps to detect incidents and topics [111, 133, 180] or to conduct analytics afterwards the information streams that people generated about a topic [60, 93, 130]. Automatically filtering relevant information about a real-world incident from Social Web streams and making the information accessible and findable in the given context of the incident are non-trivial scientific challenges. However, the engineering and evaluation of a system tackling these two problems has not been answered sufficiently by the literature yet.

  - How can we build an information exploration system with the Twitter Analytical Platform?
  - How well do the proposed strategies for information exploration perform in fulfilling the information needs?

  The answers to these questions will be given in Chapter 6 where we construct a system, relying on the analytical platform introduced in Chapter 2, for fulfilling the information needs from users during incidents and evaluate its performance in seeking relevant information.

1.4 Thesis Outline and Origin of Chapters

This thesis consists of seven chapters. After introducing the motivation of this thesis, the main contributions are presented in Chapters 2-6. For each of these chapters, we first describe the main research challenge and the corresponding research questions, continue with a dedicated background section, and summarize the main findings and contributions. The work in these chapters is based on multiple publications at workshops and conferences.

- **Chapter 2** is based on the paper published at the 2014 IEEE International Conference on Big Data (IEEE BigData 2014) [162].
- **Chapter 3** starts with work presented in a notebook paper published at the 20th Text REtrieval Conference (TREC 2011) [156] and continues with the work presented in the paper published at the 2nd workshop on Making Sense of Microposts[6] (MSM 2012) [158], where it won the *hypios*[7] *award for best 'innovation-related paper'*. The extended version of this work has been published at the 12th International Conference on Web Engineering (ICWE 2012) [157].
- **Chapter 4** is based on the paper published at the 22nd International World Wide Web Conference (WWW 2013) [159]. In addition, the work on this topic was in its then current stage presented at the PROMISE Winter School in 2013 as a poster, based on which a paper has been invited to be published in a tutorial book [160].
- **Chapter 5** contains findings that have been published at the 9th Asia Information Retrieval Societies Conference (AIRS 2013) [161].
- **Chapter 6** includes the works that have been published as a demo paper at the 21st International World Wide Web Conference (WWW 2012) [6] and a full research paper at the 23rd ACM Conference on Hypertext and Social Media (HT 2012) [5].

Finally, Chapter 7 concludes this thesis by summarizing the main findings and contributions made in this thesis and answering the research questions raised in Section 1.3. Furthermore, we provide an outlook of interesting research directions opened up by the work that has been done in this thesis.

[6] co-located with the 21st International World Wide Web Conference (WWW 2012)
[7] http://www.hypios.com

Chapter 2

Twitter Analytical Platform

In this chapter, we first conduct a survey of Social Web data analytics through typical use cases and abstract the common procedures into the Social Web data analytics pipeline. While the conceptual pipeline requires multiple functions, infrastructures, and the corresponding orchestration logic, we take Twitter data as the analysis target and propose a platform in which the specific workflows can be programmed in a domain specific language and thus executed. Finally, we present the agile implementation of a prototype search engine for Twitter streams with a small effort of coding. This prototype will be improved with additional analytical components that will be discussed in the following chapters. The contributions of this chapter have been published as [162].

2.1 Introduction

With the development of Web 2.0 technologies, Social Web applications, such as Twitter[1], have been attracting millions of users and media entities to share personal activities and publicize messages [91]. Given the immense amount of messages published on the Social Web every day, its popularity makes it an attractive source for conducting large-scale data analytics. Taking Twitter as the example, in the era of "Big Data" with emphasis on the 5 V's (Volume, Velocity, Variety, Value, and Veracity), the characteristics of *Volume* and *Velocity* in Twitter data analytics are represented by the hundreds of millions of messages posted every day and the TPS (Tweets per second) record broken

[1] http://twitter.com/, accessed July 30th, 2014

as important events are talked up with microblog posts[2]. The *Variety* comes both from combining the textual messages (limited to 140 characters) with the metadata and the need of integrating external knowledge (e.g. knowledge bases). The final two 2 V's, *Value* and *Veracity*, cannot be naturally obtained from Twitter data - human insights and ideas drive those two dimensions. Thus, deriving valuable and high-quality insights from Twitter data become non-trivial challenges.

In this chapter, we take Twitter as the analysis target mainly because of its openness. For the same reason, it has attracted numerous researchers to conduct analytics on various scenarios, ranging from sport events [128, 149] and natural disasters [175] to political elections [97, 115] and users' cultural characteristics [62]. Previous works [157–159] in the context of information retrieval for Twitter data focused on fulfilling the general information need of users with a list of ranked search results, enhanced by analytical results, including frameworks for relevance estimation and duplicate detection. Besides scientific contributions, applications have also been developed based on analytical results obtained from Twitter data. For instance, Sakaki et al. [133] have established an early warning system for earthquakes in Japan; later a similar system was also established in the United States Geological Survey [51]. Gao et al. [61] proposed a Twitter-based user modelling framework, which developers can leverage to build their personalized applications. Abel et al. [1] introduced a framework of adaptive faceted search, which leverages the semantics for efficient information exploration in tweets. In order to provide a systematic solution to content analysis tasks, IBM created the framework UIMA [66], which later became an Apache project, to analyze unstructured information with the aim of providing relevant knowledge to end users. However, there is to our knowledge not yet a dedicated solution for Twitter (or more generally microblog-based) data analytics that (i) allows for efficient customization of the tasks, (ii) with an extensible set of functionality, (iii) which can be employed both for research and application development purposes.

In this chapter, we tackle this challenge by answering the following research questions:

- What are the characteristics of Social Web data, which make analytics a non-trivial challenge?
- What are the common core procedures across Social Web data analyt-

[2]https://blog.twitter.com/2013/new-tweets-per-second-record-and-how, accessed July 30th, 2014

ics?

- How can we accommodate essential procedures for Social Web data analytics in a scalable platform?
- How can we efficiently build workflows for Social Web data analytics?

2.2 Background: Social Web Data Analytics

Nowadays, the data being generated on Social Web applications including Twitter, Facebook, Flickr etc., is of massive volume. The popularity of these services makes it possible to turn users into "Social Sensors" [133] for conducting analytics in different application scenarios, including commercial marketing, recommendation systems [3, 61], political elections [121], and, public infrastructures [5, 152]. However, one has to understand the characteristics of Social Web data before such kind of analytics can be implemented. The objective of Social Web Data Analytics is to provide users with "knowledge to act", in order to help them to make correct decisions. In this section, we will present a number of Social Web data characteristics, that make data analytics challenging.

As the content on the Web is created in a collaborative manner, a large number of applications and Social Web portals allow users to produce, consume and edit content as well as to vote and comment on other users' content. Thus, Social Web documents (of any type, e.g. image, video, audio, microblog message, blog post) are "rich" resources, with user-generated meta-data and signals allowing us to perform complex analytics.

Size and Ownership. Social Web portals have been growing quickly and often are continuing to grow; they may be generating hundreds of millions of items per day. For example as of August 2013, on an average day, 500 million tweets are being posted on Twitter, 1.6 million public photos are uploaded to Flickr and 500 terabytes of data are ingested into the Facebook database.

Not only size and growth make data collection difficult, the fact that most Social Web portals allow users and developers only very limited access to the data adds another dimension to the problem.

Unstructured Format. Before knowledge can be extracted from Social Web data, a lot of effort has to be expended on the refinement and transformation of the data [127]. This is not a simple process as multiple software

tools may be involved in the analytics pipeline. Previous studies [3, 4, 7, 57] have shown that more insights can be obtained by enriching the Social Web content.

Data Noise. The simplicity of authoring and the fact that large financial incentives exist for adversaries (to produce spam) lead to a substantial fraction of Social Web content to be of low quality. Given the collected data, one of the key challenges is to filter out the noise. Filtering can be based on simple manually defined rules, or rather complex NLP-based techniques.

2.3 Social Web Data Analytics Pipeline

Having analyzed the characteristics of Social Web Data, we now propose a Social Web Data Analytics pipeline in four steps: the analytics tasks follow a process of (i) collecting data, (ii) filtering the data, (iii) enriching the data with knowledge from other sources, and (iv) mining the refined data.

In order to orchestrate these different steps, they need to be connected in a pipeline. Additionally, the data needs to be translated into one generic data model. After the application of the analytics pipeline, the results can then be exploited to support secondary applications, such as interpreting the data through visualizations [128] or providing public sectors with real-time information during emergency situations [5]. Moreover, by integrating different components into our analytics pipeline, we can connect them together to make the analytical tasks more complex and more powerful.

To provide the reader with a better intuition of the proposed steps and their integration, we now discuss the major pipeline components in the context of three Twitter-based use cases:

- **User Profiling.** Previous works showed the feasibility to leverage Social Web activities for user modelling and personalization [61]. User profiles can be build from a user's semantically enriched tweets [4]. These user profiles in turn can then be employed for applications such as news recommendation [3].

- **Crisis Management.** During emergency circumstances, the information from Social Web streams have been shown to be good resources for interested parties (e.g. the police, the council, even news media) [91]. For example, the Social Web, Twitter in particular, broke the news when a US Airways plane ditched in the Hudson river on January 15th,

2009. It would be helpful for the concerning parties to collect these relevant information based on what the more comprehensive decision could be made.

- **Brand-building.** Major corporations have realized the importance of brand-building on Social Web applications. Apart from proactively engaging with customers through multiple channels, it is necessary to monitor public sentiment towards their products or services and to react accordingly [146].

  Therefore, it would be useful if one can quickly build an application for monitoring Social Web data streams and automatically categorize the information into different priority levels so that Public Relation efforts can be spend in a optimal way.

In the rest of this section, we will describe how these three cases fit into our Social Web Data Analytics pipeline.

2.3.1 Data Collection

Let us now discuss data collection issues within the context of the three introduced use cases.

User Profiling

When new users begin to use applications supported by the User Profiling module, their historical activities should be acquired from Twitter to build a model that is as accurate and complete as possible. As users keep using the application, we also need to monitor their latest activities as interests may vary over time [2].

Crisis Management

During emergency circumstances or cases where such possibilities may be expected, the basic information about an incident can be provided by either an emergency broadcasting system, e.g. P2000 in the Netherlands, or the predefined targets that we are concerned about. This may include several relevant keywords, the incident type, and possibly a geo-location. There are two ways to collect tweets that are potentially relevant to an incident:

- We can monitor the keywords that describe the incident and its type and the physical area around the incident (identified through geo-coordinates).
- With full access to historical data, we can build an index of Twitter messages' content; the potentially relevant tweets can be acquired by issuing a query against the index.

Having acquired the original tweets that may be relevant to the incident, we can further analyze their likelihood of relevance and assign them to different facets or categories.

Brand-Building

To monitor the tweets discussing a certain produce or service, we can use its name as a keyword or simply follow the account of the brand (including replies & direct tweets to this account). However, the tweets acquired may be much more than what want for brand-building purposes. Here the next phase of the pipeline becomes important: *data filtering*.

2.3.2 Filtering Social Web Data

The filtering of messages is an important component in a number of analytical tasks:

User Profiling

During the interest-based profiling of a user, we usually track the user by her ID via the Twitter Streaming API (see Section 2.6.1). This process however will not only provide us with her posting activities but will also include those messages that mention her. Here, filtering is simple: we retain all messages that are authored by the given user and remove the remainder from the stream.

Crisis Management

A huge volume of messages can be received from Social Web streams during an incident and they can describe multiple aspects of an incident. We can imagine that during incidents which receive a lot of attention some tweets

may be retweeted frequently. However, from the informativeness point of view, these retweets add little value for handling the incident. Therefore, we can filter them out.

Brand-Building

Many companies assign cool names to their produces or services, but frequently they may be easily confused with other entities of the same name. For example, Microsoft may want to monitor users' opinions about their Windows products after a new version release. However, "Windows" can refer to the operating systems distributed by Microsoft or a range of other concepts. By keyword matching, we collect all tweets mentioning "Windows" independent of the underlying semantics. To tackle this problem, we can use semantics extraction tools (see Section 2.3.3) to identify the concepts in tweets. Then we can rely on our filtering component to only retain the tweets that discuss the target concept.

2.3.3 Enriching Social Web Data

As Social Web data is often unstructured and noisy (see Section 2.2), we require approaches that enrich the data with further evidences (semantics). We can use existing techniques, including natural language processing, knowledge bases, Semantic Web resources etc., to extract valuable meta-data automatically. However, the specific method depends on the type and the characteristics of the data. In some cases, the results need a further normalization step before they can be used for analytics (in particular numerical data with units attached often requires normalization).

Unstructured textual raw data is difficult to exploit for analytics in case of Twitter, due to the severe length restrictions and the informal nature of most messages. With semantics identification tools, we can make better use of textual data by linking the concepts to a structured knowledge base. Aggregation tools can add summary results for complex data, especially from lists.

We now discuss the enrichment process with our three use cases in mind:

User Profiling

Given the Twitter messages' content posted by some user, we can extract named entities and topics to better understand the semantics of her Twit-

ter activities. For this purpose, we utilize Named Entity Recognition (NER) tools such as OpenCalais[3], which have been shown to work well for microblog messages [177]. This leads to semantically enriched documents whose identified semantic concepts form the basis for the user profile.

Crisis Management

In response to a crisis, Social Web users may talk about different aspects such as the reasons, locations, damages, casualties, etc. The identified concepts can be used to organize information facets and allow relevant parties to quickly zoom into the aspect that is most pertinent to them.

Brand-Building

Customers may talk about their experiences of not only using products, but also about purchasing, delivery and after-sale services. In large companies these different types of messages should be categorized according to the message type, the country of origin, the type of user, etc. Enriching tweets with semantics and identifying the related concepts in contents and metadata may help in this procedure. Moreover, brands may want to prioritize the messages to first review and respond to messages with a negative sentiment.

2.3.4 Mining Social Web Data

The core challenge of Social Web Data Analytics is to extract "knowledge to act". Data mining techniques are primarily designed to handle large-scale data, extract actionable knowledge, and gain insightful results. Therefore, we consider "Mining Social Web Data" as the last phase of Social Web Data Analytics. The data collection, filtering, as well as enrichment can be considered as the preparation for the eventual mining process.

User Profiling

User profiles can be derived without a specific application in mind [61]. However, they are most useful in practice, when employed for a specific task, such as advertisement targeting. In this example, the user profile is used as input to a classification model which determines whether or not to show a particular ad to the user.

[3] http://www.opencalais.com, accessed July 30th, 2014

Crisis Management

The Social Web contents discussing an incident may provide information on different aspects. One can either manually define rules for categorization, or use classification algorithms to achieve the same. The latter is the only feasible approach for large-scale data sources, as manual rules can never capture all particularities of unstructured documents.

Brand-Building

The complaints from a user may have effects on a product's or service's reputation, depending on her influence on followers and the attractiveness of her messages. Therefore, it would be effective if the complaints can be categorized (classified) with respect to their priorities.

Having introduced the three use cases and analyzed the detailed requirements in each of the four steps in the proposed Social Web Data Analytics pipeline, we are going to provide a more concrete and complete solution in the rest of this chapter. We implement the 4-step pipeline model in the form of *workflows* that can be customized with a domain specific language and enabled by a set of tools.

2.4 Twitter Analytical Platform

Having analyzed the characteristics and key enabling technologies, we will now provide our systematic solution for conducting data analytics of Twitter data. Towards this end, we have designed TAP (the **T**witter **A**nalytical **P**latform), which allows us to develop customized analytical workflows with Twitter data. Our platform is open-source[4] and can be easily extended.

2.4.1 Architecture

The architecture of TAP is summarized in Figure 2.1. The analytical tasks are implemented as workflows that can be executed on the platform. The workflows rely on the tools provided in the *TAP Functionality Stack*, which features data collection, filtering, enrichment, and mining capabilities. The workflows can be programmed in the domain-specific language TAL (**T**witter

[4]https://github.com/ktao/tap/wiki

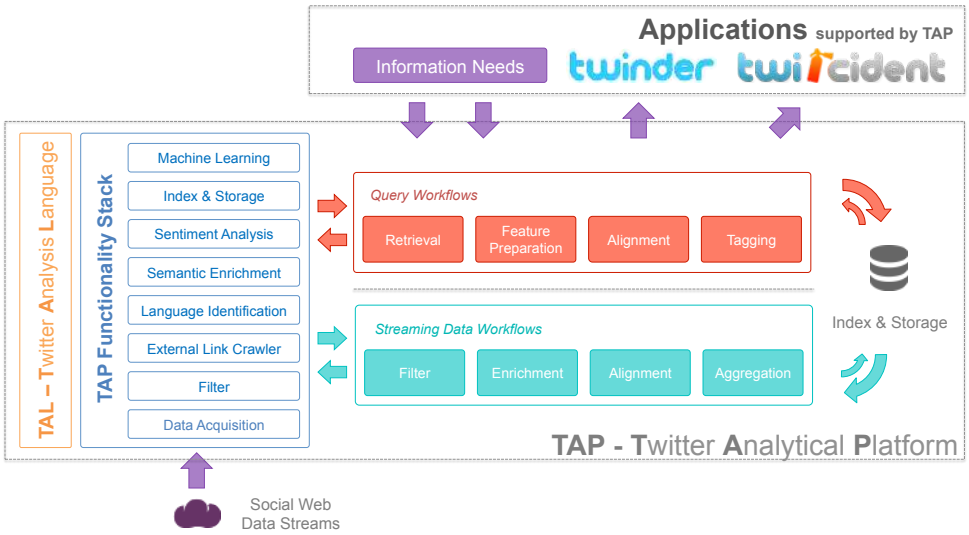


Figure 2.1: Architecture of Twitter Analytical Platform

Analysis **L**anguage), whose data model and syntax will be described in Section 2.5. With this language, one can select a set of analysis tools in the *TAP functionality stack* as we will discuss in Section 2.6. The currently supported analysis tools in the TAP functionality stack are listed in the blue block in Figure 2.1. We have designed a unified interface for these tools.

The intermediate results and historical data can be indexed and stored in an internal facility, which is depicted by the grey component in Figure 2.1. Currently, TAP relies on typical tools indexing and schema-free databases, which achieve good scalability, so that the Twitter data of large *Volume* and high *Velocity* can be handled – it provides the solution to the *Size* problem that we identified in Section 2.2.

2.4.2 Workflow Design

Our platform provides users with the freedom to create and customize their own analytical workflows. The workflows can be categorized into two types: (i) streaming workflows, and, (ii) query workflows. The categorization depends on the length of the acquired Twitter data (infinite or finite). Typical applications are supported by TAP in a hybrid mode, i.e. they rely on both types of workflows with various purposes. However, one can, of course, solely use one of them according to the requirements of the envisioned analytical task.

Streaming Workflow

The streaming workflows start with obtaining a Twitter data source from which Twitter messages arrive continuously. The workflows of this type are typically used for pre-processing the data collected from Twitter.

Here, some of the analysis tools can already be applied to the tweets before a specific information need is specified. These pre-computed intermediate results are stored in (and later served from) the internal storage.

For instance, in the use case of user profiling, one can implement a user modelling service with TAP by following the methodologies from our previous works [155]. The semantic enrichment of users' Twitter activities [4] can be implemented with a streaming workflow in TAP and the enriched Twitter messages will be indexed and stored for later user profiling, which can be implemented as a query workflow.

Query Workflow

The query workflows serve the information needs specified by users, e.g. a keyword search query, the sentiment over a brand or a new product to be monitored. The preprocessed (i.e. filtered and semantically enriched) Twitter messages can be fetched from the internal index & storage facility for data mining purposes. The necessary features are either fetched (if pre-computed) or generated on the fly and served to the selected learnt machine learning algorithms.

Following our user profiling example, a query workflow can be designed to compute her user profile. We collect her streamed activities and build the user profile with the user modelling strategy proposed in [61]. However, even if we have not been monitoring the user, we can still collect the user activities from external services and apply the same pre-processing tools in the streaming workflow. It should be noted that there is no restriction on the functionality depending on the workflow type. However, in this case, the efficiency of the user profiling service may be lower as the high-quality user profiles rely on the results from a chain of analysis tools.

2.5 Twitter Analysis Language

Having introduced the architecture and the workflow design, we now present TAL (*T*witter *A*nalysis *L*anguage), our domain specific language, with which

we can program the *TAL scripts* for customizing the various analytical workflows. The language will provide an interface for building the analytics workflows efficiently. In this section, we introduce the essentials of this language, including the data model and the syntax.

Note, that latest specification and example usages can be found at the development page of our Twitter Analytical Platform[5].

2.5.1 Data Model

TAP provides a unified data model to accommodate the source data, the intermediate results, and the output. Therefore it tackles the problem of *Unstructured Format* identified in Section 2.2. Given the fact that TAL is focused on data analytics with tweets, the core element in the data model is a single Twitter message. Thus, if we denote a single tweet as t with a subscript, the data model can be represented as follows.

$$t_1, t_2, t_3, \ldots, t_n, \ldots \tag{2.1}$$

as stream or

$$t_1, t_2, t_3, \ldots, t_n \tag{2.2}$$

as finite list.

The tweets arrive in the order indicated by the subscript. For both workflow types, one can utilize the available tools discussed in the TAP Functionality Stack to conduct data analytics.

The core element of the data model in TAL is the representation of a tweet. It has numerous attributes which can either be directly received from Twitter or derived from existing attributes through external services. The attributes can be either a value, a nested value, or a list of values. For example, a tweet that is received from the Twitter Streaming API looks as follows:

```
{"t1": {
    "text":"Pageview logs of Wikipedia are publicly
         available at http://t.co/WD9hNUmL5z , must be
         useful for some analysis. #RAMSS2013 #WWW2013",
    "source": "web",
    "author": {
        "username": "taubau",
        "id": 17730501,
        "created_at":"Sat Nov 29 07:47:38 +0000 2008",
```

[5] https://github.com/ktao/tap

```
        "statuses_count":797,
        "friends_count":369,
        "followers_count":160 },
    "created_at": "Tue May 14 19:16:20 +0000 2013",
    "id": "334386718419587100" }
    "hashtags":["RAMSS2013","WWW2013"],
    "language":"en",
    "urls":{ "http://t.co/WD9hNUmL5z":s
      "http://dumps.wikimedia.org/other/pagecounts-raw/"},
    ... (more attributes)},
... (more tweets),
 "meta": {
  "started_at": "Tue May 1 00:00:00 +0000 2013",
  ... (more meta information)}
}
```

Every single tweet in either the stream or the list is supposed to be a *tweet element*, as shown in the above data model. Besides the tweet elements, there is also the *meta element*, which contains a summary or global information about the whole data stream or list. This data model can accommodate various operations on the data acquired from supported sources, including filtering, enriching, and mining tools provided by the TAP functionality stack.

In TAL, each attribute has a data type, which can be one of the three supported types: (i) numeric, (ii) boolean, and (iii) string values. The numeric value can be integer or double values. Furthermore, the calculation can be performed between attributes or with immediate values of these three types. The supported operators are described next.

2.5.2 Syntax

The TAL scripts contain a series of *statements*. Each of them can specify an operation, such as collecting data from sources, making changes to the attributes of tweet elements or the meta element. There are two categories of statements in TAL:

General Operation The statement of *General Operation* type is for the overall operations to the target of analytics, including data collection, indexing, and storage.

Assignment The *Assignment* statements can create or modify the attributes of elements, including both tweet elements and the meta element.

When writing scripts in TAL, the keyword *this* always refers to the data that is currently being processed. The *General Operation* statements can

define the data source or invoke the indexing as well as the storing procedure as follows:

```
[General Operation](parameters)
```

Depending on the *General Operation* chosen, certain parameters can be passed. For instance, one can specify the data source to be analyzed as tracking the keyword "twitter" via the Twitter Streaming API with the following statement (see detailed usage in Section 2.6.1).

```
source.twitter.filter("twitter", null, null)
```

One can use the keyword *this* with a dot operator (.) to specify the attribute of all the tweet elements, or with an arrow operator ($\rightarrow$) to cite a particular element, especially the meta element. Given the method of specifying an attribute in an element, the general syntax of assignment operations is:

```
this.[attribute] := [method](parameters)
this->meta.[meta attribute] := "example"
```

The statements above describe two assignment operations: (i) to assign the specified *attribute* with the value derived by the *method* with required *parameters* (if applicable) and (ii) to set the value of a meta attribute to the string value "example".

Operators

TAL provides support of expressions with a set of operators. Depending on the data types or purposes, different sets of operators can be used.

Boolean Values The logical operators supported by TAL are ! (NOT), AND, as well as OR. The logical expressions can be connected to become a complex logical expression.

Numeric Values Besides the arithmetic operators, i.e. $+$, $-$, $*$, $/$, we can use relational operators, including $==$, $!=$, $>$, $<$, $>=$, and $<=$, to determine a boolean value.

String Values TAL provides the following operators for the data type of String to formulate logical expressions: $==$ (equal), $!=$ (not equal), $contains$, $startsWith$, $endsWith$. The length of a string value can be calculated by using the operator len.

Aggregation A number of aggregation operators are currently being supported by TAL, including *minimum*, *maximum*, *median*, *average*, *sum*, *count* and *cardinality*. The first five operators can only be used on numeric attributes, while the operators *count* and *cardinality* are used for counting the elements in a certain attribute or the distinct values for the attribute. Moreover, the aggregation operator can also be combined with a condition so that we can derive statistics of a subset of elements without removal of data.

Miscellaneous TAL provides built-in support for many operators that can be used in expressions, including (i) *overlap* for determining the overlap between two lists with given identifiers and (ii) *exists* for checking whether the given attribute exists for this tweet.

Delete One can remove the intermediate results or unnecessary content from the data with the *delete* operator, especially before moving the results in streaming workflows into the internal index and storage. This in turn reduces the spatial costs.

Analysis Functions

TAL relies on the analysis tools in the *TAP Functionality Stack* to construct or derive evidences for conducting analytics. These tools can be invoked from within TAL; detailed information on currently supported analysis tools can be found in Section 2.6.

Example

To provide the reader with a better intuition, we present a number of brief examples showcasing the usage of operators in TAL.

```
this.lang := langid(this.text)
this.nURLs := count(this.urls, this.lang=="en")
this.hasURL := this.nURLs > 0
delete(this.nURLs)
```

The statements above specify the construction of a new boolean attribute `hasURL`. The value of this attribute depends on the evaluation result of the logical expression on the right side of the assignment statement. The logical expression means whether, for English tweets, the attribute `nURLs` as the result of the aggregation operator count is higher than zero. The English

tweets are marked with string "en" in the attribute of `lang` as given by the language identification tool referenced by *langid* in the statement. The result, as assigned to the attribute `hasURL`, is conserved for further analysis, while the other attribute `nURLs` as the intermediate result is then deleted. One would need to repeat this process for multiple times to formulate a list of attributes as evidences for her analytical tasks. Taking the attribute `hasURL` as example, we could make a hypothesis that this feature is quite influential in predicting whether a tweet is relevant to a given topic. We will present our analytics results on this in Chapter 3.

2.5.3 Implementation

To implement the Twitter Analysis Language, we employ the *Spoofax Language Workbench* [83], a platform for the development of textual domain-specific languages with state-of-the-art IDE support. Spoofax provides highly declarative meta-languages, which abstract over the implementation of language processors. This allows us to focus solely on the design of TAL. In particular, we provide a modular syntax definition of TAL in SDF3, name binding and typing rules in NaBL and TS [88], and a mapping to Java code in Stratego [23]. Given these declarative specifications, Spoofax derives an Eclipse editor plugin [176] for designing workflows in TAL and generating Java code from them. Figure 2.2 shows an example of transforming TAL script named "example.tal" in the upper editor into the resulting Java code "example.java" in the below editor. The generated Java code can be executed in TAP. We make the latest progress on the implementation of TAL publicly available[6].

2.6 TAP Functionality Stack

As mentioned in previous sections, the analytics workflows depend heavily on the analysis tools supported by TAP. In this section, we are going to introduce each of them with example usage in TAL.

2.6.1 Data Collection

The first step of creating a workflow is to acquire the data to be analyzed. As mentioned in Section 2.4.2, TAL supports processing both data streams and

[6] https://github.com/guwac/metaborg-tal

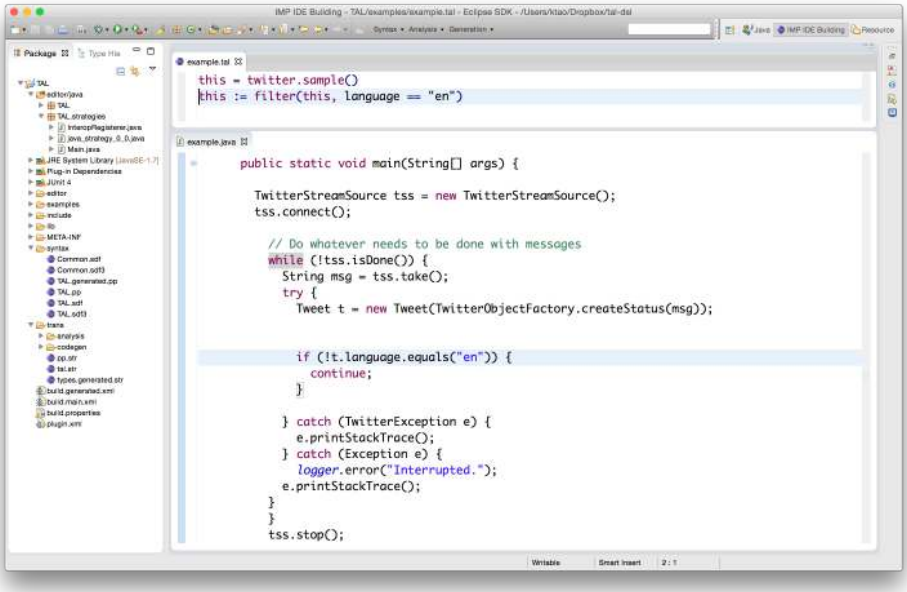


Figure 2.2: Eclipse editor plugin for TAL derived by Spoofax, transforming TAL script "example.tal" into Java code "example.java"

a tweet list of finite lengths. TAP provides access to the Twitter streaming API[7], the retrieval interface of an internal index and storage infrastructure, and is able to obtain previous tweets from the Twitter REST API[8].

Streaming Data

TAL supports acquiring real-time data from a public stream endpoint of the Twitter Streaming API. Specifically, users can use either a *filter* endpoint with parameters or a *sample* endpoint. Depending on the endpoint selected, the streaming data can be acquired with the general statement "twitter.[endpoint]". For instance, the following example specifies that we need to monitor the tweets either mentioning a keyword **twitter** or tagged with a geo-location within New York City.

```
twitter.filter("track=twitter&locations=-74,40,-73,41")
```

[7]https://dev.twitter.com/docs/api/streaming, accessed July 30th, 2014
[8]https://dev.twitter.com/docs/api/1.1, accessed July 30th, 2014

Search Result

Besides streaming data, a tweet list of finite length is often used as source to fulfill a specific information need. The most common practice is to take advantage of information retrieval tools to generate a ranked list of search results. As mentioned in Section 2.4.1, TAL maintains an internal index so that one can search with keyword queries. In order to obtain the search result list, one can fill the query as a parameter in the statement starting with the keyword *search*.

TAL can accept the search query that follows the query language specification of the preferred index. For instance, if the Indri search engine [186] is deployed as the indexing component in TAP, one can specify the search query with the Indri query language. Besides the original information given by Twitter, the search result given by TAL will not only provide all the preprocessing results in the persistent layer but also attach the query along with the retrieval score to the tweet. Therefore, a storage facility is needed to achieve this. One can configure the information to be stored so that it can be retrieved along with the search result. An example usage of this function is shown below:

```
search([search query])
```

Previous Tweets

Apart from the index, TAL maintains an internal storage for previous tweets. In this way, it allows us to retrieve a selection of stored tweets. For example, we can retrieve the tweets posted by a specific user, given her Twitter user ID:

```
twitter.select(author.id = 17730501)
```

2.6.2 Filter

With acquired data of either continuously arriving streams or lists of finite length, the effectiveness of selecting relevant information is limited by the functionality supported by the sources therefore further refinement is still needed for analysis in many cases. The solution to this problem is to first acquire a superset of the information needed and filter out the extra part afterwards. To achieve this, an essential approach is to filter on the attributes of tweets. For example, we can remove the non-English tweets if we focus

on analyzing English tweets as we did in some previous work [158, 159]. To achieve this in TAL, one can utilize the *filter* function, with a logical expression that can be evaluated on each tweet:

```
filter([logical expression])
```

The filter method will iterate over all the tweets and evaluate the logical expression. Only the tweets with an evaluation result of *TRUE* will be retained. We can leverage this function to deal with the *data noise* problem that we have noticed in Section 2.2.

2.6.3 External Link Crawler

Due to the length limitation of 140 characters, tweets are too short to accommodate the full stories so that frequently URLs are included. TAP provides support for the extraction of the main content in the resources referenced by these URLs. Moreover, the URLs in the tweets are often shortened by various services. We implement two functions to address these issues: *URL Expansion* and *Web Content Extraction*. Note that Twitter currently processes all URLs with the *t.co* URL wrapper and includes the `expanded URLs` in the URL entities of tweets. However, it will not expand the URLs shortened by a third-party service.

URL Expansion

Assuming that a list of URL entities are given in the field `urls` in every tweet, they can be expanded as follows:

```
this.urls.expandedurls := urlexpand(this.urls)
```

The URL expander will iterate over the URL entities in the list as specified by the attribute `urls` in every tweet and attempt to follow the links until it receives an *HTTP Success* response. The expanded URLs will be stored in the new attribute named `expandedurls` in every element of the list `urls`.

Web Content Extraction

Previous studies have shown the effectiveness of exploiting the linkage for inferring semantics from tweets [4]. The Web Content Extraction function

allows for extracting the main content that the URLs refer to. To use this function, we need to specify a list of URLs. As the result, a list of URL entities, including an attribute as specified for storing the content of the Web pages referenced by the URLs, will be returned. For example:

```
this.urls.content := extcrawl(this.urls)
```

The results will be added into the corresponding URL entities in the list as attribute `content`.

2.6.4 Language Identification

Given the text in the attribute, its language can be identified and a new attribute will be added to store the language identifier. As of March 2013, the language identification of original Twitter messages are provided via the streaming API[9]. However, we reserve the functionality in our design since it can be applied to other attributes as well.

For instance, the following command identifies the main language used in the `text` attribute; the results are stored in the `lang` attribute:

```
this.lang := enrich.langid(this.text)
```

2.6.5 Semantic Enrichment

Previous work has shown that semantics are meaningful for various analytical tasks, including understanding the user preferences [3], relevance of tweets to given topics [157, 158], and serving the modelling interface for applications such as recommender systems and digital tour guides [120]. TAP provides access to services like OpenCalais[10] (oc), DBpedia Spotlight[11] (dbp), AlchemyAPI[12] (alch), Zemanta[13] (zmt) and Wikipedia Miner[14] (wm), which can annotate the named entities mentioned in the input text. The results, in the form of a list of named entities, can be attached to the analysis object with the given attribute name. For example,

```
this.semantics := semantics.dbp(this.text)
```

[9] https://dev.twitter.com/docs/platform-objects/tweets, accessed July 30th, 2014
[10] http://http://www.opencalais.com/, accessed July 30th, 2014
[11] http://spotlight.dbpedia.org/, accessed July 30th, 2014
[12] http://www.alchemyapi.com, accessed October 6th, 2014
[13] http://www.zemanta.com, accessed October 6th, 2014
[14] http://wikipedia-miner.cms.waikato.ac.nz/, accessed July 30th, 2014

The statement above calls the DBpedia Spotlight service. The outcomes will be stored as a new attribute `semantics`. Moreover, semantic enrichment can also be applied to the external resources. This can be realized by specifying the attribute of the content crawled by the External Link Crawler.

2.6.6 Sentiment Analysis

TAP provides built-in support for categorizing the sentiment of Twitter messages. For example:

```
this.sentiment := sentiment(this.text)
```

The results of the sentiment analysis will be attached to the tweets as a new field. The result value can be: *positive*, *negative*, or *neutral*.

2.6.7 Index & Storage

TAP provides an internal index and storage facility, which allows us to retrieve tweets based on keyword or more complex queries that follow the given query language specifications. For example, the following statement asks TAP to store the tweets and add them into index:

```
store()
index()
```

The internal index and storage facilities are commonly used to preserve the preprocessed information that can be used for analytics at a later usage (again). Together with the *Data Collection* function (see Section 2.6.1) this setup provides a solution to the *Ownership* problem that we have identified in Section 2.2.

2.6.8 Machine Learning

In existing works, researchers have applied various machine learning algorithms for analytical tasks. TAP relies on Weka toolkit[15] to provide support for a wide range of classification and clustering algorithms.

[15] http://www.cs.waikato.ac.nz/ml/weka/, accessed July 30th, 2014

Classification

Classification algorithms are supervised learning methods. Hence, we need to specify the model derived from training data. Additionally, an attribute-feature mapping file that describes the relationship between the attributes in the TAP data model and the features in the learning model needs to be provided. Simple normalization operations can also be defined in the mapping file. Consider the following TAL example to this effect:

```
this.class := ml.classify(this,[MODEL],[MAPPING]}
```

The above statement classifies each tweet into categories according to the trained model. Assuming that the candidate tweets are readied for the classification, e.g. all evidences are included as attributes, one can pass the classification model file name as well as the attribute-feature mapping file to the function of *ml.classify*. The result will be assigned as the attribute `class` in each tweet. Given the classification results, one can use it as output or take is as evidence for further analyses.

Clustering

Clustering is a type of unsupervised learning method, hence a model file is not needed. However, one needs to specify the configuration parameters depending on the specific algorithm that is applied. For example, the clustering method can be used to cluster the documents according to their themes [29]. This allows us to diversify the microblog content in terms of subtopics:

```
this.theme := ml.cluster(this,[CONFIG],[MAPPING]}
```

The clustering result for each tweet is a cluster tag, which is assigned as an attribute. Therefore, we can diversify the content conveyed by the tweets by avoiding tweets in the same cluster.

2.7 Twinder Prototype

In this section, we showcase our solution to the problem of building a search engine for Twitter streams with TAP. We start with a prototype system and demonstrate how to accommodate the the analytics results [157, 159] into the prototype in the following chapters. The goal is to improve the quality of

the search results. With the use case of Twinder, we show that our platform allows us to build a search engine for Twitter streams with little effort. For instance, the streaming and the query workflows of the prototype system can be programmed in less than 10 lines of TAL scripts. The implementation of Twinder presented in this chapter has been made publicly available[16].

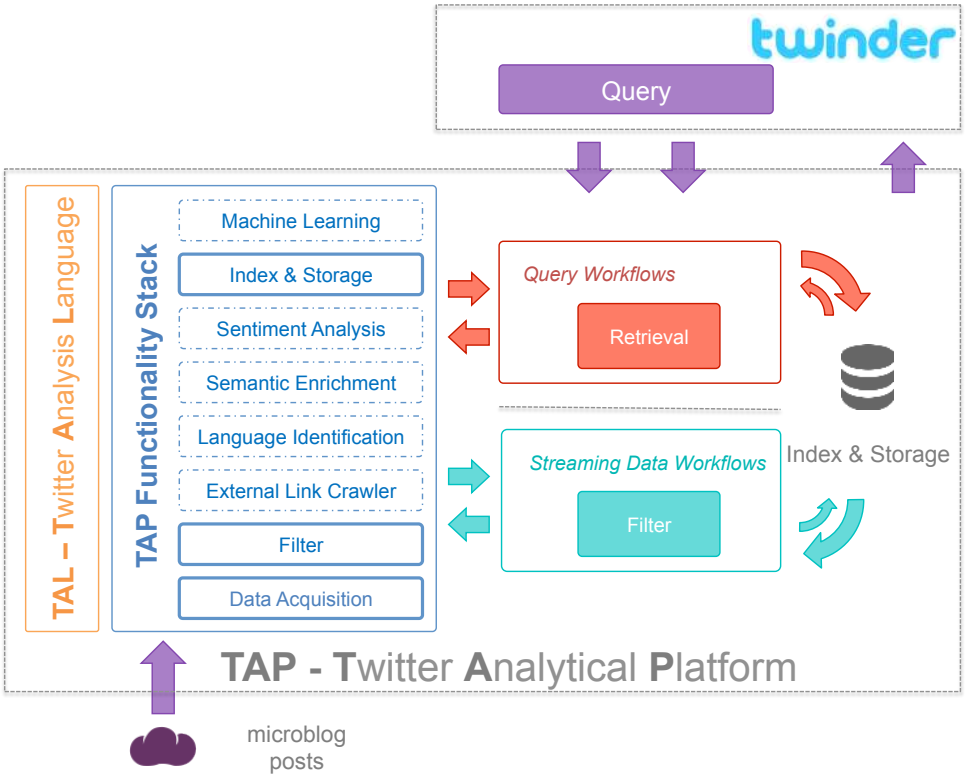


Figure 2.3: Twinder prototype architecture

We start building a search engine prototype for Twitter streams with two workflows: one streaming and one query workflow. Based on TAP, the architecture of Twinder is depicted in Figure 2.3. The streaming workflow obtains tweets from Twitter's public stream and adds them to the internal storage and index. The query workflow will be executed whenever a keyword query arrives. As seen from Figure 2.3, the prototype makes use of only a few tools provided in the functionality stack. A list of tweets will be returned from the internal index based on the retrieval score given by the information retrieval approach. Finally, Twinder will render the results given by the query workflow into a Web page of search results as shown in Figure 2.4.

[16]https://github.com/ktao/twinder-project

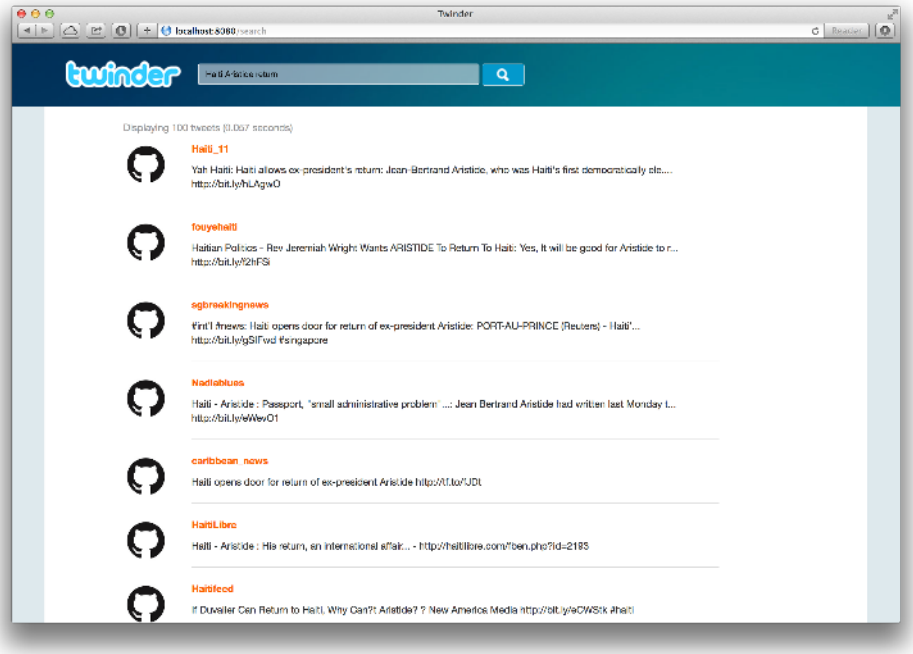


Figure 2.4: Twinder prototype screenshot

2.8 Discussion

In this chapter, we have introduced the **Twitter Analytical Platform**, which enables developers to conduct Twitter data analytics with various analysis tools. The specific analytical tasks can be customized by encoding them in the domain-specific **Twitter Analysis Language** which is also proposed in this chapter. We demonstrated the applicability of the TAP by building a prototype search engine for Twitter streams and integrated recently introduced techniques to improve its retrieval effectiveness. We summarize our contributions in Table 2.1.

Research Question	Summary of Contributions
What are the characteristics of Social Web data, which make analytics a non-trivial challenge?	▶ We summarized the characteristics of Social Web data, which make the analytics challenging due to the (i) immense size and limited accessibility, (ii) the unstructured data model, and (iii) the noisiness.
What are the common core procedures across Social Web data analytics?	▶ Based on three use cases, we proposed a four-step pipeline of Social Web data analytics, which includes (i) data collection, (ii) filtering, (iii) enriching, and (iv) mining Social Web data (Section 2.3).
How can we accommodate essential procedures for Social Web data analytics in a scalable platform?	▶ We presented *TAP* (*T*witter *A*nalytical *P*latform), a data analytics platform for Twitter that features support for both streaming and batch processing of Twitter data with scalable infrastructures (Section 2.4).
How can we efficiently build workflows for Social Web data analytics?	▶ We introduced the domain-specific language *TAL* (*T*witter *A*nalysis *L*anguage), which allows us to utilize TAP's set of analysis tools and in turn enables us to create and customize analytical workflows in a simple manner (Section 2.5 and 2.6). ▶ Based on TAP, we demonstrated the ability of our solution to perform data analytics tasks and support applications through an in-depth analysis of different implemented features aimed at the improvement of retrieval effectiveness in adhoc search settings (Section 2.7).

Table 2.1: Overview on research questions investigated in Chapter 2

Chapter 3

Relevance: Finding Relevant Microposts

Based on the platform introduced in Chapter 2 (which provides general supports for analytical tasks exploiting Social Web data), we now present common applications that can rely on the platform's functionality and orchestration abilities. We take search among microposts as the typical scenario and investigate the characteristics that make them relevant to a given topic. The contributions of this chapter have been published in [156–158].

3.1 Introduction

Microblogging sites such as Twitter or Sina Weibo[1] have emerged as large information sources for exploring and discussing news-related topics [91]. For instance, Twitter is used as a major platform for publishing and disseminating information related to various events[2], e.g. US presidential election in 2012[3] and Super Bowl in 2014[4]. Hence, searching for relevant information in microblogging services is challenging.

Among such immense amount of information, users who search for microposts about a certain topic typically perform a keyword search, as this is the standard way of accessing the Web today, made popular by Web search engines such as Bing and Google. However, Teevan et al. revealed that users

[1] http://www.weibo.com/, accessed July 30th, 2014
[2] http://yearinreview.twitter.com/en/tps.html, accessed July 30th, 2014
[3] https://blog.twitter.com/2012/election-night-2012, , accessed July 30th, 2014
[4] https://blog.twitter.com/2014/celebrating-sb48-on-twitter, accessed July 30th, 2014

exhibit a different search behaviour on Twitter compared to Web search [163]. For example, keyword queries on Twitter are significantly shorter than those issued for Web search: on Twitter people typically use 1.64 words to search while on the Web they use, on average, 3.08 words. This can be explained by the length limitation of 140 characters per Twitter message: as long keyword queries easily become too restrictive, people tend to use broader and fewer keywords for searching. Such shortness also leads to the problem of lack of semantics, so that the queries tend to be ambiguous or underspecified. Thus, searching within massive amount of microposts for tweets that are relevant to a given topic is a non-trivial research challenge.

To tackle this challenge, we focus on the fundamental problem of search on Twitter, which is to estimate the relevance of microposts to a given topic. Since 2011, the Text REtrieval Conference (TREC) has been organizing a specific track for microblog [99, 123, 145]. The main task in this track is defined as follows: given a keyword query Q and the timestamp t of Q, retrieve the *interesting and relevant tweets* for Q that are at least as old as t (to simulate Twitter's streaming nature). Inspired by the challenge introduced in this track, we aim at providing high-quality search results with respect to informativeness and coverage from different information sources, such as news media, experts with authority in related fields, and politicians. The corpus used for this track in 2011, as known as *Tweets2011*[5]. It contains 16 million tweets that cover a duration of two weeks starting from January 24th, 2011. The corpus was obtained by sampling the Twitter stream in order to create *a reusable, representative sample of the Twitter sphere* [113] (see Section 3.3.3). Apart from the corpus documents, TREC also provides 50 topics and their corresponding relevance judgements for evaluation purposes. Our studies of relevance estimation on Twitter search throughout the following subsections is based on this corpus.

In this chapter, we first consider the language gap between the queries and the Twitter messages as a core challenge. To overcome this problem, we introduce approaches that expand the original queries with keywords, which are more in line with the type of language people use on Twitter. In order to make the query rich in semantics, our approaches make use of the background knowledge from Linked Open Data [19] and external news articles due to their wide coverages of different topics. Then we further investigate what characteristics make a micropost relevant to a given query. By conducting extensive experiments with standard corpus, we verify the effectiveness of our methods.

[5] http://trec.nist.gov/data/tweets/, accessed July 30th, 2014

We formulate the task of query expansion as following.

Problem 1 (Query Expansion) *Given a query, the task of query expansion is to better understand the semantics in the original query and enrich it with potential knowledge from external sources so that relevant microposts that do not contain keywords from the original query can also be discovered.*

The expanded queries are the input to the retrieval system which matches them against the content of the indexed microposts. However, the language variation on Twitter, including the omission of vowels, the usage of abbreviations, and the repetition of letters [67, 172], may lead to low accuracy. Therefore, we develop another solution framework to consider all the factors that may influence relevance estimation.

Problem 2 (Feature-based Relevance Estimation) *Given a set of candidate microposts that are potentially relevant to a query, the task of feature-based relevance estimation is to determine which ones match the information need, based on a comprehensive set of micropost characteristics.*

With proper feature engineering, one can conduct various data analytical tasks with Social Web data. For example, Leskovec et al. [94] proposed a feature-based method to categorize the links in online social networks into positives (indicating relations such as friendship) and negatives (indicating relations such as opposition). Naveed et al. [119] analyzed a set of content-based characteristics in tweets that are indicative of a tweet's retweet likelihood. In this chapter, our relevance estimation framework employs a set of features that not only include classical retrieval scores, but also syntactic characteristics, semantics, and contextual information.

Having approached these two problems, we finally put the solutions into practice and achieve in better effectiveness in search results comparing to the prototype search engine for Twitter streams, Twinder as introduced in Section 2.7.

The research questions that we will answer in this chapter can be summarized as follows.

- How can we enrich search queries on Twitter with background knowledge in order to better understand the meaning behind them (see Problem 1)?
- Which micropost features allow us to best predict a micropost's relevance to a query (see Problem 2)?

- How can we put our analytical findings into our prototype Twinder so that the overall retrieval effectiveness of the system improves?

3.2 Related Work

Since the introduction of the Microblog Track at TREC 2011, numerous research efforts have been spent on microblog search. In the early attempts, researchers proposed solutions based on the existing retrieval methods that were proved to be effective for the normal Web search. For example, in this chapter we adopt the retrieval model of RM2 [92] in our proposed query expansion framework. Lv et al. [106] compared different estimations for the Relevance Models and showed that RM3 [77] is better and more stable. Since the temporal aspects are especially important to microblog search [52], the studies on temporal ranking of documents are meaningful for retrieving very fresh content. Therefore, researchers can benefit from the existing work on the retrieval models that emphasize the importance of temporal aspects. For example, Li et al. [96] introduced the time-based language models to incorporate the recency into the both query-likelihood models and relevance models. Dong et al. [48] presented their method for improving realtime search by detecting the fresh URLs from microblog streams. Efron et al. [53] showed that considering temporal information improved the retrieval effectiveness of a Twitter corpus.

As both documents and queries of microblog search are shorter than on the normal Web [163], it is intuitive to expand them with related information. Massoudi et al. [110] proposed a dynamic query expansion model for microposts retrieval, which incorporates the top terms representing the evolving language usage related to a given query. Bandyopadhyay et al. [14] tried to address the vocabulary mismatch problem by applying the query expansion methods that rely on the Web as the source of terms. However, relying solely on news articles turned out to be counter-productive for the retrieval effectiveness. McCreadie et al. [112] suggested that exploiting the hyperlinked documents improved retrieval effectiveness over using only the content of tweets or using the presence of a URL within the tweet as a feature.

Making sense of the semantics can benefit the applications that rely on microposts. For instance, Jadhav et al. [76] developed an engine that enriches the semantics of Twitter messages and allows for issuing SPARQL queries on Twitter streams. Abel et al. [4] introduced a method to improve the quality of users profiles by semantic enrichment of users' posting history on Twitter.

A straightforward way to extract semantics from tweets is to employ NER tools. However, it has been found that the performance of standard NER tools decreases severely on tweets [129]. Liu et al. [101] pointed out that there are two main issues that led to this: i) the amount of information that can be conveyed in 140 characters is limited; ii) there is a lack of training data. There has been many research efforts spent on NER specifically for tweets. In 2013, the MSM workshop[6] [27] organized an "Information Extraction" challenge to attract research efforts from the community to tackle this problem [55, 63, 171]. The early attempts of acquiring the training data were achieved by employing crowdsourcing platforms [58]. The NER techniques are also exploited to improve search effectiveness. For example, Guo et al. [68] presented a method to detect the named entities in a given query and classify them into predefined classes.

Learning to Rank has been widely applied to real-time Twitter search. For instance, Zhang et al. [190] presented a query-biased ranking model with considering the differences between queries, e.g. the unique expansion terms. Duan et al. [50] investigated features such as Okapi BM25 relevance scores or Twitter specific features (length of a tweet, presence or absence of a URL or hashtag, etc.) in combination with RankSVM to learn a ranking model for tweets. In an empirical study, they found that the length of a tweet and information about the presence of a URL in a tweet are important features to rank relevant tweets. In Section 3.4, we will re-visit some of the features proposed by Duan et al. [50] and introduce novel semantic measures that allow us to estimate whether a micropost is relevant to a given query or not.

To provide a user interface for consuming information conveyed by Twitter messages, Bernstein et al. [17] proposed an application that enables the exploration of tweets by means of tag clouds. However, their interface is targeted towards the browsing of tweets that have been published by users whom a user is following and not for searching the entire Twitter corpus.

3.3 Exploiting Background Knowledge for Search on Twitter

As mentioned in Section 3.1, we consider the language gap between the query and tweets as a core challenge of search on Twitter. For example, with a query such as "Roger Federer Wimbledon 2009", we expect to find very few

[6]Making Sense of Microposts (#MSM2013), co-located with the 22nd International World Wide Web Conference (WWW)

microposts that contain all query concepts. As human beings, we can identify the query contains the tennis player *Roger Federer* as well as the event of the Wimbledon Championships in 2009. It is reasonable to assume that the user would like to see information on the matches and results of Roger Federer's during the 123^{rd} Wimbledon Championships in the search results. However, matching such concepts with original keywords in the query will miss a lot of result items, because users may not use them while authoring their tweets but rather prefer to include abbreviations, nicknames etc. Having studied the topics of the TREC 2011 Microblog Track, we find this indeed to be the case for most queries: only 18 of the provided 50 search queries yield ten or more result tweets in the corpus when used as conjunctive queries.

To overcome this problem, we implement approaches that expand the original queries with keywords, which are more in line with the type of language people use on Twitter. This solution is inspired by Twitter-based user modeling methods [3] and aims to create a semantically rich profile for a query that is then translated into a weighted keyword query. Therefore, each query is modelled as a list of weighted concepts. The concepts, e.g. *ATP*, *Roger Federer*, *2009 Wimbledon Championships*, *Wimbledon*, *Sport*, and *Grand Slam*, are automatically derived with a Named-Entity-Recognition service from (i) news articles by mainstream media outlets, and (ii) from a set of phrases commonly used in tweets, so-called memes.

In this chapter, we will present a framework for constructing query expansion strategies with multiple choices in different design dimensions. We propose two strategies that exploit not only the semantics in queries but also concepts extracted from related tweets and news articles and verify their effectiveness by comparing them with the baseline line run, which is to search with adhoc queries, and the upper bound which is manually crafted by identifying relevant tweets ranked at the top.

3.3.1 Query Expansion Framework

An overview of our proposed query expansion framework is depicted in Figure 3.1. It visualizes our approach of query expansion by modeling it as a profile. We model each query as a *query profile* that should provide an accurate and comprehensive summary of the topic. When users submit a query to search for information, they are often not able to include the specific information nugget they are looking for. For example, a user's intention behind submitting a query such as "Roger Federer Wimbledon 2009" may be to learn more about the final between *Roger Federer* and *Andy Roddick*. If the user

was searching for Roger Federer's opponent, he would obviously not have added the concept "Andy Roddick" to his query. Our solution framework sets out to solve problems such as this one.

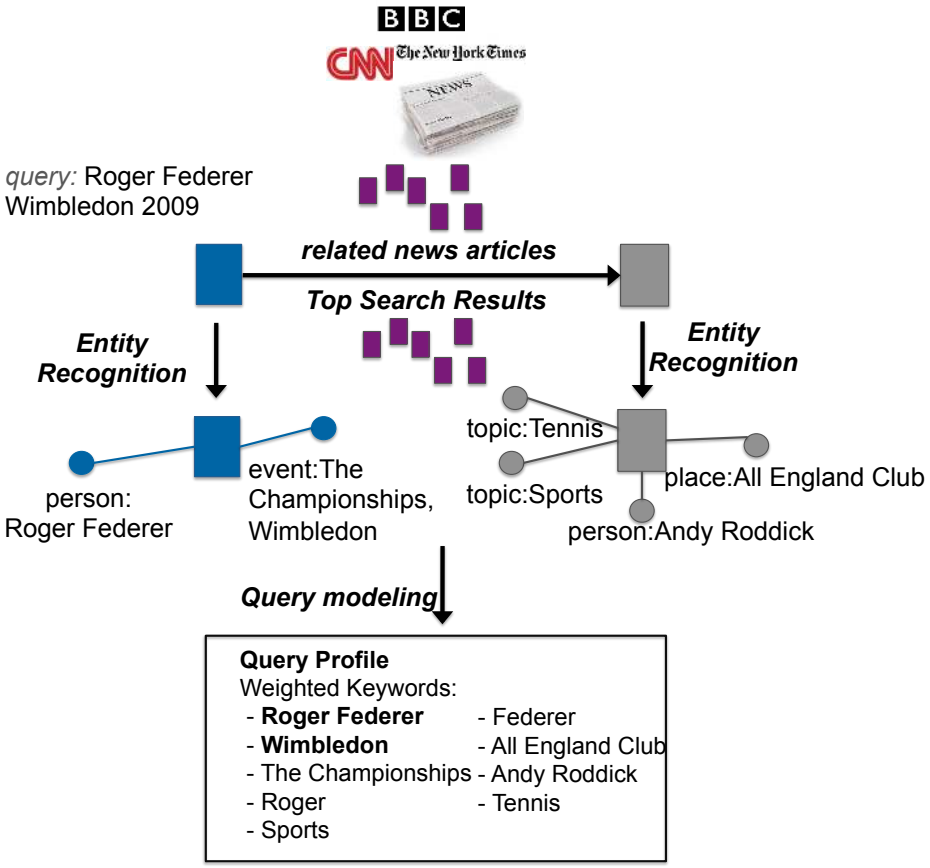


Figure 3.1: Overview of the query expansion framework

Query profiles are supposed to be expandable so that more information can be integrated when available. To specify the importance of an element in the profile, a weight is assigned to each element (see Definition 1). The elements in the profile are a set of concepts that are related to queries, which can be the words from the original queries, the labels of the entities detected by NER services, and the partial names or the synonyms that may be used to express the information needs in different ways.

Definition 1 (Query Profile) *The profile QP of a* query *Q is a set of weighted concepts c, determined by a set of strategies. Each concept has weight $w_s(c, Q)$, which is determined by the strategy s, and the query Q.*

$$QP_s(Q) = \{(c, w_s(c, Q))|c \in s(Q)\}$$

In the query profile, the additional keywords derived from knowledge concepts can be gained from (i) *semantic enrichment of queries*, (ii) *semantic enrichment based on related tweets*, and (iii) *related news articles*. By aggregating query profiles generated from 3 sources, the keywords will be combined together with varying importance weights depending on the query translation rules. The translated query can then be issued to the retrieval system to retrieve a ranked list of results.

Semantic Enrichment of Query

On average, the topics provided by the TREC 2011 Microblog Track contain 3.5 words (the shortest topic is "NSA"[7] and the longest one is "William and Kate fax save the date"[8]). Although these queries are longer than naturally occurring Twitter search queries as pointed out by Teevan et al.'s findings [163], they contain abbreviations, part of names, and nicknames. One example (see Table 3.1) is the event "Wimbledon 2009" (in the query "Roger Federer Wimbledon 2009"), which refers to the 123^{rd} Wimbledon Championships, an important tennis event. However, in tweets it may also be referred to as "Wimbledon". Similarly, Federer may be referred to by his nickname, e.g., "Federer Express", "Swiss Maestro". If these variants of a person's name and nicknames are considered when building a query profile, a wider variety of tweets can be found.

Topic	**[Roger Federer] [Wimbledon 2009]**	
Entity Name	**Annotated Text**	**Possible Concept Labels**
Roger Federer	Roger Federer	Roger, Federer, Roger Federer
2009 Wimbledon Championships	Wimbledon 2009	Wimbledon, 123^{rd} Wimbledon, 2009 Wimbledon, etc.

Table 3.1: Example of named-entity recognition and possible concepts in the topic

[7] Query with the identifier of MB006 in TREC 2011 Microblog Track
[8] Query with the identifier of MB018 in TREC 2011 Microblog Track

Semantic Enrichment with Related Tweets

Besides the semantics from the original query string, the information from other sources can also be utilized to further enrich the query profile. By searching Twitter with a query profiled by QP_q, we can retrieve the top-k "related tweets", $t_i (i = 1 \dots k)$, in the result list $R_t(QP_q)$[9]. Performing entity extraction on these tweets results in a set S_t, which contains a set of entities assignments $s(t_i)$ for each tweet t_i. As a result, we can derive a query profile $QP_{tenrich}(QP_q)$ by assigning a proper weight $w_{tenrich}(c_t, S)$ to the concept c_t. Intuitively, it should be determined by the importance of c_t in S. This query profile can be combined with other query profiles with a maximum weight in order to avoid the problem of query drift, which means the changes between the original query, which conveys the information need of users and its expanded form [192].

$$QP_{tenrich}(QP_q) = \{(c_t, w_{tenrich}(c_t, S_t)) | c_t \in \bigcup_{i=_1}^{k} s(t_i), t_i \in R_t(QP_q), i = 1 \dots k\} \tag{3.1}$$

$$S_t = \{s(t_i) | t_i \in R_t(QP_q), i = 1 \dots k\} \tag{3.2}$$

Semantic Enrichment with Related News

Since search queries on Twitter are often timely [163], i.e. related to current news, we also consider an external corpus of news articles as additional source of query enrichment. By maintaining an index of the latest news articles within a certain time period, we can retrieve the *related news articles*. Then similarly, we may derive a query profile $QP_{nenrich}(QP_q)$, enriched with concepts identified in news articles $n_i (i = 1 \dots k)$. Again, the weighting function determines the influence of the concepts based on their occurrences across all news articles S_n.

$$QP_{nenrich}(QP_q) = \{(c_n, w_{tenrich}(c_n, S_n)) \\ | c_n \in \bigcup_{i=1}^{k} s(n_i), n_i \in R_n(QP_q), i = 1 \dots k\} \tag{3.3}$$

[9]The subscript t means to search against the corpus of tweets.

$$S_n = \{s(n_i)|n_i \in R_n(QP_q), i = 1 \dots k\} \tag{3.4}$$

Query Profile Aggregation

Having identified additional named entities to enrich the original query with, we aggregate the entities into the query profile. To avoid query drift it can be beneficial to assign the greatest weights to the keywords found in the query originally submitted by the user.

For example, given two query profiles QP_1 and QP_2 and the corresponding weighting ratios of r_1 and r_2 respectively, the aggregated query profile is given as follows:

$$QP_{aggregated}(QP_1, r_1, QP_2, r_2) = \{(c_i, w_1(c_i) * r_1 + w_2(c_i) * r_2)| \\ (c_i, w_1(c_i)) \in QP_1, (c_i, w_2(c_i)) \in QP_2, i = 1 \dots n\} \tag{3.5}$$

where $r_1 > 0$, $r_2 > 0$, $r_1 + r_2 = 1$.

Query Profile Translation

With extra concepts, the query can be expanded to a more comprehensive version. Having created a query profile, in the final step we need to translate this profile into a query that can be interpreted by a retrieval system. Of interest to us are in particular retrieval systems that contain query languages which support the use of weighted keywords and concepts.

One example of such a retrieval system is Indri. In Indri's query language[10], entities consisting of more than one concept are treated as a phrase (using the #1(....) operator). As an illustration, consider the Indri query below, which was enriched with named entities extracted from the original query ("Roger Federer Wimbledon 2009").

```
#weight(
0.48000 #1(Roger Federer)
0.32000 #1(2009 Wimbledon Championships)
0.04000 #1(Roger)
0.04000 #1(Federer)
```

[10] http://www.lemurproject.org/lemur/IndriQueryLanguage.php, accessed July 30th, 2014

```
0.04000 #1(2009)
0.04000 #1(Wimbledon)
0.04000 #1(Championships)  )
```

All the concepts shown in the above example seem reasonable. However, there are cases in which the semantics are extracted incorrectly from the queries. To address this issue, we can apply strategies that assign higher weights to the concepts that are high in confidence. For example, the name *Roger Federer*, which has been identified by multiple NER services[11], is assigned the highest weight. The full name of the event *2009 Wimbledon Championships* is also considered as important as the annotated text "Wimbledon 2009" consists of more than one term. Other possible aliases are assigned a score as low as *0.04* to minimize the potential impact of query drift.

3.3.2 Query Expansion Strategies

Having introduced the tools for customizing the query profiling process, we can build pipelines for strategies to expand queries with related concepts from different sources. We propose two query expansion strategies as follows.

SEQwSRT. The strategy *SEQwSRT* stands for **S**emantic **E**nriched **Q**ueries **w**ith **S**emantics from **R**elated **T**weets. Figure 3.2 shows an example of this query expansion strategy that combines semantics extracted from the original query with the semantics from related external resources. In the case of SEQwSRT, the semantically enriched query profile will be aggregated with a query profile generated from related tweets, which are considered to be the *related external resources*. The importance of the two query profiles are adjusted by the weights r_1 and r_2. In this way, we integrate the information from related tweets in order to learn how users talk about the topic of a query, which words they are using, which people they are referring to etc.

SEQwSRTN. Often, users write postings on Twitter about what is going on at the moment in the world. News articles contain details about the most important events. We expect to be able to benefit from searching for related news articles to find additional entities that can be included in the query profiles. Therefore, we propose another strategy *SEQwSRTN* (**S**emantic **E**nriched **Q**ueries **w**ith **S**emantics from **R**elated **T**weets and **N**ews) to further include concepts from related news articles. This means to further aggregate

[11]The applied NER services, which are supported by Twitter Analytical Platform (see Section 2.6.5), include AlchemyAPI, DBpedia Spotlight, OpenCalais, Zemanta.

the query profile generated by the strategy of SEQwSRT, possibly with different weighting configuration, with a query profile, containing concepts from related news articles.

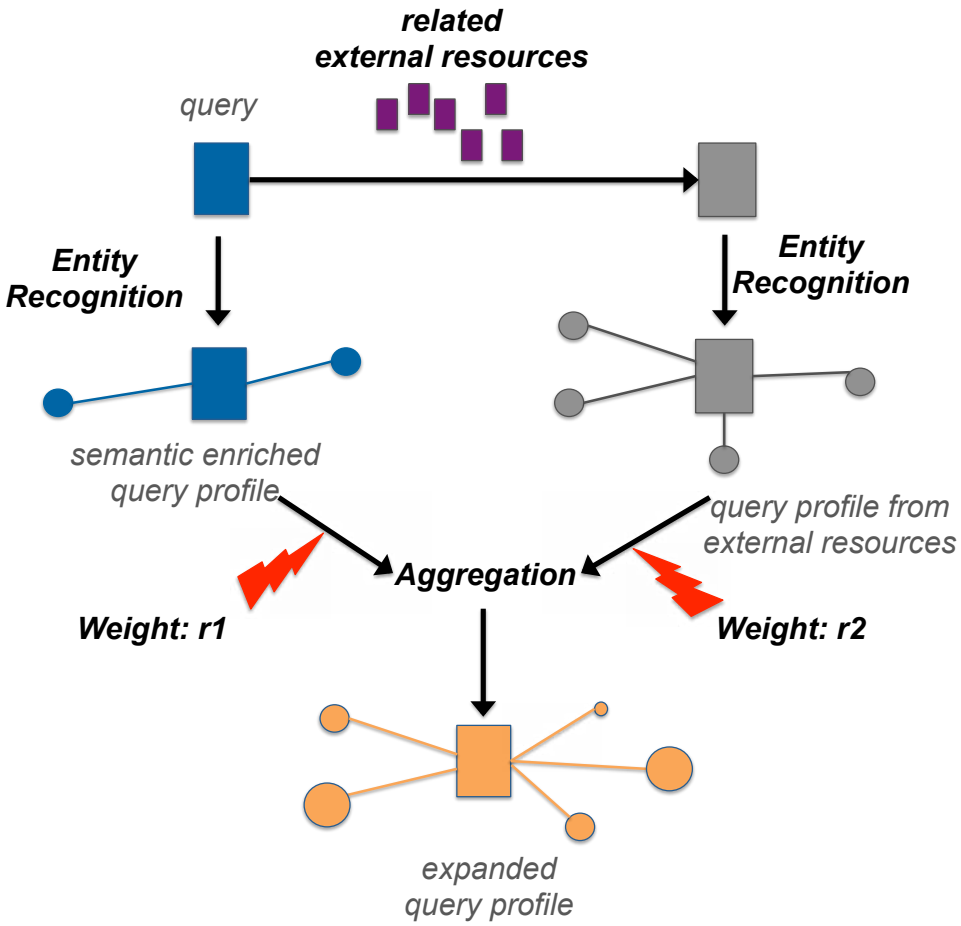


Figure 3.2: Example of Query Expansion Strategy

3.3.3 Evaluation of Query Expansion

To assess the effectiveness of our query expansion approach on Twitter, we evaluate the strategies proposed in Section 3.3.2 by comparing them with a baseline strategy that does not employ query expansion, as well as a manually created result as the upper bound.

Dataset Description

As mentioned in Section 3.1, we make use of the Tweets2011 corpus for this evaluation. The original corpus consists of approximately 16 million tweets, posted over a period of 2 weeks (January 24 until February 8th, inclusive). Since over time, less tweets are available for public access, we were only able to crawl 15 million tweets (crawled in June/July 2011), of which nearly 5 million tweets were detected to be written in English. Employing NER [12] on the English tweets resulted in over 6 million named entities among which we found approximately 0.14 million distinct entities.

The external news corpus was derived by extracting articles from 62 RSS feeds of prominent news media such as BBC, CNN, or the New York Times[13], from January 21st to February 10th, 2011. A precise overview of the numbers can be found in Table 3.2.

The 49 search queries are adopted from TREC 2011 Microblog Track[14]. Besides, 40,855 relevance judgments are used for evaluation purposes. The relevance judgments adopt a three-point scale, including i) *Not Relevant*, which means the tweet does not provide any useful information, ii) *Relevant*, which indicates the tweet at least some useful information on the topic, and iii) *Highly Relevant*, which suggests the tweet either contains or links to highly informative content [123].

Comparing Strategies

We compare our proposed strategies against: (i) baseline strategy without semantic query expansion and (ii) the manually created result list, which serves as upper bound.

Baseline With a separate index created for each query that includes all tweets up to the query's time stamp only (to avoid dealing with corpus statistics that are computed over future tweets), we retrieve up to 1000 results for each query. We filter out the non-English tweets, re-tweets, tweets with less than 100 characters, tweets with less than 50 characters if URLs are ignored, and tweets with words that contain a single letter three or more times in

[12] In this case, we used OpenCalais - http://www.opencalais.com/, accessed October 6th, 2014.

[13] The datasets, including the RSS feeds and the news articles, have been made publicly available at http://ktao.github.io/phd/#datasets.

[14] 50 queries were proposed for the TREC 2011 Microblog Track; one query had to be dropped as none of the participating system retrieved a single relevant document for it.

Corpus	#Elements
Crawled Tweets2011 Corpus	14,958,450
English Twitter Corpus	4,766,901
Entities extracted from Twitter Corpus	6,193,060
Queries	49
Relevance Judgment	40,855
(Tweet, Query) Pair Judged as Relevant	2,825
(Tweet, Query) Pair Judged as Non-Relevant	37,349
RSS News Feeds	62
News Articles	13,959
Entities extracted from News Articles	357,559

Table 3.2: Statistics of Tweets2011 dataset

sequence (e.g. "ooooooooh" or "aaaaaaah"). Language modeling with relevance model RM2 [186] was employed for the retrieval process. This approach makes use of pseudo-relevance feedback, i.e. additional terms are extracted from the top-k retrieved documents. In contrast to our strategies though, semantics do not enter the model, the expansion terms are chosen based on computed term-document statistics.

Manual The manual strategy was created by manually processing each query for 5 minutes as follows: the manually formulated queries were submitted them to a database[15]. By scanning the results of the English-tweet corpus sorted in descending order of tweet time, relevant tweets were marked. Tweets being duplicates of already marked tweets were subsequently ignored. We did not follow hyperlinks mentioned in tweets, only the tweet text itself was considered (though it was sometimes possible to determine the potential informativeness based on the URL itself, e.g. a link http://www.bbc.co.uk/.../ was considered more informative than http://bit.ly/rrxSt9). No external source (e.g. news articles) were used to learn more about a topic, potential new query concepts were learnt while scanning the tweets. Once the 5 minutes were up, the next topic was processed. On average, 20.8 tweets were

[15]Regular expressions were used, e.g. "%bbc%cut%" for searching the corpus for tweets relevant to the topic "BBC World Service staff cuts" (Query with the identifier of MB001 in TREC 2011 Microblog Track).

Strategy	Topics	P@30	MAP	>Median P@30	>Median MAP
Baseline	all	0.0993	0.1110	7	14
	high	0.0323	0.1207	3	11
SEQwSRT	all	0.3014†	0.2291†	25	29
	high	0.1051†	0.1999†	16	19
SEQwSRTN	all	0.1959†	0.1553	14	18
	high	0.0859†	0.1508	14	16
Manual	all	0.3946†	0.2719†	42	40
	high	0.1242†	0.2705†	22	24

Table 3.3: Experimental Results of Query Expansion. Statistically significant improvements over the baseline are marked with † (paired t-test, two-sided, $\alpha = 0.01$).

marked as relevant. The minimum was 0 (for the query of "organic farming requirements"[16]) and the maximum was 42 (for the query of "Egyptian curfew"[17]). The tweet time of the oldest manually marked tweet was recorded and one more query (also created by the annotator) was submitted; all tweets retrieved this way with a time stamp older than the manually selected ones were appended until a maximum of 1000 tweets was reached.

Experimental Results

The experimental results of the 2 proposed strategies (see Section 3.3.2) and our baseline as well as manual approach are reported in Table 3.3. The evaluation measures are Precision at 30 documents (P@30) and mean average precision (MAP). Both two measures were adopted as offical measures by TREC 2011 Microblog Track. On the one hand, the measure of P@30 tells us the fraction of relevant documents within the top-30 documents. On the other hand, MAP measures the quality across recall levels with one figure. The results for the 49 queries are shown in rows marked with *all*, while the subset of 33 queries that contain highly relevant tweets are shown in rows marked *high*. We also compared our query expansion strategies with the participants of TREC 2011 Microblog Track. The final two columns list the number of queries for which the P@30/MAP effectiveness of our query expansion strategies improved over the median P@30/MAP computed across all participants of the track [123].

[16]Query with the identifier of MB047 in TREC 2011 Microblog Track
[17]Query with the identifier of MB039 in TREC 2011 Microblog Track

The *Manual* strategy outperforms the automatic strategies on all measures by a considerable margin. This is not surprising, as the manual run involved a great amount of human effort. In 42 of the 49 queries, the manual result outperforms the median P@30 across all submitted runs. The fraction of queries that improve over the median decreases when considering only the query set with highly relevant tweets, a result which can be explained by the fact that the human annotator only considered the tweet text and possibly the URL string, but did not follow the hyperlinks present in the tweets. Having analyzed the provided relevance judgments, we find 81.9% of the relevant tweets to contain URLs. Among the automatic strategies, the strategy of *SEQwSRT* performs best, with $P@30 = 0.3$ (all) and $P@30 = 0.1$ (high). The strategy of *SEQwSRTN* has a less positive effect which may be caused by query drift introduced by related news articles. Furthermore, we have applied the two-sided paired t-tests to show the statistical significance of the effectiveness differences. Specifically, we have shown that the improvements achieved by the strategy of *SEQwSRT* are statistically significant in terms of both P@30 and MAP. Therefore, we can conclude that the retrieval effectiveness can be improved by expanding the queries with concepts that are linked to semantics extracted from queries and related tweets.

3.4 Feature-based Relevance Estimation

In the previous section, we proposed a framework for query expansion to incorporate relevant concepts in a query profile so that more potentially interesting tweets can be discovered. Based on the experimental results in Section 3.3.3, our query expansion strategy indeed improved retrieval effectiveness. However, we have found that concepts introduced by related news articles did not yield benefits in terms of retrieval effectiveness. Therefore, we seek to exploit the characteristics of microposts and combine them with the success that we achieved from last section. To this end, we consider the retrieval score given by the relevance model for either the original query or the expanded version can be seen as query-sensitive features of a $(tweet, query)$ pair. Besides these, we utilize other query-insensitive characteristics that differentiate between relevant tweets and non-relevant tweets. For example, more relevant tweets contain URLs compared to non-relevant tweets.

Hence, we can rely on the set of tweets whose relevance was judged by TREC assessors for our analysis and investigate query-sensitive as well as query-insensitive features. In this section we:

- present a set of strategies for the extraction of features from Twitter messages that allow us to predict the relevance of a post for a given query,
- take the success achieved in query expansion in Section 3.3 as a query-sensitive feature of a $(tweet, query)$ pair,
- analyze the features and characteristics of relevant and interesting tweets,
- evaluate the effectiveness of the different features for predicting the relevance of tweets to a query and investigate the impact of the different features on the quality of the relevance classification, and
- study to what extent the success of the classification depends on the type of queries (e.g. queries of short-term topic vs. queries of long-term interest) for which relevant tweets should be identified.

3.4.1 Features of Microposts for Relevance Estimation

In this section, we provide an overview of the different features that we analyze to estimate the relevance of a micropost to a given query. We present query-sensitive features that measure the relevance with respect to the query (keyword-based and semantic-based relevance) and query-insensitive measures that do not consider the actual query but solely exploit syntactic or semantic tweet characteristics. Finally, we also consider contextual features that, for example, characterize the creator of a micropost.

Query-Sensitive Features

A straightforward approach is to interpret Twitter messages as traditional Web documents and apply standard text retrieval measures to estimate the relevance of tweet to a given query.

Feature F1: keyword-based relevance score. To calculate the retrieval score for pair of (topic, tweet), we employ the language modeling approach to information retrieval [186]. A language model θ_t is derived for each document (tweet). Given a query Q with terms $Q = \{q_1, ..., q_n\}$ the document language models are ranked with respect to the probability $P(\theta_t|Q)$, which according to the Bayes theorem can be expressed as follows.

$$P(\theta_t|Q) = \frac{P(Q|\theta_t)P(\theta_t)}{P(Q)} \propto P(\theta_t) \prod_{q_i \in Q} P(q_i|\theta_t). \tag{3.6}$$

This is the standard query likelihood based language modeling setup which assumes term independence. Usually, the prior probability of a tweet $P(\theta_t)$ is considered to be uniform, that is, each tweet in the corpus is equally likely. The language models are multinomial probability distributions over the terms occurring in the tweets. Since a maximum likelihood estimate of $P(q_i|\theta_t)$ would result in a zero probability of any tweet that misses one or more of the query terms in Q, the estimate is usually smoothed with a background language model, generated over all tweets in the corpus. We employed Dirichlet smoothing [186].

$$P(q_i|\theta_t) = \frac{c(q_i, t) + \mu P(q_i|\theta_C)}{|t| + \mu}. \tag{3.7}$$

Here, μ is the smoothing parameter, $c(q_i, t)$ is the count of term q_i in t and $|t|$ is the length of the tweet. The probability $P(q_i|\theta_C)$ is the maximum likelihood probability of term q_i occurring in the collection language model θ_C (derived by concatenating all tweets in the corpus). Due to the very small probabilities of $P(Q|\theta_t)$, we utilize $\log(P(Q|\theta_t))$ as feature scores.

Hypothesis H1: the greater the keyword-based relevance score (that is, the less negative), the more relevant and interesting the tweet is to the topic.

Based on the semantics that are extracted from the microposts, we calculate two further relevance features.

Feature F2: semantic-based relevance score. This feature is also a retrieval score calculated according to *Feature F1* though with an expanded version of the original query. Based on the experimental results presented in Section 3.3.3, we select the best-performing strategy *SEQwSRT* to expand the original query.

Hypothesis H2: the greater the semantic-based relevance score, the more relevant and interesting the tweet is.

Feature F3: isSemanticallyRelated. It is a boolean value that shows whether there is a semantic overlap between the query and the tweet. This feature requires us to employ Named-Entity Recognition on the query as well

as the tweets. If there is an overlap in the identified concepts then it is set to *true*.

Hypothesis H3: if a tweet is considered to be semantically related to the query then it is also relevant and interesting for the user.

Syntactic Features

Syntactic features describe elements that are mentioned in a Twitter message. We analyze the following properties:

Feature F4: hasHashtag. This is a boolean property that indicates whether a given tweet contains a hashtag. Twitter users typically apply hashtags in order to facilitate the retrieval of the tweet. For example, by using a hashtag people can join a discussion on a topic that is represented via that hashtag. Users, who monitor the hashtag, will retrieve all tweets that contain it. Teevan et al. [163] showed that such monitoring behavior is a common practice on Twitter to retrieve relvant Twitter messages. Therefore, we investigate whether the occurrence of hashtags (possibly without any obvious relevance to the query) is an indicator for the interestingness of a tweet.

Hypothesis H4: tweets that contain hashtags are more likely to be relevant than tweets that do not contain hashtags.

Feature F5: hasURL. Dong et al. [48] showed that people often exchange URLs via Twitter so that information about trending URLs can be exploited to improve Web search and particularly the ranking of recently discussed URLs. Hence, the presence of a URL (boolean property) can be an indicator for a relevant tweet.

Hypothesis H5: tweets that contain a URL are more likely to be relevant than tweets that do not contain a URL.

Feature F6: isReply. On Twitter, users can reply to the tweets of other people. This type of communication may be used to comment on a certain message, to answer a question or to chat. Thus, reply messages are more private oriented. Chen et al. [36] studied the characteristics of reply chains and discovered that one can distinguish between users who are merely interested in news-related information and users who are also interested in chatting.

For deciding whether a tweet is relevant to a news-related topic, we therefore assume that the boolean *isReply* feature, which indicates whether a tweet is a reply to another tweet, can be a valuable signal.

Hypothesis H6: tweets that are formulated as a reply to another tweet are less likely to be relevant than other tweets.

Feature F7: length. The length of a tweet (the number of characters) may also be an indicator for the relevance. We hypothesize that the length of a Twitter message correlates with the amount of information that is conveyed it.

Hypothesis H7: the longer a tweet, the more likely it is to be relevant.

For the above features, the values of the boolean properties are set to 0 (false) and 1 (true) while the length of a Twitter message is measured by the number of characters divided by 140 which is the maximum length of a Twitter message.

There are further syntactic features that can be explored such as the mentioning of certain character sequences including emoticons, question marks, etc. In line with the *isReply* feature, one could also utilize knowledge about the re-tweet history of a tweet, e.g. a boolean property that indicates whether the tweet is a copy from another tweet or a numeric property that counts the number of users who re-tweeted the message. However, in this thesis we are merely interested in original messages that have not been re-tweeted yet and therefore also only in features which do not require knowledge about the history of a tweet. This allows us to estimate the relevance of a message as soon as it is published.

Semantic Features

In addition to the semantic relevance scores described as *Feature F2*, we can also analyze the semantics of a Twitter message independently from the topic of interest. We therefore utilize again the NER services to extract the following features:

Feature F8: #entities. The number of named entities that are mentioned in a Twitter message may also provide evidence for the potential relevance of a tweet. We assume that the more entities can be extracted from a tweet, the more information it contains and the more valuable it is. For

example, in the context of the discussion about birth certificates we find the following two tweets in our dataset:

t_1: *"Despite what her birth certificate says, my lady is actually only 27"*
t_2: *"Hawaii (Democratic) lawmakers want release of Obama's birth certificate"*

When reading the two tweets, without having a particular topic or information need in mind, it seems that t_2 has a higher likelihood to be relevant to some topic for the majority of the Twitter users than t_1 as it conveys more entities that are known to the public and available on DBpedia. In fact, the entity extractor is able to detect one entity, *db:Birth_certificate*, for tweet t_1 while it detects three additional entities for t_2: *db:Hawaii*, *db:Legislator* and *db:Barack_Obama*.

Hypothesis H8: the more entities a tweet mentions, the more likely it is to be relevant and interesting.

Feature F9: diversity. The diversity of semantic concepts mentioned in a Twitter message can be exploited as an indicator for the potential relevance of a tweet. Here, we count the number of distinct types of entities that are mentioned in a Twitter message. For example, for the two tweets t_1 and t_2, the diversity score would be 1 and 4 respectively as for t_1 only one type of entity is detected (*yago:PersonalDocuments*) while for t_2 also instances of *db:Person* (person), *db:Place* (location) and *owl:Thing* (the role *db:Legislator* is not further classified) are detected.

Hypothesis H9: the greater the diversity of concepts mentioned in a tweet, the more likely it is to be interesting and relevant.

Feature F10: sentiment. Naveed et al. [119] showed that tweets which contain negative emoticons are more likely to be re-tweeted than tweets which feature positive emoticons. The sentiment of a tweet may thus impact the perceived relevance of a tweet. Therefore, we classify the semantic polarity of a tweet into positive, negative or neutral using sentiment analysis services (see Section 2.6.6).

Hypothesis H10: the likelihood of a tweet's relevance is influenced by its sentiment polarity.

Contextual Features

In addition to the aforementioned features, which describe characteristics of the Twitter messages, we also investigate features that describe the context in which a tweet is published. In our analysis, we focus on the *social context*, which describes the creator of a Twitter message, and investigate the following four contextual features:

Feature F11: #followers. The number of followers can be used to indicate the influence or authority of a user on Twitter. We assume that users who have more followers are more likely to publish relevant and interesting tweets.

Hypothesis H11: the higher the number of followers a creator of a message has, the more likely it is that her tweets are relevant.

Feature F12: #lists. On Twitter, people can use so-called *lists* to group users, e.g. according to the topics about which these users post messages. If a user appears in many Twitter lists then this may indicate that her messages are valuable to a large number of users. We thus analyzes the number of lists in which a user appears in order to infer the value of a user's tweets.

Hypothesis H12: the higher the number of lists in which the creator of a message appears, the more likely it is that her tweets are relevant.

Feature F13: Twitter age. Twitter was launched more than five years ago. Over time, users learn how to take advantage of Twitter and possibly also gain experience in writing interesting tweets. Therefore, we assume that the experienced users are more likely to share interesting tweets with others. We measures the experience of a user by means of the time which passed since the creator of a tweet registered with Twitter.

Hypothesis H13: the older the Twitter account of a user, the more likely it is that her tweets are relevant.

Contextual features may also refer to temporal characteristics such as the creation time of a Twitter message or characteristics of Web pages that are linked from a Twitter message. One could for example categorize the linked Web pages to discover the types of Web sites that usually attract attention on Twitter.

3.4.2 Features Analysis

In this section, we describe and characterize the Twitter corpus with respect to the features that we presented in the previous section.

Dataset

Again, we use the Tweets2011 data for the analysis (see Table 3.2). Since we are inspired by the features differentiating between relevant tweets and non-relevant tweets, the $(tweet, query)$ pair are categorized into two groups, including 2,825 judged as relevant and 37,349 judged as non-relevant.

Feature Characteristics

In Table 3.4 we list the average values of the numerical features and the percentages of true instances for the boolean features that have been extracted. Relevant and non-relevant tweets show, on average, different values for the majority of the features. As expected, the average keyword-based and semantic-based relevance scores of tweets which are judged as relevant to a given topic, are much higher than the ones for non-relevant tweets: -10.7 and -10.3 in comparison to -14.4 and -14.2 respectively (the higher the value the better, see Section 3.4.1). Similarly, the semantic relatedness is given more often for relevant tweets (25.3%) than for non-relevant tweets (4.7%). For the query-sensitive features, we thus have first evidence that the hypotheses hold (H1-H3).

With respect to the syntactic features, we observe that 81.9% of the relevant tweets mention a URL in contrast to 53.9% of the non-relevant tweets. Hence, the presence of a URL seems to be a good relevance indicator. Contrary to this, we observe that *hasHashtag* and *length* exhibit, on average, similar values for the relevant and non-relevant tweets. Given an average number of 2.4 entities per tweet, it seems that relevant tweets feature richer semantics than non-relevant tweets (1.9 entities per tweet). Furthermore, the semantic diversity, i.e. the distinct number of different types of concepts that are mentioned in a tweet, is more than 10% higher for relevant tweets.

As part of the sentiment analysis the majority of the tweets were classified as neutral. Interestingly, Table 3.4 depicts that for relevant tweets the fraction of negative tweets exceeds the fraction of positive tweets (4.9% versus 2.4%) while for non-relevant tweets it is the opposite (6.5% versus 10.7%). Given the average sentiment scores, we conclude that relevant and interest-

Category	Feature	Relevant	Non-relevant
keyword relevance	keyword-based	-10.699	-14.408
semantic relevance	semantic-based	-10.298	-14.206
	isSemanticallyRelated	25.3%	4.7%
syntactic	hasHashtag	19.1%	19.3%
	hasURL	81.9%	53.9%
	isReply	3.4%	14.1%
	length (in characters)	90.282	87.819
semantics	#entities	2.367	1.882
	diversity	1.796	1.597
	positive sentiment	2.4%	10.7%
	neutral sentiment	92.7%	82.8%
	negative sentiment	4.9%	6.5%
contextual	#followers	6501.45	4162.364
	#lists	209.119	101.054
	Twitter age	2.351	2.207

Table 3.4: The feature characteristics

ing tweets seem to be more likely to be neutral or negative than tweets that are considered as non-relevant.

The average scores of the contextual features that merely describe characteristics of the creator of a tweet reveal that the average publisher of a relevant tweet has more followers (*#followers*), is more often contained in Twitter lists (*#lists*) and is slightly older (*Twitter age*, measured in years) than the average publisher of a non-relevant tweet. Given these numbers, we gain further evidence for our hypotheses (H11-H13). Thus, contextual features may indeed be beneficial within the retrieval process.

3.4.3 Evaluation of Features for Relevance Estimation

Having analyzed the dataset and the proposed features, we now evaluate the quality of the features for predicting the relevance of tweets for a given query. We first outline the experimental setup before we present our results and analyze the influence of the different features on the effectiveness for the different types of topics.

Experimental Setup

To evaluate the effectiveness of our feature-based relevance estimation framework and to analyze the impact of the different features on the relevance estimation, we relied on logistic regression to classify tweets as relevant or

Features	Precision	Recall	F-measure
keyword relevance	0.3036	0.2851	0.2940
semantic relevance	0.3050	0.3294	0.3167
query-sensitive	0.3135	0.3252	0.3192
query-insensitive	0.1956	0.0064	0.0123
without semantics	0.3410	0.4618	0.3923
without sentiment	0.3701	0.4466	0.4048
without context	0.3827	0.4714	0.4225
all features	0.3725	0.4572	0.4105

Table 3.5: Performance results of relevance estimations for different sets of features

non-relevant to a given query.

Due to the small size of the topic set (49 queries), we use 5-fold cross validation to evaluate the learned classification models. For the final setup of the evaluation, all 13 features were used as predictor variables. As the number of relevant tweets is considerably smaller than the number of non-relevant tweets, we employed a cost-sensitive classification setup to prevent the relevance estimation from following a best match strategy where simply all tweets are marked as non-relevant. In our evaluation, we focus on the precision and recall of the relevance classification (the positive class) as we aim to investigate the characteristics that make tweets relevant to a given topic.

Influence of Features on Relevance Estimation

Table 3.5 shows the performances of the relevance estimation based on different sets of features. Learning the classification model solely based on the keyword-based or semantic-based relevance scoring features leads to an F-measure of 0.29 and 0.32 respectively. Semantics thus yield a better effectiveness than the keyword-based relevance estimation. By combining both types of features (see query-sensitive in Table 3.5) the F-measure increases only slightly from 0.3167 to 0.3192. As expected, when solely learning the classification model based on the topic-independent features, i.e. without measuring the relevance to the given topic, the quality of the relevance prediction is extremely poor (F-measure: 0.01).

When all features are combined (see *all features* in Table 3.5), a precision of 0.37 is achieved. That means that more than a third of all tweets, which our framework classifies as relevant and thus returns as results to the user, are indeed relevant, while the recall level (0.46) implies that our approach discovers nearly half of all relevant tweets. Since microblog messages are

Performance	Measure	Score
	precision	0.3693
	recall	0.4625
	F-measure	0.4107
Feature Category	**Feature**	**Coefficient**
keyword-based	keyword-based	0.1716
semantic-based	semantic-based	0.1039
	isSemanticallyRelated	0.9559
syntactic	hasHashtag	0.0627
	hasURL	1.1989
	isReply	-0.5303
	length	0.0007
semantics	#entities	0.0225
	diversity	0.0243
	negative sentiment	0.4906
	neutral sentiment	0.2270
	positive sentiment	-0.6670
contextual	#followers	0.0000
	#lists	0.0001
	Twitter age	0.1878

Table 3.6: The feature coefficients for the model trained across all queries

very short, a significant number of tweets can be read quickly by a user when presented in response to her search request. In such a setting, we believe such a classification accuracy to be sufficient.

Overall, the semantic features appear to play an important role as they lead to an effectiveness improvement with respect to the F-measure from 0.39 to 0.41. Similarly, the sentiment features allow for an increase of the F-measure. However, Table 3.5 also shows that contextual features seem to have a negative impact on the retrieval effectiveness. In fact, the removal of the contextual features leads to an effectiveness improvement in recall, precision and F-measure.

We will now analyze the impact of the different features in more detail.

One of the advantages of the logistic regression model is, that it is easy to determine the most important features of the model by considering the absolute weights assigned to them. For this reason, we have listed the relevant-tweet estimation model coefficients for all involved features in the last column of Table 3.6. The features influencing the model the most are:

- *hasURL*: Since the feature coefficient is positive, the presence of a URL in a tweet is more indicative of relevance than non-relevance. That means, that hypothesis H5 holds.

- *isSemanticallyRelated*: The overlap between the identified named entities in the topics and the identified named entities in the tweets is the second most important feature in this model, thus, hypothesis H3 holds.
- *isReply*: This feature, which is *true* (= 1) if a tweet is written in reply to a previously published tweet has a negative coefficient which means that tweets which are replies are less likely to be in the relevant class than tweets which are not replies, confirming hypothesis H6.
- *sentiment*: The coefficient of the positive and negative sentiment features are also strong indicators for estimating the relevance of a tweet which is in line with our hypothesis H8. In particular, the coefficients suggest that negative tweets are more likely to be relevant while positive tweets are more likely to be non-relevant.

We note that the keyword-based relevance score, while being positively correlated with relevance, does not belong to the most important features in this model. It is superseded by semantic as well as syntactic features. Contextual features do not play a significant role in the relevance estimation process.

When considering only the query-insensitive features, we observe that interestingness is related to the potential amount of additional information (i.e. the presence of a URL), the overall clarity of the tweet (a reply tweet may only be understandable in the context of the contextual tweets) and the different aspects covered in the tweet (as evident in the diversity feature).

Influence of Query Characteristics on Relevance Estimation

In all reported experiments so far, we have considered the entire set of topics available to us. We now investigate to what extent certain topic characteristics impact the effectiveness of relevance estimation and to what extent those differences lead to a change in the logistic regression models. Our ambition is to explore to what extent it is useful to adapt the configuration of the relevance estimation model to the particular type of search topic. We categorized the topics with respect to three dimensions:

- Popular/Unpopular: The topics were split into popular (interesting to many users) and unpopular (interesting to few users) topics. An example of a popular topic is *2022 FIFA soccer*[18] – in total we found 24. In contrast, topic *NIST computer security*[19] was classified as unpopular

[18]Query with the identifier of MB002 in TREC 2011 Microblog Track
[19]Query with the identifier of MB005 in TREC 2011 Microblog Track

as one of 25 topics.

- Global/Local: In this split, we considered the interest for the topic across the globe. The already mentioned query MB002 is of global interest, since soccer is a highly popular sport in many countries, whereas the topic *Cuomo budget cuts*[20] is mostly of local interest to users living in New York where Andrew Cuomo is the current governor. We found 18 topics to be of global and 31 topics to be of local interest.
- Persistent/Occasional: This split is concerned with the interestingness of the topic over time. Some topics persist for a long time, such as MB002 (the FIFA world cup will be played in 2022), whereas other topics are only of short-term interest, e.g. *Keith Olbermann new job* [21]. We assigned 28 topics to the persistent and 21 topics to the occasional topic partition.

Our discussion of the results focuses on two aspects: (i) the effectiveness differences and (ii) the difference between the models derived for each of the two partitions (denoted $M_{splitName}$). The results for the three binary topic splits are shown in Table 3.7. While the effectiveness measures are based on 5-fold cross-validation, the derived feature weights for the logistic regression model were determined across all topics of a split. For each topic split, the three features with the highest absolute coefficient are underlined.

Popularity We observe that the recall is considerably higher for unpopular (0.53) than for popular topics (0.41). To some extent this can be explained when considering the amount of relevant tweets discovered for both topic splits: while on average 67.3 tweets were found to be relevant for popular topics, only 49.9 tweets were found to be relevant for unpopular topics (the average number of relevant tweets across the entire topic set is 58.44). A comparison of the most important features of $M_{popular}$ and $M_{unpopular}$ shows few differences with the exception of the sentiment features. While sentiment, and in particular positive and negative sentiment, are among the most important features in $M_{popular}$, these features (in particular the negative sentiment) are ranked much lower in $M_{unpopular}$. We hypothesize that unpopular topics do not evoke strong emotions among users and thus sentiment features play a less important role.

Global vs. Local This split did neither result in major differences in the retrieval effectiveness nor in models that are significantly different from each

[20] Query with the identifier of MB001 in TREC 2011 Microblog Track
[21] Query with the identifier of MB030 in TREC 2011 Microblog Track

Performance	**Measure**	**popular**	**unpopular**	**global**	**local**	**persistent**	**occasional**
	precision	0.3702	0.3696	0.3660	0.3727	0.3450	0.4308
	recall	0.4097	0.5345	0.4375	0.4748	0.4264	0.5293
	F-measure	0.3890	0.4370	0.3986	0.4176	0.3814	0.4750
Category	**Feature**	**popular**	**unpopular**	**global**	**local**	**persistent**	**occasional**
keyword-based	keyword-based	0.1035	0.2465	0.1901	0.1671	0.1542	0.1978
semantic-based	semantic-based	0.1029	0.1359	0.1018	0.0990	0.0808	0.1583
	semantic distance	1.1850	0.5809	0.9853	0.9184	0.8294	1.1303
syntactic	hasHashtag	0.0834	0.0476	0.1135	0.0429	0.0431	0.0803
	hasURL	1.2934	1.1214	1.2059	1.2192	1.2435	1.0813
	isReply	-0.5163	-0.5465	-0.6179	-0.4750	-0.3853	-0.7712
	length	0.0016	-0.0001	0.0003	0.0009	0.0024	-0.0023
semantics	#entities	0.0468	-0.0072	0.0499	0.0107	0.0384	-0.0249
	diversity	-0.0540	0.1179	-0.1224	0.0830	0.0254	0.0714
	negative sentiment	0.8264	0.0418	0.6780	0.3798	0.0707	0.8344
	neutral sentiment	0.2971	0.2102	0.1695	0.2653	0.3723	0.0771
	positive sentiment	-1.0180	-0.3410	-0.7119	-0.6476	-0.6169	-0.6578
contextual	#followers	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
	#lists	0.0002	0.0001	0.0002	0.0001	0.0004	0.0001
	Twitter age	0.1278	0.2743	0.0477	0.2646	0.1588	0.2377

Table 3.7: Influence comparison of different features among different topic partitions

other, indicating that—at least for our currently investigated features—a distinction between global and local queries is not useful.

Temporal Persistence It is interesting to note that the effectiveness (all metrics) is considerably higher for the occasional (short-term) topics than for the persistent (long-term) topics. For topics that have a short lifespan, recall and precision are notably higher than for the other types of topics. In the learnt models, we observe again a change with respect to sentiment features: while the negative sentiment is an important indicator for occasional topics, it is among the least important features for topics that are more persistently discussed on Twitter.

The observation that certain topic splits lead to models that emphasize certain features also offers a natural way forward: if we are able to determine for each topic in advance to which theme or topic characteristic it belongs to, we can select the model that fits the query best and therefore further optimize the effectiveness of the feature-based relevance estimation.

3.4.4 Synopsis

We have introduced a feature-based relevance estimation framework that analyzes various features to determine the relevance and interestingness of microposts for a given query. In an extensive analysis, we investigated tweet-based and tweet-creator based features along two dimensions: query-sensitive features and query-insensitive features. We gained insights into the importance of the different features on the retrieval effectiveness.

Our main discoveries about the factors that lead to relevant tweets are as follows:

- The learned models which take advantage of semantics and query-sensitive features outperform those which do not take the semantics and query-sensitive features into account.
- Contextual features that characterize the users who are posting the messages have little impact on the relevance estimation.
- The importance of a feature differs depending on the query characteristics; for example, the sentiment-based features are more important for popular than for unpopular topics.

3.5 Relevance Estimation in Twinder

In Section 2.7, we introduced Twinder as a search engine for micropost streams. Having investigated the problems of query expansion in Section 3.3 and feature-based relevance estimation in Section 3.4, we now show how these works can be applied. By realizing the scientific outcomes from this chapter in TAL, we show that the value of our ideas in a real application and further showcase the ability of the Twitter Analytical Platform.

3.5.1 Twinder Architecture with Relevance Estimation

The updated system architecture of Twinder is shown in Figure 3.3. Implemented within *Twitter Analytical Platform*, there are two components accepting external information: (i) microposts pre-processing and (ii) query preparation.

In particular, the former component is continuously receiving microposts from Social Web Data streams. The query-insensitive features are extracted for all tweets in real-time and the results are subsequently made persistent in the storage facility. Besides, we maintain the index in the prototype system and consider the indexing as a part of the pre-processing component.

Once a query is submitted, the query-sensitive features will be computed after the query is expanded with enriched semantics. Then by combining these features with the query-insensitive features that have been computed in advance, the *Feature-based Relevance Estimation* component will classify the candidate tweets and label them accordingly.

The implementation details of main components are given in Section 3.5.2.

3.5.2 Implementation in TAL

In order to integrate Feature-based Relevance Estimation into Twinder, we need to create a streaming workflow that processes the microposts in real-time for later tasks, and define a workflow that is triggered by search requests.

Pre-Processing Microposts

In the following, we give a possible pre-processing implementation in TAL. The pre-processing component will detect the messages' language and transform the necessary data fields in parallel. After filtering out non-English

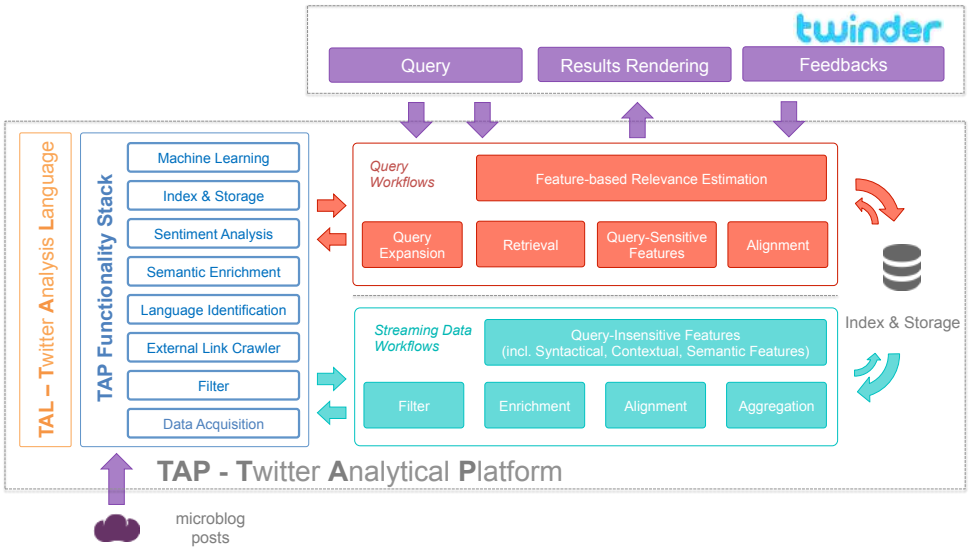


Figure 3.3: The architecture of Twinder with relevance estimation integrated

tweets, we make use of DBpedia Spotlight and a sentiment analysis tool to enrich microposts with semantics. Finally, Twinder appends the content of these microposts and stores the pre-processed results.

```
twitter.sample()

# add an attribute named lang to represent the language identification result
this.lang := langid(this.text)
filter := this.lang == "en" # filter out non-English tweets

this.semantics := semantics.dbp(this.text)
this.sentiment := sentiment(this.text)

store()
index()
```

Search for Microposts

Having computed and stored the pre-processed results, the query-insensitive features are ready. However, the query-sensitive features can only be determined after we receive search requests from users. The component *Query Preparation* will take care of this and provide them to the component *Feature-based Relevance Estimation*. A classification model will be preloaded into the component as well as a feature list description file. Hence, the relevance estimation results can be computed with the query-insensitive features from the storage and the query-sensitive features from the component of *Query Prepa-*

ration. Taking the query of "BBC World Service" as example, we implement this workflow as follows in TAL.

```
this.meta.query := "BBC World Service staff cuts"
# LM-based retrieval score, semantically enriched LM-score
search(this.meta.query)

# prepare the evidences
this.nEntity := count(this.semantics) # the number of entities
this.nURLs := count(this.urls) # the number of URLs
this.nHashtags = count(this.hashtags) # the number of hashtags
this.nEntityTypes := cardinality(this.semantics.type)
this.meta.query.semantics := semantics.dbp(this.meta.query)

# check the overlap between entities from query and entities from tweet
this.semanticOverlap := overlap(this.query.semantics,this.semantics)

# get the features ready
# keyword-based relevance is given by "source search" --> this.query.score

# semantic-based relevance is given by "source search" --> this.squery.score
this.isSemanticallyRelated := this.semanticOverlap > 0

# syntactic features
this.hasHashtag := this.nHashtags > 0
this.hasURL := this.nUrls > 0
this.isReply := this.replyTo != NULL
this.length := len(this.text)

# semantic features
this.positiveSentiment := this.sentiment == "positive"
this.negativeSentiment := this.sentiment == "negative"
this.neutralSentiment := this.sentiment=="neutral"

#contextual features
##followers given by this.author.followers_count --> specify in mapping file
##list given by this.author.listed_count --> specify in mapping file
# using unix-timestamp for calculating...
this.TwitterAge := (this.created_at - this.user.created_at) / 3600 / 24 / 365

this.query.relevance := ml.classify(this,RELEVANCE.model,RELVANCE_TAL.mapping)
```

Finally, the relevance estimation result is specified by the field of *this.query.relevance*.

3.5.3 Demonstration

We have implemented the feature-based relevance estimation in Twinder. Figure 3.4 depicts the search result page for the query of "Haiti Aristide return"[22].

[22]Query with the identifier of MB003 in TREC 2011 Microblog Track

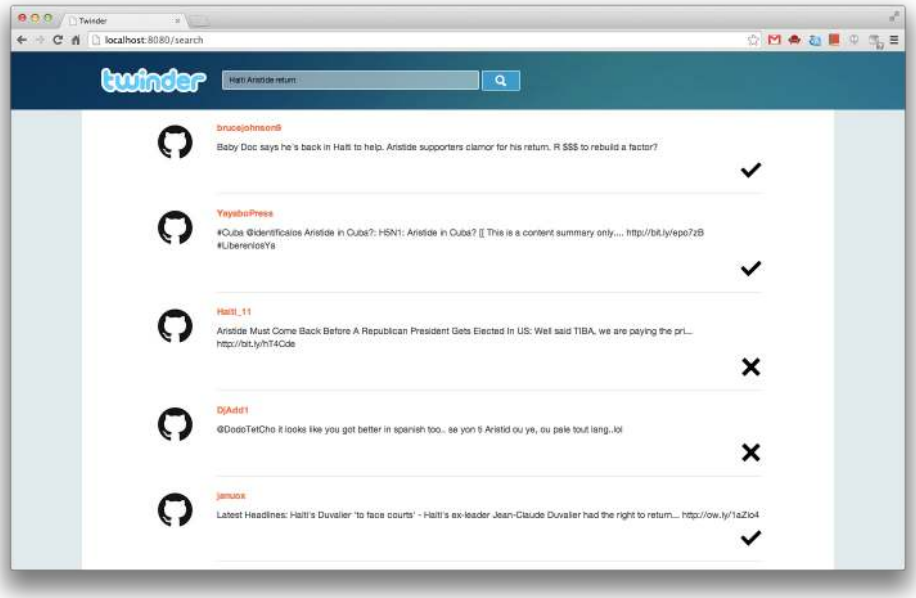


Figure 3.4: The search results rendered in Twinder with applying relevance estimation

The microposts with a check symbol are predicted as relevant while the results for the ones with cross are the opposite.

3.6 Discussion

To solve the problems and the research questions raised in the beginning of this chapter, we introduced and evaluated (i) the query expansion framework exploiting background knowledge and (ii) the feature-based relevance estimation framework. Further, we integrated the feature-based relevance estimation into Twinder, our search engine for Twitter messages. We summarize our findings in Table 3.8.

We have presented the query expansion framework that enriches queries with background knowledge. The suitability of semantic enrichment techniques adopted in such a framework has been validated through the improvement in the retrieval effectiveness. Specifically, one of our proposed strategies, which enriches the original query with background knowledge from Linked Open Data, performs significantly better than the original query. This result leads us to the conclusion that semantic enrichment is a tech-

Research Question	Summary of Findings
How can we enrich search queries on Twitter with background knowledge in order to better understand the meaning behind them?	▶ By exploiting the named entities in the query string and the concepts extracted from related tweets, we can build an enriched profile with which the retrieval effectiveness can be significantly improved.
Which micropost features allow us to best predict a micropost's relevance to a query?	▶ The semantics and query-sensitive features are influential in the estimation of the relevance between microposts and the given query. ▶ The length of tweets and the contextual features have little impact on the prediction.
How can we put our analytical findings into our prototype Twinder so that the overall retrieval effectiveness of the system improves?	▶ The design of our Twitter Analytical Platform enables scalability in different levels, including processing, storage, etc.; ▶ By implementing the Feature-based Relevance Estimation in TAL, we enhance the Twinder search engine.

Table 3.8: Overview on research questions investigated in Chapter 3

nique which should be investigated further in this setting. Future work will focus on the extraction of content from hyperlinks present in tweets, as our analysis has shown that the majority of relevant tweets contain URLs.

Furthermore, we have analyzed features that can be used as indicators of a tweet's relevance and interestingness to a given query and propose a feature-based relevance estimation framework. To achieve this, we investigated features along two dimensions: query-sensitive features and query-sensitive features. We evaluated the utility of these features with a machine learning approach that allowed us to gain insights into the importance of the different features for the relevance classification. Our main discoveries about the factors that lead to relevant tweets are the following: (i) The learned models which take advantage of semantics and query-sensitive features out-

perform those which do not take the semantics and query-sensitive features into account. (ii) The length of tweets and the social context of the user posting the message have little impact on the prediction. (iii) The importance of a feature differs depending on the characteristics of the queries. For example, the sentiment-based feature is more important for popular than for unpopular topics and the semantic similarity does not have a significant impact on the topics about entertainment.

The introduction of the feature-based relevance estimation approach is beneficial for search & retrieval of microblogging data and contributes to the foundations of engineering search engines for microposts. Hence, we put the scientific findings in this chapter into practice by integrating them into our prototype Twitter search system, Twinder, in the context of the Twitter Analytical Platform. In the following chapters, we will further analyze the redundancy and diversity in microposts.

Chapter 4

Redundancy: Near-Duplicate Detection for Microposts

In the context of information retrieval for Twitter data, in the last chapter we have presented our analytical findings that convey the importance of both query-sensitive and query-insensitive features to improve the search effectiveness. Having noticed the occurrences of duplicate content among results of microblog search, we are motivated to further explore the redundancy in Twitter data. By a preliminary investigation, we indeed have found that 20% of items within a ranked list of search results. However, this was not penalized by the evaluation metrics that were used in Chapter 3. Hence, in this chapter, we study and analyze the characteristics of duplicate microposts of different levels. The outcomes from such analytics may bring us deeper understanding of the duplication in Twitter. Further more it can inspire us to investigate the methods to detect the near-duplicates and identify their duplication levels. Similarly to Chapter 3, we enhance our prototype Twinder with the analytical results of near-duplicate detection presented in this chapter. The main contributions of this chapter have been published in [159, 160]

4.1 Introduction

On microblogging platforms such as Twitter or Sina Weibo, where the number of messages on influential events exceeds millions, solving the problem of information overload and providing solutions that allow users to access new information efficiently are non-trivial research challenges. Many of the mi-

croposts convey the same information in slightly different forms which puts a burden on users of microblogging services when searching for new content.

Traditional Web search engines apply techniques for detecting near-duplicate content [73, 108] and provide diversification mechanisms to maximize the chance of meeting the expectations of their users [126]. However, there exists little research that focuses on techniques for detecting near-duplicate content and diversifying search results on microblogging platforms. The conditions for inferring whether two microposts comprise highly similar information and can thus be considered near-duplicates differ from traditional Web settings. For example, the textual content is limited in length, people frequently use abbreviations or informal words instead of proper vocabulary and the amount of messages that are posted daily is at a different scale.

In this chapter, we bridge the gap and explore near-duplicate detection as well as its preliminary effects on redundancy of search results in the microblogging sphere. More specifically, we solve the problem that can be formulated as below.

Problem 3 (Near-Duplicate Detection) *Given a search result list of microposts, the task of near-duplicate detection is to automatically identify the items with contents of repetition, syntactically and semantically so that such messages could be filtered out or aggregated in the final search results.*

To tackle the problem, we conduct a study of micropost search results to get a deeper understanding of the duplicate contents on Twitter. As a result, we infer a model in which the duplication between microposts are categorized into 5 levels. We then apply techniques of syntactic, semantic, and contextual nature to engineer sets of features. These features are derived for individual pairs of tweets to determine their relationship to each other. We utilize machine learning algorithms to automatically identify the near-duplicate pairs and their level of duplication. Furthermore, we integrate this functionality into Twinder to demonstrate the validity of our work in real usage.

The main research questions answered in this chapter can be summarized as follows.

- How much duplicate content exists in typical microblog search results?
- How can we automatically detect the duplicate content along with the duplication level?

- How does removing or aggregating duplicate contents affect the quality of the search results with respect to diversity?

In Section 4.2, we introduce the related works on near-duplicate detection on Twitter. Then we will present our duplication model of microposts deduced from studying the dataset in Section 4.3. Based on that, we will propose our near-duplicate detection framework for microposts in Section 4.4. In Section 4.5, we will conduct extensive evaluations of our methodology. We will make preliminary analyses of its impact on the diversity of microblog search results. Finally, in Section 4.6 we describe the addition of the duplication detection component to Twinder.

4.2 Related Work

Near-duplicate detection technologies have been studied extensively for many types of contents. As driven by the prosperous development of the Web and search engines, most efforts have been spent on documents on the Web [74, 148]. Theobald et al. [164] targeted the problem incurred by the add-on portions on Web pages such as navigational bars and advertisements. Zhang et al. [189] showed the effectiveness of their method for detecting near-duplicate documents in both English and Chinese. Ture et al. [169] presented a solution to extract similar pairs of documents across two different languages. Koppula et al. [89] proposed a method to partially solve the problem by mining rules for de-duplication using only URL strings without fetching the content explicitly. The content reuse in online news articles [173] and blog entries [85] were studied as particular cases of Web documents. Besides extensive work on near-duplicate detection focusing on the Web, methods were also developed for other types of textual content, such as email [69], SMS [170] messages, and scientific publications [167]. As networking bandwidth increased and multimedia applications became popular, researchers developed approaches for detecting near-duplicate content among images [49] and videos [142]. From an algorithmic perspective, Mitzenmacher et al. [117] recently proposed efficient methods for high similarity estimation and Sundaram et al. [154] provided a solution to deal with large datasets.

A wide range of applications can benefit from near-duplicate detection algorithms. For instance, successful identification of near-duplicate content can reduce the costs of crawling, indexing, and storage [108]. The problem has been tackled most frequently on the pairwise level [74, 164, 169]. Similar effects can be achieved in a more efficient way by distinguishing the Web

mirrors site with the approaches of mining URL patterns [89]. The methods presented by Hajishirzi et al. [69] and Vallés et al. [170] for Email and SMS messages can potentially be used to fight spam given that such messages are mostly the same and sent *en masse* [87]. The approach of near-duplicate detection for scientific articles by Tsagkias et al. [167] can support identifying republished works or plagiarism. The identified structural similarity between Web pages [80] can be utilized for information extraction.

Among the numerous previous works on duplicate detection from which the Web search engines benefit, the most typical works are Broder et al.'s [25] shingling algorithm and Charikar's [33] random projection approach. Henzinger conducted a large-scale evaluation to compare these two methods [73] and Manku et al. [108] proposed to use the latter one for near-duplicate detection during Web crawling.

The duplicate detection algorithms proposed for textual corpora rely on the fingerprint generation [74, 148, 164] with characters [107] or document vector [87], the connectivity information between documents [46], document structure information such as anchor text and anchor window [70], or phrase usages [43]. Unfortunately, these methods are not suitable in a microblogging context due to the characteristics of microposts, which are mainly caused by the shortness and the authoring quality. Moreover, there is a lack of solutions to the problem of identifying near-duplicate content in the context of microblogs. In this chapter, we thus aim to bridge the gap and research near-duplicate detection on Twitter. We evaluate the impact of our method on the redundancy and its influence on the diversity of microblogging search results.

4.3 Duplicate Content on Twitter

In this section, we provide the outcomes of our study of duplicate content on Twitter. We present a definition of near-duplicate tweets in 5 levels and show concrete examples. We then analyze near-duplicate content in a large Twitter corpus and investigate to what extent near-duplicate content appears in Twitter search results.

All our examples and experiments utilize the "Tweets2011" corpus which is provided by TREC 2011 Microblog Track [113, 123].

4.3.1 Different Levels of Near-Duplicate Tweets

We define duplicate tweets as tweets that convey the same information either syntactically or semantically. Particularly, we distinguish near-duplicates in 5 levels.

Exact copy. The duplicates at the level of *exact copy* are identical in terms of characters. An example tweet pair (t_1, t_2) in our Twitter corpus is:

t_1 *and* t_2*:* `Huge New Toyota Recall Includes 245,000 Lexus GS, IS Sedans - http://newzfor.me/?cuye`

Nearly exact copy. The duplicates of *nearly exact copy* are identical in terms of characters except for *#hashtags*, *URLs*, or *@mentions*. Consider the following tweet:

t_3*:* `Huge New Toyota Recall Includes 245,000 Lexus GS, IS Sedans - http://bit.ly/ibUoJs`

Here, the tweet pair of (t_1, t_3) is a near-duplicate at a level of *nearly exact copy*.

Strong near-duplicate. A pair of tweets is *strong near-duplicate* if both tweets contain the same core messages syntactically and semantically, but at least one of them contains more information in form of new statements or hard facts. For example, the tweet pair of (t_4, t_5) is strong near-duplicate:

t_4*:* `Toyota recalls 1.7 million vehicles for fuel leaks: Toyota's latest recalls are mostly in Japan, but they also... http://bit.ly/dHOPmw`
t_5*:* `Toyota Recalls 1.7 Million Vehicles For Fuel Leaks http://bit.ly/flWFWU`

Weak near-duplicate. Two *weak near-duplicate* tweets either (i) contain the same core messages syntactically and semantically while personal opinions are also included in one or both of them, or (ii) convey semantically the same messages with differing information nuggets. For example, the tweet pair of (t_6, t_7) is a weak near-duplicate:

t_6*:* `The White Stripes broke up. Oh well.`
t_7*:* `The White Stripes broke up. That's a bummer for me.`

Low-overlapping. The *low-overlapping* pairs of tweets semantically contain the same core message, but only have a couple of common words, e.g.

the tweet pair of (t_8, t_9):

t_8: `Federal Judge rules Obamacare is unconsitutional...`
t_9: `Our man of the hour: Judge Vinson gave Obamacare its second unconstitutional ruling. http://fb.me/zQsChak9`

If a tweet pair does not match any of the above definitions, it is considered as *non-duplicate*.

4.3.2 Near-Duplicates in Twitter Search Results

The near-duplicates in Twitter search results lead to redundancy thus inefficiency in fulfilling information needs. Therefore, we need a better understanding about the severity of this problem by studying a representative dataset. Towards this end, we again make use of the Tweets 2011 corpus [113] from which the example tweets in Section 4.3.1 are selected. The introduction of this dataset can be found in Section 3.3.3.

In this dataset, TREC assessors judged the relevance between 40,855 topic-tweet pairs for 49 topics. A total of 2,825 topic-tweet pairs were judged as relevant. In other words, each topic on average has 57.65 relevant tweets. Employing Named Entity Recognition (NER) services on the content of these relevant tweets results in 6,995 and 6,292 entity extractions by using DBpedia Spotlight [114] and OpenCalais respectively. There are 1,661 external resources referred by the links mentioned in these relevant tweets, from which we further extract 35,774 and 56,801 entities respectively by using the two same services. For each topic, we manually labelled all pairs of relevant tweets according to the levels of near-duplicates that we defined in in Section 4.3.1. More specifically, a junior researcher first labeled the duplicate pairs according to the definitions given in Section 4.3.1, based on which two senior researchers checked the labeling and finally achieved an agreement ratio of 98.1%. The inconsistency was then resolved after a discussion among all three contributors for the annotation. In total, we labelled 55,362 tweet pairs. As a result, we found that 2,745 pairs of tweets are duplicates, 1.89% of them were labelled as exact copy and 48.71% of them were judged as weak near-duplicates (see Figure 4.1).

For each of the 49 topics, we ranked the tweets according to their relevance to the corresponding topic based on previous work [157] to investigate to what extent the ranked search results contain duplicate items. It is should be noted that the retweets have been removed before this ranking process.

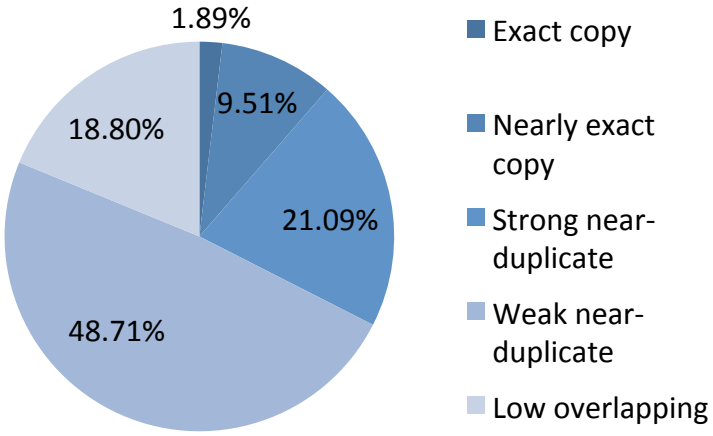


Figure 4.1: Ratios of near-duplicates in different levels

In the top 10, 20, 50 items and whole range of search results, we find that 19.4%, 22.2%, 22.5%, and 22.3% respectively are duplicates. Given that approximately one fifth of the items are duplicates, we consider duplicate detection an important step in the processing pipeline to improve the quality of the search results.

4.4 Duplicate Detection Framework

We consider the problem of duplicate detection as a classification task that can be performed in two steps: (i) deciding whether a pair of tweets is duplicate or not; and (ii) determining the duplicate level. For both steps, we rely on a collection of features that exploit syntactic elements, the semantics in both tweets and the content of the referred Web pages, as well as context information about tweets and users. Finally, we employ logistic regression classifiers to ensemble the characteristics from pairs of tweets into the detection of duplicates and the determination of the levels.

4.4.1 Features of Tweet Pairs

We now provide an overview of the different features that we extract from tweet pairs for the task of duplicate detection. Given a pair of tweets (t_a, t_b), four sets of features are constructed. In the following sections, we elaborate on the definition of the features and the hypotheses that led us to include them in our strategies.

Syntactic Features

We construct syntactic features by matching the tweet pairs with respect to their overlap in letters, words, hashtags and URLs.

Feature DF1: Levenshtein distance. This feature indicates the number of characters required to change one tweet to the other. Each change can be a deletion, insertion, or substitution. Hence, the Levenshtein distance evaluates the difference between a pair of tweets with respect to the usages of words, phrases, etc. As the furthest Levenshtein distance between a pair of tweets is $L_{max} = 140$ (the maximum length of a tweet), we normalize this feature by dividing the original value by L_{max}. Therefore, the final value of this feature is in the range of $[0, 1]$.

Hypothesis DH1: The smaller the Levenshtein distance between a pair of tweets, the more likely they are duplicates and the higher the duplicate score.

Feature DF2: Overlap in terms. This feature compares tweet pairs term by term. Although the tweets of near-duplicates use similar sets of words, the ordering of words may differ. Therefore we determine the overlap in terms between tweet pairs. In our implementation, this feature is measured by using the Jaccard similarity coefficient as following:

$$overlap(w(t_a), w(t_b)) = \frac{|w(t_a) \cap w(t_b)|}{|w(t_a) \cup w(t_b)|} \tag{4.1}$$

Here, $w(t_a)$ and $w(t_b)$ are the sets of words that are used in t_a and t_b respectively. As we use the Jaccard similarity coefficient to measure the overlap, the value of this feature is in the range of $[0, 1]$. Similarly, the following features that describe overlap in different aspects are measured by the Jaccard similarity coefficient.

Hypothesis DH2: The more overlap in terms we find between a pair of tweets, the higher the duplicate score.

Feature DF3: Overlap in hashtags. Hashtags are often used by users in tweets to get involved in the discussion about a topic, and also to make their voice easier to be found by others. This feature measures the overlap

in hashtags between tweet pairs.

Hypothesis DH3: The more common hashtags we find between a pair of tweets, the more likely they are duplicates and the higher the duplicate score.

Feature DF4: Overlap in URLs Due to the length limitation of tweets, users often make use of URLs to provide pointers to relevant more detailed information. Hence we determine the overlap of the links contained in the given pair of tweets. If a pair of tweets contain the same URL, they are probably about the same topic and are likely to be duplicates.

Hypothesis DH4: The more overlap in URLs we find between a pair of tweets, the more likely they are duplicates and the higher the duplicate score.

Feature DF5: Overlap in expanded URLs. Various Twitter client applications and sharing functions used by news media sites shorten the URLs in order to give more space for real content [10]. As a result, we may miss some actual overlap in URLs if we only check the original URLs. For this reason, we measure the overlap in expanded URLs between tweets. The expanded URLs can be obtained via the redirected locations given in the HTTP responses.

Hypothesis DH5: The more common URLs we find between a pair of tweets after expanding the URLs, the more likely they are duplicates and the higher the duplicate score.

Feature DF6: Length difference. Besides matching letters, words, hashtags, and URLs, we also calculate the difference in length between two tweets and normalize it by L_{max}:

$$length\_difference = \frac{abs(|tweet_a| - |tweet_b|)}{140} \tag{4.2}$$

Hypothesis DH6: The smaller the difference in length between two tweets, the higher the likelihood of them being duplicates and the higher their duplicate score.

Semantic Features

Apart from syntactic features of tweet pairs, semantic information may also be valuable for identifying duplicates, especially when the core messages or important entities in tweets are mentioned in different order. For this reason, we analyze the semantics in both tweets of a pair and construct features that may help with distinguishing duplicate tweets. We utilize NER services, including DBpedia Spotlight, OpenCalais, as well as the lexical database WordNet to extract the following features.

Feature DF7: Overlap in entities. Given extracted entities or concepts, we can determine the overlap between the sets of entities in tweet pairs. The near-duplicate tweet pairs should contain the same core message and therefore the same entities should be mentioned.

Hypothesis DH7: Tweet pairs with more overlapping entities are more likely to have a high duplicate score.

Feature DF8: Overlap in entity types. Lets now consider the types of entities extracted. For example, if t_b contains the entities of type *person* and *location*, t_a should also contain the same type of entities if they are a near-duplicate tweet pair. Otherwise, more types of entities may indicate more information or fewer types may suggest only a partial coverage of the core message. Therefore, we construct features that measure the overlap in entity types between tweet pairs.

Hypothesis DH8: Tweet pairs with more overlapping entity types are more likely to have a high duplicate score.

Feature DF9: Overlap in topics. Besides outputting entities with types, OpenCalais can classify the input textual snippets into 18 different categories a.k.a. *topics*. In this case, each tweet may be assigned more than one topic label or no topic at all. Therefore, it is possible to construct a feature by checking the overlap in topics.

Hypothesis DH9: The tweet pairs that share more topics are more likely to have a high duplicate score.

Feature DF10: Overlap in WordNet concepts. We constructed this feature to compute the overlap based on lexical standards. To achieve this, we make use of the lexical database WordNet [116] to identify the nouns in pairs of tweets and calculate their overlap in these nouns. Practically, we use JWI (MIT Java Wordnet Interface)[1] to find the root concepts of the nouns in the tweets, e.g. W_a is the set of WordNet noun concepts that appear in t_a, W_b stands for the set of WordNet concepts that appear in t_b:

$$overlap_{WordNet} = \frac{|W_a \cap W_b|}{|W_a \cup W_b|} \tag{4.3}$$

Hypothesis DH10: The more overlap in WordNet noun concepts we find in a pair of tweets, the more likely they are to be duplicates and the higher their duplicate score.

Feature DF11: Overlap in WordNet synset. Making use of merely WordNet noun concepts may not fully cover the overlap in information because different tweets may use different words or synonyms to convey the same information. In WordNet, synsets are interlinked by means of conceptual-semantic and lexical relations. That is to say, synonyms or words that denote the same concept and are interchangeable in many contexts, are grouped into unordered sets (synsets)[2]. Therefore, we can make use of synsets to include all words with similar meaning.

Hypothesis DH11: If the words in synsets are included for checking overlap between tweet pairs then the overlap feature may have a more positive correlation with the duplicate scores.

Feature DF12: WordNet similarity. There are several existing algorithms for calculating the semantic relatedness between WordNet concepts, e.g. the method proposed by Lin et al. [98] can measure the semantic relatedness between two concepts with a value between $[0, 1]$. The WordNet concepts are paired according to their relatedness. Practically, we follow Algorithm 1 to compute this feature for a tweet pair (t_a, t_b). In the description of the algorithm, W_a stands for the set of WordNet noun concepts that appear in t_a.

[1] http://projects.csail.mit.edu/jwi/, accessed July 30th, 2014
[2] http://wordnet.princeton.edu, accessed October 6th, 2014

Algorithm 1: WordNet similarity of a tweet pair

input : Tweet Pair (t_a, t_b)
output: WordNet similarity of Tweet Pair (t_a, t_b)

```
acc ← 0;
if |t_a| > |t_b| then
    swap(t_a, t_b);
foreach WordNet noun concept c_a in t_a do
    maximum ← 0;
    foreach WordNet noun concept c_b in t_b do
        if maximum < similarity_lin (c_a, c_b) then
            maximum ← similarity_lin (c_a, c_b);
    acc ←acc + maximum;
return acc / |W_a|;
```

Hypothesis DH12: The higher the WordNet similarity of a tweet pair, the higher the likelihood of the tweets being duplicates and the higher their duplicate score.

Enriched Semantic Features

Due to the length limitation of tweets, 140 characters may not be enough to tell a complete story. Furthermore, some tweets, created by sharing buttons from other news sites for example, may even break the complete message. Thus, we make use of the external resources that are linked from the tweets. This step yields additional information and further enriches the tweets' semantics. Finally, we build a set of so-called *enriched* semantic features.

We construct six enriched semantic features, which are constructed in the same way as the semantic features introduced in Section 4.4.1. The only difference is that the source of semantics contains not only the content of the tweets but also the content that we find by retrieving the content of the Web pages that are linked from the tweets.

Contextual Features

Besides analyzing syntactic and semantic aspects, which describe the characteristics of tweet pairs, we also evaluate the effects of the context in which the tweets were published. We investigate three types of contextual features:

temporal difference of the creation times, similarity of the tweets' authors, and the client application that the authors used.

Feature DF13: Temporal difference. For several popular events, e.g. the UK Royal wedding and the Super Bowl, users have posted thousands of tweets per second. During these events, breaking news are often retweeted not long after being posted. Therefore, it is reasonable to assume that the time difference between duplicate tweets is rather small. We normalize this feature by dividing the original value by the length of the temporal range of the dataset (two weeks in our setup).

Hypothesis DH13: The smaller the difference in posting time between a pair of tweets, the higher the likelihood of it being a duplicate pair and the higher the duplicate score.

Feature DF14: User similarity. Similar users may publish similar content. We measure user similarity in a lightweight fashion by comparing the number of followers and the number of followees. Hence, we extract two features: the differences in *#followers* and *#followees* to measure the similarity of the authors of a post. As the absolute values of these two features vary in magnitude, we normalize this feature by applying log-scale and dividing by the largest difference in log-scale.

Hypothesis DH14: The higher the similarity of the authors of a pair of tweets, the more likely that the tweets are duplicates.

Feature DF15: Same client. This is a boolean feature that is set to true when the pair of tweets were posted by the same client application. With authorization, third-party client applications can post tweets on behalf of users. Hence, different Twitter client applications as well as sharing buttons on various Web sites are being used. As the tweets that are posted from the same applications and Web sites may share similar content, provenance information and particularly information about the client application may be used as evidence for duplicate detection.

Hypothesis DH15: The tweet pairs that are posted from the same client application tend to be near-duplicates.

4.4.2 Feature Analysis

As previously stated, we take the *Tweets2011* corpus as our Twitter stream sample for the task of duplicate detection. Before we turn to evaluating our duplicate detection strategies, we first perform an in-depth analysis of the features that we presented in Section 4.4.1. Regarding semantic features, we tried employing both DBpedia Spotlight and OpenCalais. In fact, we later found only a slight difference in performances. Therefore in practice, we construct two features and derivative features (introduced later) by using *DBpedia Spotlight* because it yields slightly better results. We extracted these features for the 55,362 tweet pairs with duplication judged (see Section 4.3.2). In Table 4.1, we list the average values and the standard deviations of the features and the percentages of *true* instances for the boolean feature respectively (*same client*). Moreover, Table 4.1 shows a comparison between features of duplicate (across all 5 levels of duplication) and non-duplicate tweet pairs.

Unsurprisingly, the Levenshtein distances of duplicate tweet pairs are on average 15% shorter than the ones of non-duplicate tweet pairs. Similarly, duplicate tweet pairs share more identical terms than non-duplicate ones: the duplicates have a Jaccard Similarity of 0.2148 in terms, whereas only 0.0571 for the non-duplicates. Hence, these two features which compare the tweets in letters and words may be potentially good indicators for duplicate detection. Although there is a difference in common hashtags between the duplicates and the non-duplicates, the overlap in hashtags does not seem to be a promising feature as indicated by the low absolute value. This may be explained by the low usage of hashtags. The two features that are based on the overlap in hyperlinks show similar characteristics but are slightly more distinguishing. As expected, we discover more overlap in links by expanding the shortened URLs.

Tweet pairs may convey the same messages with syntactically different but semantically similar words. If this is the case then the syntactic features may fail to detect the duplicate tweets. Therefore, the features that are formulated as overlap in semantics are expected to be larger in absolute values than the syntactic overlap features. Overall, the statistics that are listed in Table 4.1 are in line with our expectations. We discover more overlap in the duplicates along 3 dimensions, including entities, entity types, and topics, by exploiting semantics with NER services. More distinguishable differences can be found in the features constructed from WordNet. The duplicate tweet pairs have more overlap in WordNet noun concepts or synsets (0.38) than the non-duplicate pairs (0.12). The feature of WordNet similarity is also

Category	Feature	Duplicate	Std. deviation	Non-duplicate	Std. deviation
syntactic	Levenshtein Distance	0.5340	0.2151	0.6805	0.1255
	overlap in terms	0.2148	0.2403	0.0571	0.0606
	overlap in hashtags	0.0054	0.0672	0.0016	0.0337
	overlap in URLs	0.0315	0.1706	0.0002	0.0136
	overlap in expanded URLs	0.0768	0.2626	0.0017	0.0406
	length difference	0.1937	0.1656	0.2254	0.1794
semantics	overlap in entities	0.2291	0.3246	0.1093	0.1966
	overlap in entity types	0.5083	0.4122	0.3504	0.3624
	overlap in topics	0.1872	0.3354	0.0995	0.2309
	overlap in WordNet concepts	0.3808	0.2890	0.1257	0.1142
	overlap in WordNet Synset concepts	0.3876	0.2897	0.1218	0.1241
	WordNet similarity	0.6090	0.2977	0.3511	0.2111
enriched semantics	overlap in entities	0.1717	0.2864	0.0668	0.1230
	overlap in entity types	0.3181	0.3814	0.1727	0.2528
	overlap in topics	0.2768	0.3571	0.1785	0.2800
	overlap in WordNet concepts	0.2641	0.3249	0.0898	0.0987
	overlap in WordNet Synset concepts	0.2712	0.3258	0.0927	0.1046
	WordNet similarity	0.7550	0.2457	0.5963	0.2371
contextual	temporal difference	0.0256	0.0588	0.2134	0.2617
	difference in #followees	0.3975	0.1295	0.4037	0.1174
	difference in #followers	0.4350	0.1302	0.4427	0.1227
	same client	21.13%	40.83%	15.77%	36.45%

Table 4.1: The comparison of features between duplicate and non-duplicate tweets. Although we have 5 levels of duplication, any pair judged as duplicate on any level are considered as duplicates.

potentially a good criterion for duplicate detection: the average similarity of duplicate pairs is 0.61 compared to 0.35 for non-duplicate pairs. The comparison of the enriched semantic features shows similar findings to those we observed for the semantic features. Again, the features that compare WordNet-based concepts are more likely to be good indicators for duplicate detection. However the WordNet similarity shows less difference if we consider external resources.

Finally, we attempted to detect the duplicates based on information about the context in which the tweets were posted. Hypothesis DH13 (see Section 4.4.1) states that duplicates are more likely to be posted in a short temporal range. The result for the feature of temporal difference in Table 4.1 supports this hypothesis: the average value of this feature for the duplicate pairs is only 0.0256 (about 8 hours before normalization, see Section 4.4.1) in contrast to 0.2134 (about 3 days) for the non-duplicate ones. With respect to user similarity, we have not discovered an explicit difference between the two classes. Regarding the client applications from which duplicate tweets are posted, we observe the following: 21.1% of the duplicate pairs were posted from the same client applications whereas only 15.8% of the non-duplicate ones show the same characteristic.

4.4.3 Duplicate Detection Strategies

Having all the features constructed in Section 4.4.1 and preliminarily analyzed in Section 4.4.2, we now create different strategies for the task of duplicate detection. In practice, as requirements and limitations may vary in processing time, real-time demands, storage, network bandwidth etc., different strategies may be adopted. Given that our models for duplicate detection are derived from logistic regression, we define the following strategies by combining different sets of features, including one *Baseline strategy* and six *duplicate detection strategies* proposed by us: *Sy* (only syntactic features), *SySe* (including tweet content-based features), *SyCo* (without semantics), *SySeCo* (without enriched semantics), *SySeEn* (without contextual features), and *SySeEnCo* (all features).

Baseline Strategy

In previous work, Levenshtein distance [118] has been used as the method to identify the similarity in Twitter messages. Therefore, we use it as the only feature in the baseline strategy, which compares tweet pairs letter by letter

to classify pairs as duplicates.

Duplicate Detection Strategies

Our duplicate detection strategies exploit the sets of features that have been introduced in Section 4.4.1). In our duplicate detection framework, the duplicate detection strategies can easily be defined by grouping together different features.

Sy The *Sy* strategy is the most basic strategy in Twinder. It includes only *sy*ntactic features that compare tweets on a term level. These features can easily be extracted from the tweets and are expected to have a good performance on the duplicates for the levels of *Exact copy* or *Nearly exact copy*.

SySe This strategy makes use of the features that take the actual content of the tweets into account. Besides the *sy*ntactic features, this strategy makes use of NER services and WordNet to obtain the *se*mantic features.

SyCo The strategy of *SyCo* (without semantics) is formulated to prevent the retrieval of external resources as well as semantic extractions that rely on either external Web services or extra computation time. Only *sy*ntactic features and *co*ntextual features are considered by this strategy.

SySeCo Duplicate detection can be configured as applying features without relying on external Web resources. We call the strategy that uses the *sy*tactical features, *se*mantics that are extracted from the content of tweets, and the *co*ntextual information *SySeCo*.

SySeEn The contextual features, especially the ones related to users, may require extra storage and may be recomputed frequently. Therefore, the duplicate detection may work without contextual information by applying the so-called *SySeEn* (without contextual features).

SySeEnCo If enough hardware resources and network bandwidth are available then the strategy that integrates all the features can be applied so that the quality of the duplicate detection can be maximized.

4.5 Evaluation of Duplicate Detection Strategies

To understand how different features and strategies influence the effectiveness (i.e. the classification accuracy) of duplicate detection, we formulated a number of research questions, which can be summarized as follows:

1. How accurately can the different *duplicate detection strategies* identify duplicates?
2. What kind of *features* are of particular *importance* for duplicate detection?
3. How does the importance of features vary for *different types of search topics*?
4. How does the accuracy vary for the *different levels* of duplicates?

4.5.1 Experimental Setup

We employ logistic regression for both duplicate detection tasks: (i) to classify tweet pairs as duplicate or non-duplicate and (ii) to estimate the duplication level. Due to the limited amount of duplicate pairs (of all 5 levels, 2,745 instances) in the manually labelled dataset (55,362 instances in total, see Section 4.3.2), we use 5-fold cross-validation to evaluate the learned classification models. At most, we used 22 features as predictor variables (see Table 4.1). Since the fraction of positive instances is considerably smaller than the negative one, we employed a cost-sensitive classification setup to prevent all tweet pairs from being classified as non-duplicates. Moreover, as the precision and recall for non-duplicate are over 90%, we use the non-duplicate class as the reference class and focus on the classification accuracy of the class of duplicates. We use precision, recall, and F-measure to evaluate the results. Furthermore, since our final objective in this chapter is to reduce redundancy in search results, we also point out the fraction of false positives as the indicator of the costs of losing information by applying our framework.

4.5.2 Influence of Strategies on Duplicate Detection

Table 4.2 shows the performance of predicting the duplicate tweet pairs by applying the strategies described in Section 4.4.3. The baseline strategy, which only uses Levensthein distance, leads to a precision and recall of 0.5068 and 0.1913 respectively. It means, for example, if 100 relevant tweets are returned for a certain search query and about 20 tweets (the example ratio of 20% according to the statistics given in Section 4.3.2) are duplicates that could be removed, the baseline strategy would identify 8 tweets as duplicates. However, only 4 of them are correctly classified while 16 other true duplicates are missed. In order to combine both precision and recall in once, the F-measure is used and for the *Baseline* strategy the value is 0.2777. In contrast, the *Sy*

strategy, which is the most basic one, leads to a much better performance in terms of all measures, e.g. an F-measure of 0.3923. By combing the contextual features, the *SyCo* strategy achieves a slightly better F-measure of 0.4067. It appears that the contextual features contribute relatively little to the classification accuracy.

Strategies	**Precision**	**Recall**	**F-measure**
Baseline	0.5068	0.1913	0.2777
Sy	0.5982	0.2918	0.3923
SyCo	0.5127	0.3370	0.4067
SySe	0.5333	0.3679	0.4354
SySeEn	0.5297	0.3767	0.4403
SySeCo	0.4816	0.4200	0.4487
SySeEnCo	0.4868	0.4299	0.4566

Table 4.2: Performance Results of duplicate detection for different sets of features

Subsequently, we leave out the contextual features and compute the importance of semantics in the content of the tweets and external resources. The *SySe* (including tweet content-based features) strategy considers not only the syntactic features but also the semantics extracted from the content of the tweets. We find that the semantic features can boost the classifier's effectiveness as the F-measure increased to 0.4354. The enriched semantics extracted from external resources brought little benefit to the result as the *SySeEn* strategy has a performance with F-measure of 0.4403. Overall, we conclude that semantics play an important role as they lead to a performance improvement with respect to F-measure from 0.3923 to 0.4403.

Thus the so-called *SySeCo* strategy excludes the features of enriched semantics but again includes the contextual features. Given this strategy, we observe an F-measure of 0.4487. However, if we adopt the strategy of *SySeEnCo* (all features), the highest F-measure can be achieved. At the same time, we nearly achieve the same precision as the *Baseline* strategy but boost the recall from 0.1913 to 0.4299. This means that more than an additional 20% of duplicates can be found while maintaining accuracy levels. In this stage, we will further analyze the impact of the different features in detail as they are used in the strategy of *SySeEnCo*.

In the logistic regression approach, the importance of features can be investigated by considering the absolute value of the coefficients assigned to them. We have listed the details about the model derived for the *Sy-*

	Performance Measure	**Score**
	precision	0.4868
	recall	0.4299
	F-measure	0.4566
Category	**Feature**	**Coefficient**
syntactic	Levenshtein distance	-2.9387
	overlap in terms	2.6769
	overlap in hashtags	0.4450
	overlap in URLs	1.2648
	overlap in expanded URLs	0.8832
	length difference	1.2820
semantics	overlap in entities	-2.1404
	overlap in entity types	0.9624
	overlap in topics	1.4686
	overlap in WordNet concepts	4.5225
	overlap in WordNet Synset concepts	0.6279
	WordNet similarity	-0.8208
enriched semantics	overlap in entities	-0.8819
	overlap in entity types	0.9578
	overlap in topics	-0.1825
	overlap in WordNet concepts	-2.0867
	overlap in WordNet Synset concepts	2.5496
	WordNet similarity	0.7949
contextual	temporal difference	-12.6370
	difference in #followees	0.4504
	difference in #followers	-0.3757
	same client	-0.1150

Table 4.3: The coefficients of different features

SeEnCo (all features) strategy in Table 4.3, in which the five features with the highest absolute coefficients are underlined. The most important features are:

- *Levenshtein distance*: As it is a feature of negative coefficient in the classification model, we infer that a shorter Levenshtein distance indicates a higher probability of being duplicate pairs. Therefore, we confirm our Hypothesis DH1 made in Section 4.4.1.
- *overlap in terms*: Another syntactic feature also plays an important role

as the coefficient is ranked fourth most indicative in the model. This can be explained by the usage of common words in duplicate tweet pairs. This result supports Hypothesis DH2.

- *overlap in WordNet concepts*: The coefficients of semantic and enriched semantic vary in the model. However, the most important feature is overlap in WordNet concepts. It has the largest positive weight which means that pairs of tweets with high overlap in WordNet concepts are more likely to be duplicates, confirming Hypothesis DH10 (Section 4.4.1). However, we noticed a contradiction in the feature set of enriched semantics, in which the coefficient for overlap in WordNet concepts is negative (-2.0867) whereas the one the coefficient for the overlap in WordNet synset concept is positive (2.5496). It can be explained by the high correlation between these two features, especially for high coverage of possible words in external resources. For this reason, they counteract each other in the model.

- *temporal difference*: In line with the preliminary analysis, the shorter the temporal difference between a pair of tweets, the more likely that it is a duplicate pair. The highest value of the coefficient is partially due to low average absolute values of this feature. However, we can still conclude that Hypothesis DH13 holds (see Section 4.4.1).

Overall, we note that the hypotheses we derived for syntactic features hold. The same conclusion cannot be drawn about the hypotheses that are based on semantic features. There are several reasons for this outcome. Consider, for example, the overlap of WordNet concepts in the set of enriched semantics, which is negative. The reason for this may be twofold: (i) more general terms (such as politics, sport, news, mobile) are overlapping if we consider external resources; (ii) the features in the set of enriched semantics may mislead when we extract the features for a pair of tweets from which no external resources can be found or only one tweet contains a URL. The situation for other features, e.g. WordNet similarity, can be explained by the dependencies between some of them. More specifically, the features that are based on WordNet similarity in the sets of semantics and enriched semantics may have positive correlation. Therefore, the coefficients complement each other in values. When we consider only the contextual features, all other three features except the temporal difference do not belong to the most important features. More sophisticated techniques for measuring user similarity might be used to better exploit, for example, the provenance of tweets for the duplicate detection task.

4.5.3 Influence of Topic Characteristics on Duplicate Detection

In all reported experiments so far, we have considered the entire Twitter sample available to us. In this section, we investigate to what extent certain topic (or query) characteristics play a role for duplicate detection and to what extent those differences lead to a change in the logistic regression models.

Consider the following two topics: *Taco Bell filling lawsuit* (MB020[3]) and *Egyptian protesters attack museum* (MB010). While the former has a business theme and is likely to be mostly of interest to American users, the latter topic belongs into the category of politics and can be considered as being of global interest, as the entire world was watching the events in Egypt unfold. Due to these differences, we defined a number of topic splits. A manual annotator then decided for each split dimension into which category the topic should fall. We investigated four topic splits, three splits with two partitions each as introduced in Section 3.4.3 and one split with five partitions described as follows:

- Topic themes: The topics were classified as belonging to one of five themes, either business, entertainment, sports, politics or technology. MB002 is, e.g., a sports topic while MB019 is considered to be a political topic.

Our discussion of the results focuses on two aspects: (i) the difference between the models derived for each of the two partitions, and (ii) the difference between these models (denoted $M_{splitName}$) and the model derived over all topics ($M_{allTopics}$).

The results for the three binary topic splits are shown in Table 4.4. There are three splits shown here: popular vs. unpopular topics, global vs. local topics, and persistent vs. occasional topics. While the performance measures are based on 5-fold cross-validation, the derived feature weights for the logistic regression model were determined across all topics of a split. The total number of topics is 49. For each topic split, the three features with the highest absolute coefficient are underlined.

Popularity: A comparison of the most important features of $M_{popular}$ and $M_{unpopular}$ shows few differences with the exception of a single feature: temporal difference. While temporal difference is the most important feature in $M_{popular}$, it is ranked fourth in $M_{unpopular}$. We hypothesize that the

[3]The identifiers of the topics correspond to the ones used in the official TREC dataset.

Performance	**Measure**	**popular**	**unpopular**	**global**	**local**	**persistent**	**occasional**
	#topics	24	25	18	31	28	21
	#samples	32,635	22,727	19,862	35,500	33,474	21,888
	precision	0.4480	0.6756	0.6148	0.4617	0.4826	0.6129
	recall	0.4569	0.6436	0.5294	0.5059	0.5590	0.5041
	F-measure	0.4524	0.6592	0.5689	0.4828	0.5180	0.5532
Category	**Feature**	**popular**	**unpopular**	**global**	**local**	**persistent**	**occasional**
syntactic	Levenshtein distance	-3.4919	-0.6126	0.1342	-3.5136	-3.5916	-1.2338
	overlap in terms	2.8352	6.0905	6.7126	1.0498	1.3474	5.7705
	overlap in hashtags	0.6234	-2.5868	-1.8751	1.2671	2.2210	-2.7187
	overlap in URLs	-0.1865	5.8130	0.3275	1.6342	3.4323	0.4907
	overlap in expanded URLs	0.5180	2.7594	1.4933	0.6362	1.4936	1.0751
	length difference	1.2459	0.5974	0.5028	1.3236	1.5043	0.3793
semantics	overlap in entities	-2.3460	0.5430	-0.2263	-3.3525	-3.1071	0.5538
	overlap in entity types	1.2612	-1.1651	-0.3301	1.1571	0.8802	-0.8804
	overlap in topics	1.6607	0.7505	1.2147	1.2294	1.3911	0.7848
	overlap in WordNet concepts	5.5288	7.1115	5.8185	2.5319	4.0427	4.6365
	overlap in WordNet Synset concepts	-0.7763	-2.4393	-0.7335	3.0327	1.8752	0.5750
	WordNet similarity	-0.6254	1.8168	1.1355	-0.3909	-0.5141	1.0457
enriched semantics	overlap in entities	-1.3013	0.0583	0.0501	-0.5548	-1.9666	0.8555
	overlap in entity types	0.8997	1.2098	0.1179	0.8132	1.0279	0.3228
	overlap in topics	-0.5470	0.6581	0.0118	-0.1282	0.0292	-0.1884
	overlap in WordNet concepts	-1.8633	-1.2312	-2.3160	-2.4825	-2.5058	-2.1868
	overlap in WordNet Synset concepts	2.4274	1.0336	3.4218	2.6263	3.0461	2.5514
	WordNet similarity	0.7328	0.4342	-1.0359	0.9406	1.0470	-1.1867
contextual	temporal difference	-15.8249	-2.9890	-5.2712	-17.3894	-18.9433	-5.6780
	difference in #followees	0.7026	0.2322	-1.0797	0.5489	0.0659	-1.0473
	difference in #followers	-0.9826	1.1960	-0.3224	0.0048	-0.1281	-0.5178
	same client	-0.1800	0.3851	-0.0272	-0.0692	-0.2641	0.3058

Table 4.4: Influence comparison of different features among 3 different topic binary partitions

discussion on popular topics evolves quickly on Twitter, thus the duplicate tweet pairs should have little difference in posting time.

Global vs. Local: The most important feature in M_{global}, the overlap in terms, and the second most important feature in M_{local}, Levenshtein distance, do not have a similar significance in each others' models. We consider it as an interesting finding and the possible explanation can lie in the sources of the information. On the one hand the duplicate tweets about the local topics may share the same source thus are low in Levenshtein distances; on the other hand, different sources may report on the global topics in their own styles but with the same terms.

Temporal persistence: Comparing the $M_{persistent}$ and the $M_{occasional}$ models, yields to similar conclusions as in the previous two splits: (i) the persistent topics are continuously discussed so that the duplicate pairs are more likely to have short temporal differences, while the temporal differences between tweets on occasional topics are relatively insignificant; (ii) the occasionally discussed topics are often using the same set of words.

Theme	**business**	**entertainment**	**sports**	**politics**	**technology**
#topics	6	12	5	21	2
#samples	11,445	7,678	1,722	30,037	1,622
Measure	**business**	**entertainment**	**sports**	**politics**	**technology**
precision	0.6865	0.6844	0.5000	0.4399	0.6383
recall	0.6615	0.7153	0.6071	0.4713	0.7143
F-measure	0.6737	0.6995	0.5484	0.4551	0.6742

Table 4.5: Performance comparison across topic theme partitions

Topic Themes: The partial results of the topic split according to the theme of the topic are shown in Table 4.5[4]. Three topics did not fit in any of the five categories. Since the topic set is split into five partitions, the size of some partitions is extremely small, making it difficult to reach conclusive results. Nevertheless, we can detect trends such as the fact that duplicate tweet pairs in sports topics are more likely to contain the same source links (positive coefficient of overlap in original URLs), while duplicate pairs in entertainment topics contain more shortened links (positive coefficient of overlap in expanded URLs). The overlap in terms has a large impact on all themes apart from politics. Another interesting observation is that a short temporal difference, is a prominent indicator for the duplicates in the topics of entertainment and politics but not in the other models.

[4]Full results, which contain the coefficient of all features in each topic theme, can be found online at http://ktao.github.io/phd/)

4.5.4 Analysis of Duplicate Levels

Having estimated whether a tweet pair is duplicate or not, we now proceed to the second step of the duplicate detection task: determining the duplication level of the tweet pair. We compare the different strategies (see Section 4.4.3) in the same way as we have done in Section 4.5.2. To analyze the performance in general, we used the weighted measures, including precision, recall, and F-measure, across 5 levels. The weight of each level depends on the ratio of duplicate instances. The effectiveness constantly improves as more features are considered in terms of F-measure. A similar pattern in performance improvement can be observed from the results are summarized in Table 4.6. However, it appears that the enriched semantics are more prominent than the contextual features as the so-called *SySeEn* strategy (without contextual features) performs better than *SySeCo* strategy (without enriched semantics).

Strategies	**Precision**	**Recall**	**F-measure**
Baseline	0.5553	0.5208	0.5375
Sy	0.6599	0.5809	0.6179
SyCo	0.6747	0.5889	0.6289
SySe	0.6708	0.6151	0.6417
SySeEn	0.6694	0.6241	0.6460
SySeCo	0.6852	0.6198	0.6508
SySeEnCo	0.6739	0.6308	0.6516

Table 4.6: Performance Results of predicting duplicate levels for different sets of features

In Figure 4.2, we plot the performance of the classification for 5 different levels and the weighted average of them with all the strategies that we have introduced in Section 4.4.3. The curves for 5 different levels show a similar trend. We observe that the level of *Weak near duplicate* performs better than average and the reason can be attributed to the large ratio of learning instances (see Figure 4.1). The classification of *Exact copy* is the best due to the decisive influence of the Levensthein distance. However, we see a declining trend in performance as we integrate more other features. Hence, further optimization is possible.

4.5.5 Optimization of Duplicate Detection

To optimize the duplicate detection procedure, we exploit the fact that duplicate pairs of level *Exact copy* can easily be detected by their Levenshtein distance of 0. After the removal of mentions, URLs, and hashtags, we can

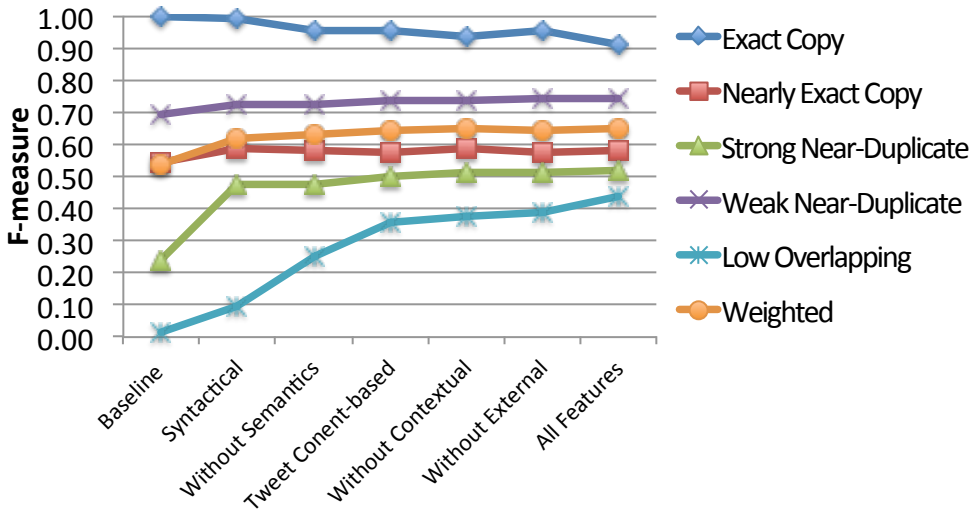


Figure 4.2: The F-measure of classification for different levels and weighted average by applying different strategies

also apply the same rule for *Nearly exact copy*. Therefore, we can optimize the duplicate detection procedure with the following cascade:

1. If the Levensthein distance is zero between a pair of tweets or after removal of mentions, URLs, and hashtags from both of them, they can be classified as *Exact copy* or *Nearly exact copy*;
2. Otherwise, we apply the aforementioned strategies to detect duplication.

After this optimization, we get a performance improvement from 0.45 to 0.55 with respect to the F-measure. The corresponding results are listed in Table 4.7 (the original results are given in Table 4.2).

4.6 Near-Duplicate Detection in Twinder

A core application of near-duplicate detection strategies is to lower the redundancy in search results in order to achieve diversification. Therefore, we integrated our duplicate detection framework into the Twinder, the prototype search engine that we first introduced in Chapter 2 and improved with our findings in relevance estimation in Chapter 3. Figure 4.3 depicts the architecture of Twinder and highlights the core modules which we designed, developed and analyzed in the context of this chapter.

Strategies	Precision	Recall	F-measure
Baseline	0.9011	0.2856	0.4337
Sy	0.7065	0.4095	0.5185
SyCo	0.6220	0.4550	0.5256
SySe	0.6153	0.4849	0.5424
SySeEn	0.5612	0.5395	0.5501
SySeCo	0.6079	0.4914	0.5435
SySeEnCo	0.5656	0.5512	0.5583

Table 4.7: Performance Results of duplicate detection using different strategies after optimization

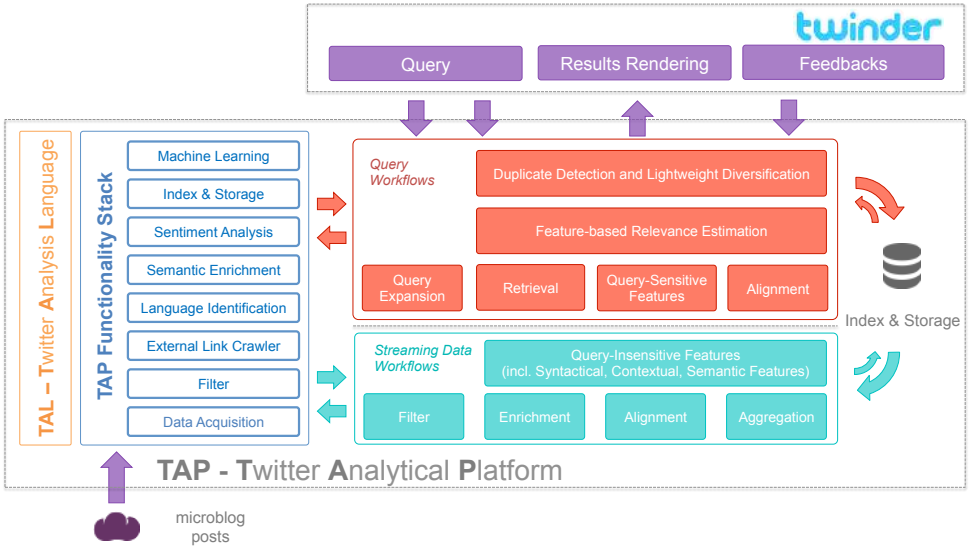


Figure 4.3: Architecture of the Twinder Search Engine with Duplicate Detection

4.6.1 Lightweight Diversification Strategy

In the updated version of Twinder, we perform the duplicate detection and diversification after the relevance estimation of the tweets (see Section 3.5). Hence, given a search query, the engine first ranks the tweets according to their relevance and then iterates over the top-k tweets of the search results to remove near-duplicate tweets and diversify the search results. Both, the duplicate detection and the relevance estimation module, benefit from the features that are extracted as part of the indexing step which is performed iteratively as soon as new tweets are monitored.

The lightweight diversification strategy applies the near-duplicate detection functionality as listed in Algorithm 2. It iterates from the top to the

Algorithm 2: Diversification Strategy

```
input  : Ranking of tweets T, k
output: Diversified top-k ranking T'@k
T'@k ← ∅;
i ← 0;
while i < k and i < T.length do
    j ← i+1 ;
    while j < T.length do
        if T[i] and T[j] are duplicates then
            remove T[j] from T;
        else
            j++;
    T'[i] = T[i]
return T'@k;
```

bottom of the top-k search results. For each tweet i, it removes all tweets at rank j with $i < j$ (i.e. tweet i has a better rank than tweet j) that are near-duplicates of tweet i.

4.6.2 Evaluation of Lightweight Diversification Strategy

In Section 4.3.2, we analyzed the ratios of duplicates in the search results. After applying the lightweight diversification strategy proposed above, we again examine the ratios. The results are listed in Table 4.8 and reveal that the fraction of near-duplicate tweets within the top-k search results is considerably smaller. For example, without diversification there exists, on average, for 22.2% of the tweets at least one near-duplicate tweet within the top 20 search results. In contrast, the diversification strategy improves the search result quality with respect to duplicate content by more than 50%. Thus there are, on average, less than 11% duplicates in the top 20 search results.

Range	Top 10	Top 20	Top 50	All
Before diversification	19.4%	22.2%	22.5%	22.3%
After diversification	9.1%	10.5%	12.0%	12.1%
Improvement	53.1%	52.0%	46.7%	45.7%

Table 4.8: Average ratios of near-duplicates in search results after diversification

4.6.3 Implementation in TAL

In order to integrate Duplicate Detection into Twinder, we again need two workflows, one for preprocessing the continuously arriving microposts, and another for providing the search result lists that will be processed by both the feature-based relevance estimation component introduced in Chapter 3 and the duplicate detection framework proposed in this chapter.

As we also exploit the external links in the microposts for duplicate detection process, we adapt the preprocessing script given in Section 3.5.2 as follows.

```
twitter.sample()

# add an attribute named lang to represent the language identification result
this.lang := langid(this.text)
filter := this.lang == "en" # filter out non-English tweets

# prepare the semantics from tweets and external webpages, sentiment
this.semantics := semantics.dbp(this.text)
this.sentiment := sentiment(this.text)
this.urls.content := extcrawl(this.urls)
this.ext_semantics :=  semantics.dbp(this.urls.content)

store()
index()
```

According to the lightweight diversification strategy introduced in Section 4.6.1, we need to implement Algorithm 2 in TAL. Therefore, we append the candidate list of potential tweets with features to each item in the search results and automatically classify the pairs into the category of duplicate or non-duplicate pairs. The implemented in TAL is presented as follows:

```
# will give LM-based retrieval score, semantic relevance score
search("Haiti Aristide return")

# invoke the relevance estimation in one statement
this.query.relevance := enrich.relevance_estimation(this)

# filter out non-relevant items and the lower ranked items
filter(this.query.relevance != true)
filter(this.query.rank > 100)

# prepare the candidate lists
this.candidate := pairs(query.rank < this.query.rank)

# prepare the features
this.candidate.syntactic := ...
this.this.candidate.semantic := ...
this.candidate.enriched_semantic := ...
```

```
this.candidate.contextual := ...

# applying the pre-trained duplicate detection model
this.candidate.duplicate := ml.classify(this,DUPLICATE.model,DUPLICATE_TAL.mapping)
this.meta.isduplicate := ...

# filter out the items with marked as duplicates
filter(this.meta.isduplicate)
```

4.6.4 Demonstration

We have implemented the near-duplicate detection function in Twinder. Figure 4.4 shows the search result page for the query of "Haiti Aristide return"[5].

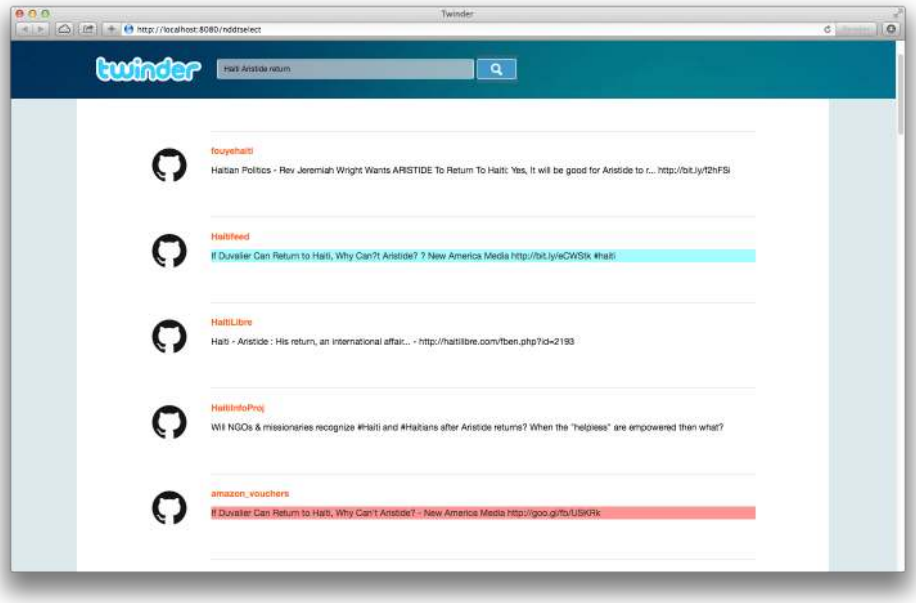


Figure 4.4: The search results rendered in Twinder with Near Duplicate Detection applied

The micropost text with a red background is detected as near-duplicate of the one with the text in cyan.

[5]Query with the identifier of MB003 in TREC 2011 Microblog Track

4.7 Discussion

In Chapter 2, we introduced Twinder, a search engine for Twitter streams, and improved the relevant estimation in Chapter 3. Having investigated the dataset, we find people are still confronted with a high fraction of near-duplicate content when searching and exploring information on microblogging platforms such as Twitter. In this chapter, we analyzed the problem of near-duplicate content on Twitter and developed a duplicate detection and lightweight search result diversification framework for Twitter.

To solve the problem and the research questions raised in the beginning of this chapter, we introduced and evaluated our duplicate detection framework. Further, we integrated this function into Twinder. We summarize our findings in Table 4.9.

Our framework is able to identify near-duplicate tweets with a precision and recall of 48% and 43% respectively by combining (i) syntactic features, (ii) semantic features, (iii) contextual features, (iv) by considering information from external Web resources that are linked from the microposts. For certain types of topics such as occasional news events, we observe performances of more than 61% and 50% with respect to precision and recall.

Our experiments show that semantic features such as the overlap of WordNet concepts are of particular importance for detecting near-duplicates. By analyzing a large Twitter sample, we also identified five main levels of duplication ranging from *exact copies* which can easily be identified by means of syntactic features such as string similarity to *low overlapping duplicates* for which an analysis of the semantics and context is specifically important. Our framework is able to classify the duplication score on that level with an accuracy of more than 60%.

Given our near-duplicate detection strategies, we additionally developed functionality for the diversification of search results. We integrated this functionality into the Twinder search engine and could show that our duplicate detection and diversification framework improves the quality of the top-k retrieval results significantly since we decrease the fraction of duplicate content that is delivered to the users by more than 45%.

However, we notice that the lower redundancy achieved in this chapter does not necessarily mean an increase in diversity in a more general sense. Therefore, we will further dedicate Chapter 5 to the investigation of this problem in order to provide a deeper insight into the diversity existing in microposts.

Research Question	Summary of Findings
How much duplicate content exists in typical microblog search results?	▶ By conducting a study on the corpus of Tweets2011, we inferred a 5-level duplication model of micropost pairs. ▶ Based on this model, we found about 20% of the search result items to be duplicate items.
How can we automatically detect the duplicate content along with the duplication level?	▶ We presented a duplicate detection framework to solve this problem by comparing syntactic characteristics, semantic similarity, and contextual information. ▶ The importance of semantic similarity between micropost pairs is more significant in duplicate detection compared to syntactic characteristics and contextual information. ▶ The topic characteristics have influences on the duplicate detection model and the performance can be improved by adapting the model to the characteristics or the theme of the query.
How does removing or aggregating duplicate contents affect the quality of the search results with respect to diversity?	▶ By integrating the duplicate detection framework into Twinder, we apply a lightweight diversification strategy and find that our approach can reduce the redundancy in the search results by about 50%.

Table 4.9: Overview on research questions investigated in Chapter 4

Chapter 5

Diversity: Exploring Subtopics in Micropost Retrieval

Given the prototype search engine for Twitter that has been introduced in Chapter 2, we have investigated approaches to estimate the relevance of tweets to the given query (see Chapter 3) as well as strategies to reduce redundancy (see Chapter 4). We have shown how these analytical tasks have been implemented in TAL, which is the key to our Twitter Analytical Platform. In this chapter, we further take the diversity of tweets as our research focus, with the motivation we got from the redundancy work. While search result diversity has been studied in the context of Web search for a number of years [8, 126], no comparable data set has been developed for microposts yet. To tackle this problem, we constructed a corpus dedicated to search research diversification. We present the methodology for building such a corpus and conduct a comprehensive analysis of its suitability for the designated purposes. Finally we apply our de-duplication framework to determine its effects on diversity in search results. The main contributions in this chapter have been published in [161].

5.1 Introduction

Given the massive amount of data being posted on Twitter about different aspects on certain events [5, 121, 133, 175], users may use the search function to get the information relevant to their interests. From the standard Web

search setup we know that queries that users pose to search engines are often ambiguous - either because different users express different query intents with the same query terms or because the query is underspecified and it is unclear which aspect of a particular query the user is interested in. Search result diversification, which aims at maximizing the coverage of a range of query intents or aspects of an underspecified or ambiguous topic within a search result ranking, has been shown in recent years to be an effective strategy to satisfy searchers in those circumstances. Instead of a single query intent or a limited number of aspects, search result rankings now cover a set of intents and a wide variety of aspects. Since 2009, with the introduction of the diversity task at TREC [41], a large increase in research efforts has been observed, e.g. [30, 138, 139, 144].

As mentioned in Section 3.2, previous research [163] has shown that the search queries issued to microblogging platforms are shorter than those submitted to traditional Web search engines. Considering the success of diversity in Web search, we believe that it is an even more important technology on microblogging platforms due to the shortness of the queries. Therefore, we formulate the main problem for this chapter as follows.

Problem 4 (Diversity) *Given micropost search results for a particular topic, the task of diversity analytics is to explore the existence of diversity, i.e. the subaspects within the general topic as specified by the query, and the feasibility of further research on automatic methods for search result diversification.*

The research in the Web search setting relies on corpora with relevance judgments dedicated to search result diversification purposes. However, to our knowledge, no publicly available microblogging data set, i.e. a corpus and a set of topics with subtopic-based relevance judgments, exists as of yet. In order to get a deeper understanding of diversity in the microblog setting, we created such a corpus[1] and describe it in this chapter. To tackle this problem, we will answer the following research questions in this chapter:

- How can we build a microblog corpus for search result diversification?
- How suitable is the corpus that we created for research on search result diversification?

[1] Please refer to http://ktao.github.io/phd/ for the dataset that we make publicly available.

- To what extent can we achieve diversity by applying the developed de-duplication strategies?

In Section 5.2, we introduce the related work on diversification methods, mainly in the Web search setting. We present a methodology for microblog-based corpus creation in Section 5.3 and conduct an analysis on its validity for diversity experiments in Section 5.4. Finally in Section 5.5, we turn to the question of how to improve search and retrieval in the diversity setting by evaluating the de-duplication approach to microblogging streams that we introduced in Chapter 4.

5.2 Related Work

Users of (Web) search engines typically employ short keyword-based queries to express their information needs. These queries are often underspecified or ambiguous to some extent [44]. Different users who pose exactly the same query may have very different query intents. In order to satisfy a wide range of users, search result diversification was proposed [15] to take the users preferences, novelty into the consideration for ranking documents. However, the problem is proved to be NP-hard [8] as it can be reduced to a maximum coverage problem.

On the Web, researchers have been studying the diversification problem mostly based on two considerations: novelty and facet coverage. To increase novelty, maximizing the marginal relevance while adding documents to the search results [28, 187] has been proposed. Later studies have focused on how to maximize the coverage of different facets [30] of a given query. Furthermore, there are works that consider a hybrid solution to combine benefits from both novelty-based and coverage-based approaches [144, 184]. Nevertheless, not all the queries need to apply the same diversification strategies, so that Santos et al. [137] proposed an algorithm that adapts on per-query basis.

One of the fundamental problems for search result diversification is to discover the different intents underlying the query, which is either ambiguous or underspecified. While researchers could bypassed this problem by utilizing the query suggestions provided by commercial search engines [136], there were also studies on tackling this exact problem. Choi et al. [39] introduced a method to identify the subtopics in news articles. For the contents from traditional media, the Latent Dirichlet Allocation (LDA) model [21] is employed to automatically derive a predefined number of topics. With

this model, Zhao et al. [191] presented a comparison of topical differences between Twitter and other media sources. Given the identified subtopics, search results can be diversified by different methods of improving the coverage of subtopics. For example, the xQuAD framework [140] was proposed to explicitly consider the relevance between the document that retrieved with the original query and the sub-queries.

In order to evaluate the effectiveness of search result diversification, different evaluation measures have been proposed. A number of them [8, 32, 40, 42] have been employed in the Diversity Task [41] of the Text REtrieval Conference (TREC), which ran between 2009 and 2012.

Given the difference [163] in querying behavior on the Web and microblogging sites, we hypothesize that the diversification problem is more challenging in the latter case due to the reduced length of the queries. The framework that we introduced for (near-)duplicate detection in Chapter 4 can be categorized as novelty-based since it exploits the dependency between documents in the initial result ranking. The evaluation though was limited due to the lack of an explicit diversity microblogging corpus (i.e. a corpus with topics and subtopics as well as relevance judgments on the subtopic level). In this chapter, we now tackle this very issue. The We describe our methodology for the creation of a Twitter-based diversity corpus and investigate its properties. Finally, we also employ our de-duplication framework (see Chapter 4) and explore its effectiveness on this newly developed data set.

5.3 Methodology: Creating a Diversity Corpus

In this section, we describe the corpus building procedure. It starts with an introduction to the source dataset. Then we proceed to the creation of an annotation pool and our approach for assigning the subtopics.

5.3.1 Source Dataset and Topic Selection

We collected tweets from the public Twitter stream between February 1, 2013 and March 31, 2013. The dates were chosen to coincide with the time interval of the TREC Microblog 2013 track[2]. In total, we have crawled 259,125,669 tweets. The statistics of this source dataset are shown in Table 5.1.

[2]TREC Microblog 2013 track: https://github.com/lintool/twitter-tools/wiki/TREC-2013-Track-Guidelines, accessed July 30th, 2014

Corpus	#Elements
Crawled Tweets	259,125,669
Selected News Events	50
Manual Annotation Entries	25,000
Effective Topics after Annotation	47
Subtopic Assignments	7,431
Subtopic Assignments per Topic	158.11

Table 5.1: Statistics of dataset for corpus building

After the crawl, in order to create topics, we consulted Wikipedia's *Current Events Portal*[3] for the months February and March 2013. The portal listed around 10 events for each day from which 50 news events were selected from the duration of 2 months. Furthermore, we derived the adhoc queries, which are short in length (see examples in Table 5.2), based on the description of the news events. We hypothesized that only topics with enough importance and more than local interests are mentioned here and thus, it is likely that our Twitter stream does contain some tweets which are pertinent to these topics. Another advantage of this approach is that we were able to also investigate the importance of time as we picked topics which are evenly distributed across the two-month time span.

5.3.2 Subtopic Annotation

Having defined the documents and topics, two decisions need to be made: (i) how to derive the subtopics for each topic, and (ii) how to create a pool of documents to judge for each topic (and corresponding set of subtopics). Previous benchmarks have developed different approaches for the former one. These approaches either derive subtopics post-hoc, i.e. after the pooling of documents for judgments has been created or rely on external sources such as query logs to determine the different interpretations and/or aspects of a topic. The setup followed by virtually all benchmarks is to create a pool of documents to judge based on the top retrieved documents by the benchmark participants, the idea being that a large set of diverse retrieval systems will retrieve a diverse set of documents for judging.

[3]Wikipedia Current Events Portal, http://en.wikipedia.org/wiki/Portal:Current_events, accessed April 9th, 2013

Annotation Pool Creation

Since in our work we do not have access to a wide variety of retrieval systems to create the pool, we opt for a different approach: we *manually* created complex Indri[4] queries for each topic. We consider this approach a valid alternative to the pool-based approach, as in this way we still retrieve a set of diverse documents. A number of examples are shown in Table 5.2 with the corresponding *Indri queries*. The Indri query language allows us to define, among others, synonymous terms within $< .. >$ as well as exact phrase matches with #1(...). The *#combine* operator joins the different concepts identified for retrieval purposes. Since we do not employ stemming or stopword removal in our retrieval system, many of the synonyms are spelling variations of a particular concept. The queries were created with background knowledge, i.e. where necessary, we looked up information about the news event to determine a set of diverse terms. The created Indri queries are then deployed with the query likelihood retrieval model. Returned are the top $10,000$ documents (tweets) per query. In a post-processing step we filter out duplicates (tweets that are similar with cosine similarity > 0.9 to a tweet higher in the ranking) and then present the top 500 remaining tweets for judgment to two annotators, denoted as *Annotator 1* and *Annotator 2*. After the manual annotation process, the duplicates are injected into the relevance judgments again with the same relevance score and subtopic assignment as the original tweet.

Subtopic Assignment

The annotators split the 50 topics among themselves and manually determined for each of the 500 tweets whether or not they belong to a particular subtopic (and which one). Thus, we did not attempt to identify subtopics beforehand, we created subtopics based on the top retrieved tweets. Intuitively, we create a subtopic whenever the tweet can answer a new question or fulfill some information need. Tweets which were relevant to the overall topic, but did not discuss one or more subtopics were considered non-relevant. For example, for the topic *Hillary Clinton steps down as United States Secretary of State* we determined the first tweet to be relevant for subtopic *what may be next for Clinton*, while the second tweet is non-relevant as it only discusses the general topic, but no particular subtopic:

[4]Indri is a query language supported by the Lemur Toolkit for Information Retrieval, http://www.lemurproject.org/, accessed July 30th, 2014.

News Event Topics	Manually created Indri queries	Adhoc queries	Identified Subtopics
Hillary Clinton steps down as United States Secretary of State	#combine(<#1(hillary clinton) #1(hilary clinton) #1(secretary clinton) #1(secretary of state)> <#1(steps down) #1(step down) leave leaves resignation resigns resign #1(stepping down) quit quits retire retires>)	*hillary clinton resign*	Clinton's successor what may be next for Clinton details of resignation Clinton's political positions
Syrian civil war	#combine(<syria syrian aleppo daraa damascus homs hama jasmin baniyas latakia talkalakh> <#1(civil war) war unrest uprising protest protests protestors demonstration demonstrators rebel rebels rebellion revolt revolts revolting resistance resisting resist clash clashes clashing escalation escalate escalated fight fights fighting battle battles offensive>)	*syria civil war*	casualties positions of foreign governments infighting among rebels
Boeing Dreamliner battery problems	#combine(<#1(Boeing Dreamliner) #1(boeing 787) #1(787 dreamliner)> <test tests testing tested check checks checked trial trials try> <battery batteries lithium-ion #1(lithium ion)>)	*dreamliner battery*	battery incidents cause of battery problems criticism Boeing tests

Table 5.2: Examples of selected news events, the corresponding manual and adhoc queries, and the identified subtopics

1. *Hillary Clinton transition leaves democrats waiting on 2016 decision. Hillary Clinton left the state department* $< URL >$.

2. *Clinton steps down as secretary of state. Outgoing us secretary of state Hillary Clinton says she is proud of* $< URL >$.

Thus, during the annotation process, we focused on the content of the tweet itself, we did not take externally linked Web pages in the relevance decision into account - we believe that this makes our corpus valuable over a longer period of time, as the content behind URLs may change frequently. This decision is in contrast to the TREC 2011 Microblog track, where URLs in tweets were one of the most important indicators for a tweet's relevance [158]. By following the method described above, we annotated 50 topics. The subtopics identified for a few example topics are listed as the last column in Table 5.3.

We note, that defining such subtopics is a subjective process. In other words, different annotators are likely to derive different subtopics for the same topic. However, this is a problem which is inherent to all diversity corpora which were derived by human annotators. In order to show the annotator influence, in the experimental section, we not only report the results across all topics, but also on a per-annotator basis.

Topic Refinement

At the end of the annotation process, we had to drop three topics, as we were not able to identify a sufficient number of subtopics for them. An example of a dropped topic is *2012-13 UEFA Champions League*, which mostly resulted in tweets mentioning game dates but little else. Thus, overall, we have 47 topics with assigned subtopics that we can use for our diversity retrieval experiments.

5.4 Topic Analysis

In this section, we perform a first analysis of the 47 topics and their respective subtopics. Where applicable, we show the overall statistics across all topics, as well as across the topic partitions according to the two annotators.

5.4.1 The Topics and Subtopics

In Table 5.3, we list the basic statistics over the number of subtopics identified, including the average, standard deviation, minimum, and maximum. Figure 5.1 shows concretely for each topic the number of subtopics. On average, we find 9 subtopics per topic. The large standard deviation indicates a strong variation between topics with respect to the number of subtopics (also evident in Figure 5.1). On a per annotator basis we also observe a difference in terms of created subtopics: *Annotator 1* has a considerably higher standard deviation than *Annotator 2*. This result confirms our earlier statement that the subtopic annotation is a very subjective task.

Topics	**All**	**Annotated by**	
		Annotator 1	***Annotator 2***
avg. #subtopics	9.27	8.59	9.88
s.d. #subtopics	3.88	5.11	2.14
min *#subtopics*	2	2	6
max *#subtopics*	21	21	13

Table 5.3: Statistics on subtopics numbers

The topics yielding the fewest and most subtopics, respectively, are listed as follows.

- *Kim Jong-Un orders preparation for strategic rocket strikes on the US mainland* (2 subtopics)
- *Syrian civil war* (21 subtopics)
- *2013 North Korean nuclear test* (21 subtopics).

We facilitated the annotation process with an annotation tools application on which the annotators spent on average 6.6 seconds on each tweet. For each tweet, the annotation effort is two-fold, including judging the relevance and identifying the subtopics that the tweet covers. The subtopics are created when no existing subtopic is appropriate. Hence, the total annotation effort amounted to 38 hours. However, the low average time spent per tweet was due to the considerably large number of non-relevant tweets. Apart from a very small number of tweets, each relevant tweet was assigned to exactly one subtopic, which is not surprising considering the small size of the documents.

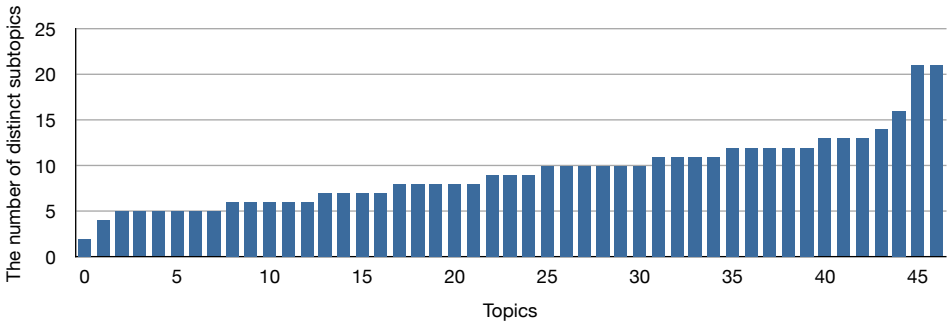


Figure 5.1: Number of subtopics identified for each topic

5.4.2 The Relevance Judgments

In Figure 5.2 we present the distribution of relevant and non-relevant documents among the 500 tweets the annotators judged per topic[5]. Twenty-five of the topics have less than 100 relevant documents, while six topics[6] resulted in more than 350 relevant documents. When considering the documents on the annotator level, we see a clear difference between the annotators: *Annotator 1* judged on average 96 documents as relevant to a topic (and thus 404 documents as non-relevant), while *Annotator 2* judged on average 181 documents as relevant. This again confirmed the subjectivity in annotators, where *Annotator 2* had been observed to be more lenient than *Annotator 1*.

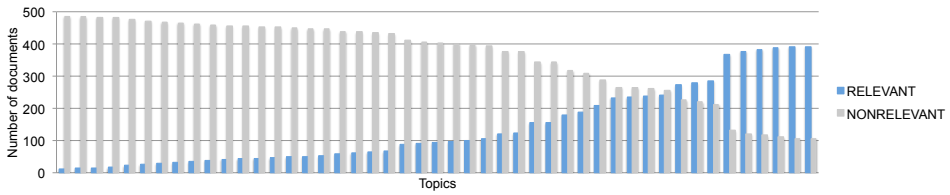


Figure 5.2: Number of tweets per topic identified as (non-)relevant during the annotation process.

In Chapter 3, we have found that the temporal recency to be a predicative feature for relevance estimation. The recent study by Efron et al. [54] provided support for the *temporal clustering hypothesis*, i.e. the relevant tweets

[5]As described earlier, the near-identical tweets, which were removed to ease the annotation load, are later added to the qrels again; they are not taken into account in the analysis presented here.

[6]Query with the identifier of AIRS007 "syria civil war", AIRS010 "Northern Mali Conflict", AIRS016 "horse meat scandal", AIRS018 "american airline merger", AIRS024 "Hugo Chávez", AIRS044 "Obama visit palestine israel"

tend to temporally cluster together. Therefore we also investigated the temporal distribution of the relevant tweets. In Figure 5.3 we plot for each topic the number of days that have passed between the first and the last relevant tweet in our data set. Since our data set spans a two-month period, we note that a number of topics are active the entire time (e.g. the topics *Northern Mali conflict* and *Syrian civil war*) while others are active for roughly 24 hours (e.g. the topics *BBC Twitter account hacked* and *Eiffel Tower, evacuated due to bomb threat*). We thus have a number of short-term topics and a number of long-term topics in our data set. In contrast to the TREC Microblog track 2011/12, we do not assign a particular query time to each topic (therefore we implicitly assume that we query the data set one day after the last day of crawling). We do not consider this a limitation, as a considerable number of topics are covered across weeks.

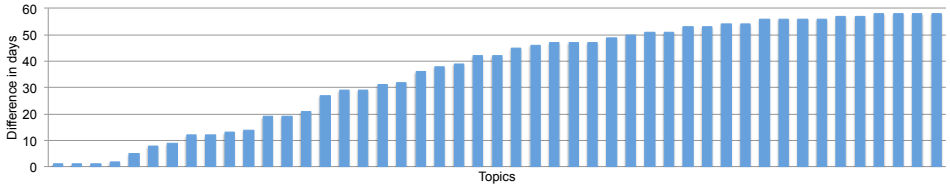


Figure 5.3: Difference in days between the earliest and the latest *relevant* tweet for each topic.

5.4.3 Diversity Difficulty

Lastly, we consider the extent to which the search results can actually be diversified. Diversification does not only depend on the ambiguity or the underspecification of the query, it is also limited by the amount of diverse content *available in the corpus*. Golbus et al. [64] recently investigated this issue and proposed the *diversity difficulty* measure (*dd*) which is a function of two factors: the amount of diversity that a retrieval system can achieve at best and the ease with which a retrieval system can return a diversified result list. Intuitively, a topic has little inherent diversity if the maximum amount of diversity a retrieval system can achieve is small. A topic is considered "somewhat more diverse" by Golbus et al. in the case where a diverse result list can be achieved but it is difficult for the system to create one. A topic has a large amount of diversity if a retrieval system not tuned for diversity is able to return a diverse result list. These intuitions are formalized in a diversity formula with $dd \in [0, 1]$. A large score ($dd > 0.9$) indicates a diverse query, while a small score ($dd < 0.5$) either indicates a topic with few subtopics or

a fair number of subtopics which are unlikely to be discovered by an untuned retrieval system. In Table 5.4 we present the diversity difficulty average and standard deviation our topics achieve, which are very similar for both annotators and also in line with the diversity difficulty scores of the TREC 2010 Web diversity track [64]. We thus conclude, that in terms of diversity difficulty, our topic set presents a well constructed data source for diversity experiments.

Statistics of *dd* (diversity difficulty)	**All topics**	**Topics assigned to** ***Annotator 1***	***Annotator 2***
avg. dd	0.71	0.72	0.70
s.d. dd	0.07	0.06	0.07

Table 5.4: Diversity difficulty scores across all topics

Finally, we observe that the diversity difficulty score of *long-term topics*, that is topics whose first and last relevant tweet cover at least a 50 day timespan, is higher ($dd_{long\text{-}term} = 0.73$), than the diversity difficulty score of *short-term* topics (the remaining topics) where $dd_{short\text{-}term} = 0.70$.

5.5 Diversification by De-Duplication

Having analyzed our corpus, we will now explore the diversification effectiveness of the de-duplication framework for microblogs in Chapter 4 on this data set.

5.5.1 Duplicate Detection Strategies on Twitter

In Chapter 4, it was found that about 20% of search results returned by a standard adhoc search system contain duplicate information. This finding motivated the development of a de-duplication approach which detects duplicates by employing (i) ***Sy***ntactic features, (ii) ***Se***mantic features, and (iii) ***Co***ntextual features in a machine learning framework[7]. By combining these feature sets in different ways, the framework supports mixed strategies named after the prefixes of the feature sets used: **Sy**, **SySe**, **SyCo**, and **SySeCo**. Not surprisingly, the evaluation showed that the highest effectiveness

[7]The work in Chapter 4 also considers the use of features derived from Web pages linked to in tweets. We ignore these features, as we did not consider URL content in the annotation process.

was achieved when all features were combined.

Given an initial ranking of documents (tweets), each document starting at rank two is compared to all higher ranked documents. The duplicate detection framework is run for each document pair and if a duplicate is detected, the lower ranked document is filtered out from the result ranking.

5.5.2 Diversity Evaluation Measures

As researchers have been studying the diversification problem intensively on the Web, a number of measures have been proposed over the years to evaluate the success of IR systems in achieving diversity in search results. We evaluate our de-duplication experiments according to the measures listed as follows.

α-(n)DCG [40] This measure was adopted as the official diversity evaluation measure at TREC 2009 [41]. It is based on Normalized Discounted Cumulative Gain (nDCG) [78] and extends it by making the gain of each document dependent on the documents ranked above it.

Precision-IA [8] This measure means the ratio of relevant documents for different subtopics within the top-k items.

Subtopic-Recall [188] The subtopic recall (in short **S-Recall**) is calculated as the number of subtopics covered by the top-k documents. The measure ranges from 0 to 1, where larger values indicate a better coverage of subtopics.

Redundancy The measure shows the ratio of repeated subtopics among all relevant documents within the top-k ranked documents. For diversity experiments, a lower redundancy value indicates a better performance.

Besides the measure of α-(n)DCG, which was adopted by TREC as the official diversity evaluation measure in 2009, the other three measures had been selected to evaluate the influences of our de-duplicate strategies. While applying the de-duplicate strategies, we expect to see the duplicate tweets to be removed and lower the redundancy. Hence, there is a chance that the removed duplicates yield to the novel contents. If this is the case, the subtopic-recall increases. However, it is possible that no more relevant tweets will be appended to the tail. For this reason, we check whether the precision decreases. Meanwhile, we can also see if the false positives in the duplicate detection have an significant impact on the retrieval effectiveness.

5.5.3 Analysis of De-Duplication Strategies

We evaluate the different de-duplication strategies from two perspectives: (i) we compare their effectiveness on all 47 topics, and, (ii) we make side-by-side comparisons between two topic splits, according to the annotator and the temporal persistence. This enables us to investigate the annotator influences and the differences in diversity between long-term and short-term topics.

Apart from the de-duplication strategies, we also employ three baselines. The **Automatic Run** is a standard query likelihood based retrieval run (language modeling with Dirichlet smoothing, $\mu = 1000$) as implemented in the Lemur Toolkit for IR. The run **Filtered Auto** builds on the automatic run by greedily filtering out duplicates by comparing each document in the result list with all documents ranked above it. In practice, we remove the tweet if it has a cosine similarity above 0.9 with any of the higher ranked documents. The de-duplication strategies are also built on top of the Automatic Run by filtering out documents (though in a more advanced manner). All these runs take the adhoc queries, i.e., very short keyword queries, as input (see examples shown in Table 5.2). The only exception to this rule is the **Manual Run** which is actually the run we derived from the manually created complex Indri queries that we used for annotation purposes with cosine-based filtering as defined above.

Overall Comparison

In Table 5.5 the results for the different strategies averaged over all 47 topics are shown. Underlined is the best performing run for each evaluation measure. It should be noted that a low score in redundancy means a high effectiveness. Statistically significant improvements over the *Filtered Auto* baseline are marked with † (paired t-test, two-sided, $\alpha = 0.05$) for α-nDCG, Precision-IA and S-Recall. The *Manual Run*, as expected, in general yields the best results which are statistically significant in all measures at level @20.

We find that the de-duplication strategies *Sy* and *SyCo* in general outperform the baselines *Automatic Run* and *Filtered Auto*, though the improvements are not statistically significant. We observe that Precision-IA degrades as the de-duplication strategies take *Automatic Run* as input, especially for Precision-IA@20. This confirms the analysis we have in Section 5.5.2, i.e., the removed duplicates do not necessarily give way to the relevant tweets. On the other hand, in terms of redundancy, the de-duplication strategies perform best. De-duplication strategies that exploit semantic features (*SySe* and

SySeCo) show a degraded effectiveness in terms of α-nDCG and Precision-IA. Therefore, we see here that low in redundancy does not necessarily mean higher diversity.

Influence of Annotator Subjectivity and Temporal Persistence

In Table 5.6, the results are shown when splitting the topic set according to the annotators. Due to the small topic size, significance tests were not performed. Here we find that although the absolute scores of the different evaluation measures for *Annotator 1* and *Annotator 2* are quite different, the general trend is the same for both. The absolute α-nDCG scores of the various de-duplication strategies are higher for *Annotator 2* than for *Annotator 1*, which can be explained by the fact that *Annotator 2*, on average, judged more documents to be relevant for a topic than *Annotator 1*. The opposite observation holds for the *Manual Run*, which can be explained by the inability of cosine filtering to reduce redundancy. Given that there are more relevant documents for *Annotator 2*'s topics, naturally the redundancy problem is more challenging than for *Annotator 1*'s topics.

Finally, Table 5.7 shows the results when comparing short-term and long-term queries. For long-term topics, the de-duplication strategies consistently outperform the baselines, while the same cannot be said about the short-term topics. We hypothesize that short-term topics do not yield a large variation in vocabulary (often a published news report is repeated in only slightly different terms) so that features which go beyond simple term matching do not yield significant benefits. Long-term topics on the other hand develop a richer vocabulary during the discourse (or the course of the event) and thus more complex syntactic features can actually help.

5.6 Discussion

In Chapter 2, we introduced the prototype search engine Twinder as the playground for conducting various analytical tasks. The quality of the retrieved microposts can be improved by our findings in relevance estimation and duplicate detection from Chapter 3 and Chapter 4, respectively. In this chapter, we presented our efforts to explore the diversity among tweets in the setting of microblog search. Specifically, we create a microblog-based corpus for search result diversification experiments.

To tackle the problem and the research questions that we introduced in

Measure	α-nDCG		Precision-IA		S-Recall		Redundancy		#Judged	
	@10	@20	@10	@20	@10	@20	@10	@20	@10	@20
Automatic Run	0.312	0.338	0.079	0.075	0.315	0.413	0.471	0.580	7.57	14.83
Filtered Auto	0.339	0.358	0.079	0.072	0.370	0.454	0.380	0.514	5.57	10.06
Sy	0.347	0.362	0.080	0.066	0.382	0.457	0.358	0.497	7.44	14.08
SySe	0.340	0.357	0.075	0.063	0.363	0.452	0.357	0.481	7.34	13.74
SyCo	0.346	0.360	0.080	0.065	0.381	0.464	0.371	0.478	7.48	14.06
SySeCo	0.341	0.358	0.077	0.064	0.365	0.457	0.376	0.489	7.40	14.02
Manual Run	0.386	0.443†	0.104†	0.099†	0.446	0.623†	0.482	0.601	10	20

Table 5.5: Comparison of different de-duplication strategies on 47 diversity topics. The number of documents covered by our judgment are listed in the last two columns. Statistically significant improvements over the *Filtered Auto* baseline are marked with † (paired t-test, two-sided, $\alpha = 0.05$) for α-nDCG, Precision-IA and S-Recall. The Redundancy measure performs best when it is lowest. For each measure, the best achieved performance is underlined.

Measure	α-nDCG @10	α-nDCG @20	Precision-IA @10	Precision-IA @20	S-Recall @10	S-Recall @20	Redundancy @10	Redundancy @20
Annotator 1								
Automatic Run	0.298	0.325	0.085	0.078	0.317	0.405	0.512	0.563
Filtered Auto	0.317	0.337	0.083	0.073	0.366	0.425	0.361	0.497
Sy	0.321	0.344	0.085	0.069	0.366	0.448	0.365	0.518
SySe	0.315	0.337	0.079	0.060	0.366	0.447	0.375	0.477
SyCo	0.318	0.346	0.086	0.067	0.359	0.466	0.339	0.464
SySeCo	0.321	0.344	0.083	0.062	0.358	0.466	0.362	0.460
Manual Run	0.442	0.489	0.127	0.111	0.537	0.667	0.451	0.582
Annotator 2								
Automatic Run	0.325	0.350	0.074	0.073	0.314	0.420	0.444	0.593
Filtered Auto	0.362	0.381	0.076	0.072	0.379	0.479	0.393	0.526
Sy	0.371	0.377	0.075	0.064	0.395	0.466	0.352	0.482
SySe	0.362	0.374	0.072	0.065	0.360	0.456	0.372	0.493
SyCo	0.371	0.373	0.075	0.063	0.400	0.462	0.369	0.482
SySeCo	0.359	0.371	0.073	0.066	0.371	0.448	0.386	0.509
Manual Run	0.338	0.403	0.087	0.090	0.367	0.583	0.505	0.615

Table 5.6: Comparison of different de-duplication strategies between annotators. For each measure, the best achieved performance is underlined.

Measure	α-nDCG @10	α-nDCG @20	Precision-IA @10	Precision-IA @20	S-Recall @10	S-Recall @20	Redundancy @10	Redundancy @20
Long-term Topics								
Automatic Run	0.346	0.386	0.074	0.075	0.336	0.494	0.518	0.597
Filtered Auto	0.387	0.415	0.075	0.072	0.431	0.560	0.371	0.518
Sy	0.400	0.419	0.077	0.069	0.458	0.558	0.336	0.499
SySe	0.389	0.414	0.072	0.066	0.421	0.548	0.354	0.493
SyCo	0.401	0.416	0.078	0.068	0.459	0.554	0.358	0.486
SySeCo	0.386	0.412	0.074	0.069	0.417	0.545	0.376	0.501
Filtered Manual	0.373	0.431	0.084	0.087	0.416	0.596	0.457	0.619
Short-term Topics								
Automatic Run	0.293	0.311	0.082	0.075	0.304	0.367	0.437	0.571
Filtered Auto	0.312	0.326	0.081	0.072	0.336	0.393	0.402	0.510
Sy	0.318	0.329	0.081	0.065	0.338	0.400	0.388	0.495
SySe	0.312	0.325	0.077	0.061	0.330	0.397	0.375	0.464
SyCo	0.315	0.329	0.081	0.063	0.337	0.413	0.396	0.471
SySeCo	0.316	0.328	0.080	0.061	0.335	0.407	0.391	0.472
Manual Run	0.391	0.448	0.116	0.106	0.464	0.638	0.492	0.590

Table 5.7: Comparison of different de-duplication strategies between topics of long/short-term. For each measure, the best achieved performance is underlined.

the beginning of this chapter, we manually created subtopic-based relevance

judgments for a total of 47 topics and evaluated the suitability of the created data set across several dimensions. The summary of our contributions is listed in Table 5.8.

Research Question	**Summary of Findings**
How can we build a microblog corpus for search result diversification?	▶ We crawled the Twitter public stream for a duration of two-month as the document source for the corpus. ▶ 50 news events were selected and we manually created the corresponding adhoc queries and complex queries. ▶ For each topic, we annotated 500 tweets with subtopic assignments. ▶ We made our corpus publicly available at http://ktao.github.io/phd/#datasets.
How suitable is the corpus that we created for research on search result diversification?	▶ We found the subjectivity in annotators by analyzing the subtopics annotated for the topics and the relevance judgments. ▶ In terms of the *diversity difficulty* measure, the corpus that we built is in line with the TREC 2010 Web diversity track while the agreement on this measure was also found between annotators.
To what extent can we achieve diversity by applying the developed de-duplication strategies?	▶ The de-duplication strategies in general reduced the redundancy in subtopics, at the cost of slight decrease in diversity. ▶ The comparison study on temporal persistence and topic recency indicated the importance of the feature suitability for the topic of different types.

Table 5.8: Overview on research questions investigated in Chapter 5

The comprehensive analysis of the corpus showed its suitability for this

purpose. The *diversity difficulty* measure showed that the dataset we created has a similar level of diversification potential as previous diversity Web tracks at TREC. The analyses of the annotators' influence on subtopic creation and relevance judgments revealed considerable subjectivity in the annotation process. At the same time though, the de-duplication retrieval experiments showed that the observed trends with respect to the different evaluation measures were largely independent of the specific annotator. The performance of the de-duplication strategies and their comparison to the results reported in Chapter 4 indicate the importance of the feature suitability for the topic type (long-term vs. short-term topics and topic recency). There are existing works for diversifying the Web search results based on explicit query reformulations such as xQuAD [135]. However, we did not opt for this approach as it relies on feedback from Web search engines.

The outcomes of this chapter provide researchers with input for their research of search results diversification on microblogging sites. For instance, the diversification strategies that performed well in the Web search setting, e.g. [138, 139], could be experimented with on this microblogging corpus. In order to provide insights for further corpus building work, the impact of the different strategies and the annotator subjectivity can be analyzed in a more intensive way. For example, we could investigate the potential sources (influences and/or motivations) for the observed annotator differences.

Chapter 6

Twitcident: Fighting Fire with Social Web Data Analytics

We have introduced the Twitter Analytical Platform in Chapter 2, which allows for customizing analytical workflows with Social Web data in the Twitter Analysis Language. Given the context of information retrieval on Twitter, the proposed platform enable us to investigate three aspects: relevance, redundancy, and diversity (see Chapters 3 to 5). In this chapter, we look at real-life challenges in order to investigate how the Twitter Analytical Platform can support an application in production. Hence, we introduce Twitcident, a system that relies on the Twitter Analytical Platform to automatically filter relevant information about a real-world incident from Twitter streams and make the information accessible and findable in the given context of the incident. Besides showing the data processing capabilities provided by the Twitter Analytical Platform, we also present and evaluate the dependent applications of faceted search and visualized analytics. The main contributions of this chapter have been published in [5, 6].

6.1 Introduction

During crisis situations such as large fires, storms or other types of incidents, people nowadays report and discuss their observations, experiences, and opinions in their Social Web streams. Therefore, valuable information that is of use for both emergency services and the general public is avail-

able online. Recent studies have shown that data from the Social Web and particularly Twitter helps to detect incidents and topics [111, 133, 180] or to analyze afterwards the information streams that people generated about a topic [60, 93, 130]. However, there is a lack of systematic solutions to fulfill information needs during these incidents. Therefore, we formulate the problem to be tackled by Twitcident as follows.

Problem 5 (Information Exploration during Incidents) *Given an incident, the task of information exploration is to provide the relevant information from the Social Web to concerning parties while making the exploration of such information efficient.*

In this chapter, we address this problem by tackling two fundamental challenges with two kinds of support provided by the Twitter Analytical Platform: (i) automatically filtering relevant information from Social Web streams and (ii) making the information accessible and findable in the given incident context. More specifically, we present how we exploit the tools provided by our platform for building Twitcident[1], a framework for filtering, searching and analyzing Twitter information streams during incidents.

With the functions provided by the Twitter Analytical Platform, we approach these challenges by enriching the semantics of short messages by named entity recognition, tweet classification, as well as linkage to related external Web resources. Semantic enrichment also builds the basis for the search and analytics functionality that is provided by Twitcident. Given the semantically enriched Social Web content about an incident, we allow users to explore the information along different types of information needs (e.g. damage, casualties). These types can be seen as the different facets, of which the meaning has been defined as *a set of meaningful labels organized in such a way as to reflect the concepts relevant to a domain* by Hearst [71]. The strategies can be developed to recommend content along different facets that facilitate the information exploration process. Therefore, we integrate faceted search strategies [1] that go beyond traditional keyword search as offered by Twitter[2] or topic-based browsing as proposed by Bernstein et al. [17]. Moreover, users can overview information by exploiting Twitcident's real-time analytics to get an understanding of how different types of information are posted over time.

The main contributions of this chapter can be summarized as follows.

[1] http://twitcident.com, accessed July 30th, 2014
[2] http://twitter.com/search, accessed July 30th, 2014

- We introduce a framework for incident-driven information filtering and search on Social Web streams. Our framework features automated incident profiling, aggregation, semantic enrichment, and filtering, which are implemented in the Twitter Analysis Language. Furthermore, it provides advanced search and analytics functionality that allows users to find and understand relevant information (Section 6.3).
- We propose and evaluate strategies for solving two fundamental research challenges: (1) information filtering and (2) exploration on Social Web streams.
    1. We compare different stream filtering strategies on a large Twitter corpus and show that the semantic filtering strategies of our Twitcident framework lead to major improvements compared to keyword-based filtering (Section 6.4).
    2. We employ faceted search strategies that enable users to find relevant information in Social Web streams. Our evaluation confirms that the semantic faceted search strategies, which are applied on top of the filtered streams, enhance the efficiency of information exploration significantly compared to keyword-based search. Contextualization (adapting to the temporal context of a search activity) and personalization (adapting to the interests of the user who performs the activity) yields further improvements (Section 6.5).
- We apply our Twitcident system to incidents that happen during everyday life (mainly targeted towards the Netherlands) and discuss experiences and insights we gained from running Twitcident in practice (Section 6.6).

6.2 Related Work

In the last decade, Social Web platforms such as Twitter provide researchers a rich source for studying problems related to popular events, such as elections [60], sport competitions [128], and natural disasters [175, 180] with data from Social Web streams. For example, a study on the U.S. midterm election in 2010 by Livne et al. [103] claimed that the proposed model could predict the election. However, other researchers argued such predictions do not work neither for a particular political system [82] nor in general [115]. Instead of ambitious attempts to predict election results, Lietz et al. [97] analyzed the microblogging behaviour of politicians during the German federal election

in 2013 and found common patterns as well as unique characteristics. With respect to sport events, Steiner et al. [149] presented a case study on the 2014 Winter Olympics with a live monitor [151] for bursting event detection from Wikipedia editing streams and an automatic tool [150] for creating media galleries of the given event. Rios et al. [128] presented their analytics of the 2010 World Cup in a visualized way by distilling massive amounts of Twitter data. During natural disasters, Social Web streams can be used as evidence for sending earthquake alerts [133]. Vieweg et al. [175] suggested that tweets can be helpful for enhancing situational awareness by analyzing data on the Red River Floods and Oklahoma Grassfires in 2009.

A number of research efforts have also considered more generally how to improve search and retrieval on Social Web streams. Marcus et al. [109] studied how to visualize Twitter streams. Bernstein et al. [17] proposed a topic-based browsing interface for Twitter in which a user can navigate through her personal Twitter stream by means of tag clouds. Given the enormous amount of messages posted in Social Web streams, how these data can satisfy information needs of individual users become a non-trivial challenge. In fact, Teevan et al. [163] confirmed studies that emphasize Twitter's role as news source [91, 134] and revealed that there are significant differences in the search behavior on Twitter compared to traditional Web search: Twitter users are specifically interested in information related to events and often use the rudimental search functionality of Twitter to monitor search results. With Twitcident, we introduce a framework that automates the process of monitoring relevant information published in Social Web streams and therefore reduces the efforts that users need to invest to satisfy their information needs. On top of the automatically filtered streams, Twitcident provides faceted search functionality as introduced in previous work [1].

6.3 Twitcident

In this section, we will overview the architecture of the Twitcident framework and detail its key components that allow for filtering, searching and analyzing of information available in Social Web streams. The Web-based front-end of the Twitcident system is depicted in Figure 6.2 and allows users, such as policemen, firefighters, and mass event organizers, to explore and analyze information from Social Web streams during incidents such as natural disasters, fires or other types of emergency events.

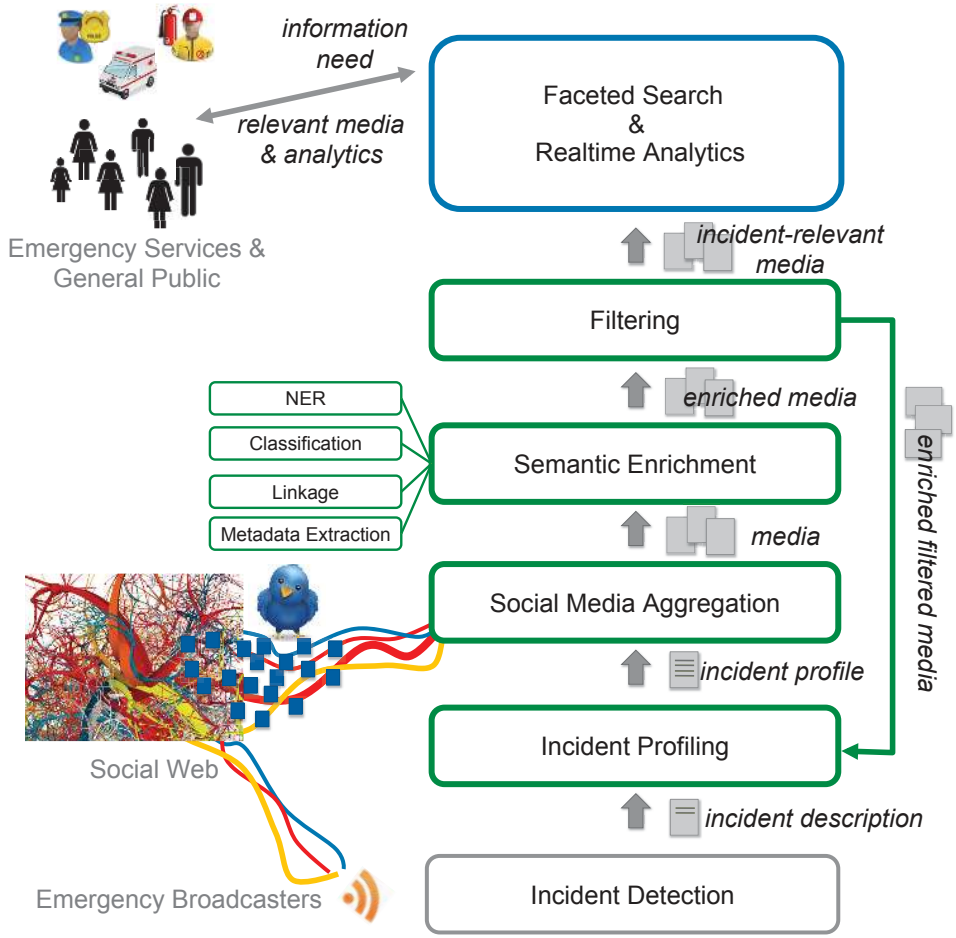


Figure 6.1: Architecture: (i) *incident profiling* and *filtering* of social media that is relevant to an incident (green boxes) and (ii) *faceted search* and *realtime analytics* to explore and overview the media (blue box). Both types of components benefit from *semantic enrichment*.

6.3.1 Architecture

The Twitcident framework architecture is summarized in Figure 6.1. The core framework functionality is triggered by an incident detection module that listens for incidents being broadcast by emergency services. Whenever an incident is detected, Twitcident starts a new thread for profiling the incident and aggregating social media and Twitter messages. The collected messages are processed by the semantic enrichment module which features named entity recognition (NER), classification of messages into different facets representing various aspects of information needs, linkage of messages to external

Figure 6.2: Screenshot of the Twitcident system: (a) search and filtering functionality to explore and retrieve particular Twitter messages, (b) messages that are related to the given incident (here: fires in Texas) and match the given query of the user and (c) real-time analytics of the matching messages.

Web resources and further metadata extraction. The semantic enrichment is one of the key enabling components of the Twitcident framework as it (a) supports semantic filtering of Twitter messages to identify those tweets that are relevant for a given incident, (b) allows for faceted search on the filtered media and (c) gives means for summarizing information about incidents and providing real-time analytics.

In the Twitcident system, both faceted search and real-time analytics are made available to client users via a graphical user interface that is displayed in Figure 6.2. The search functionality allows end-users to further filter messages about an incident while analytics deliver diagrams and gadgets that enable users to analyze and overview how people report about the incident on the Social Web. We now discuss each of the components of our architecture in detail.

6.3.2 Incident Detection

To detect incidents, the Twitcident system relies on emergency broadcasting services. In the Netherlands, incidents which require the police, fire department, or other public emergency services to take an action and which are of interest to the general public, are immediately published via the P2000 communication network. These published messages describe what type of incident has happened, where and when it happened, and also what scale the incident is classified as. The P2000 refreshes the information every minute and may have a delay of 30 seconds[3]. Figure 6.3(a) shows an example P2000 message concerning a large fire incident that happened in the city of Moerdijk, the Netherlands[4]. The figure visualizes the automatic workflow that is triggered whenever a new incident is reported. For a given incident it may happen that several P2000 messages are broadcast which requires Twitcident to first perform duplicate detection before starting a new incident monitoring thread. Therefore, the incident detection component compares the location, starting time and type of the newly reported incident with the incidents that are already monitored by Twitcident. If a new incident has been detected the Twitcident framework translates the broadcast message into an initial incident profile that is applied as query to collect relevant messages from the Social Web and Twitter in particular. All incidents that are monitored by the Twitcident system are listed on the dashboard that is depicted in Figure 6.3(b).

6.3.3 Incident Profiling and Filtering

While monitoring an incident, Twitcident continuously adapts the incident profile to improve the filtering of messages. This process is realized via the following components (see Figure 6.1): (i) incident profiling, (ii) social media aggregation, (iii) semantic enrichment and (iv) filtering.

Incident Profiling

Based on the initial incident description and the collected, enriched Social Web messages, the incident profiling module generates an incident profile that is used to refine the media aggregation and the filtering. An incident

[3] http://www.p2000-online.net, accessed July 30th, 2014

[4] http://nl.wikipedia.org/wiki/Brand_Moerdijk_5_januari_2011, accessed July 30th, 2014

profile is a set of weighted facet-value pairs that describe the characteristics of the incident:

Definition 2 (Incident Profile) *An* incident profile *of an incident $i \in I$ is a set of tuples $((f,v), w(i,(f,v)))$ where (f,v) is a facet-value pair that describes a certain characteristic f of the incident and $w(i,(f,v))$ specifies the importance of the facet-value pair for the incident that is computed by a weighting function w:*

$$P(i) = \{((f,v), w(i,(f,v))) | \\ (f,v) \in FVPs, i \in I, w(i,(f,v))) \in [0..1]\} \quad (6.1)$$

Here, $FVPs$ and I denote the set of facet-value pairs and incidents respectively. A facet-value pair characterizes a certain attribute (facet) of an incident with a certain value. Twitcident allows for various types of facets including locations, persons, incident classes or keywords. Therefore, the aforementioned fire that happened in Moerdijk may have the following incident profile: *$P(i_{moerdijk})$ = {((location, Moerdijk), 1.0), ((location, Dordrecht), 0.73), ((type, Fire), 1.0), …}*, as shown in Table 6.1.

Facet	**Value**	**Weight**
location	Moerdijk	1.0
location	Dordrecht	0.73
type	Fire	1.0
...	...	...

Table 6.1: The example incident profile for the fire in Moerdijk

The weight that is associated with each facet-value pair ranges between 0 and 1: the higher the weight, the more important the facet-value pair for the incident. We apply the relative occurrence frequency as basic weighting strategy, i.e. the fraction of messages about the incident that mention the given facet-value pair. Incident profiles are continuously updated to adapt to topic changes that arise within an incident. To prevent topic drift, we combine the current profile with the initial incident profile following a classical mixture approach: $P(i) = \lambda P_{initial}(i) + (1-\lambda) P_{current}(i)$ where we experimented with $\lambda \in [0..1]$ ranging between 0.25 and 0.5.

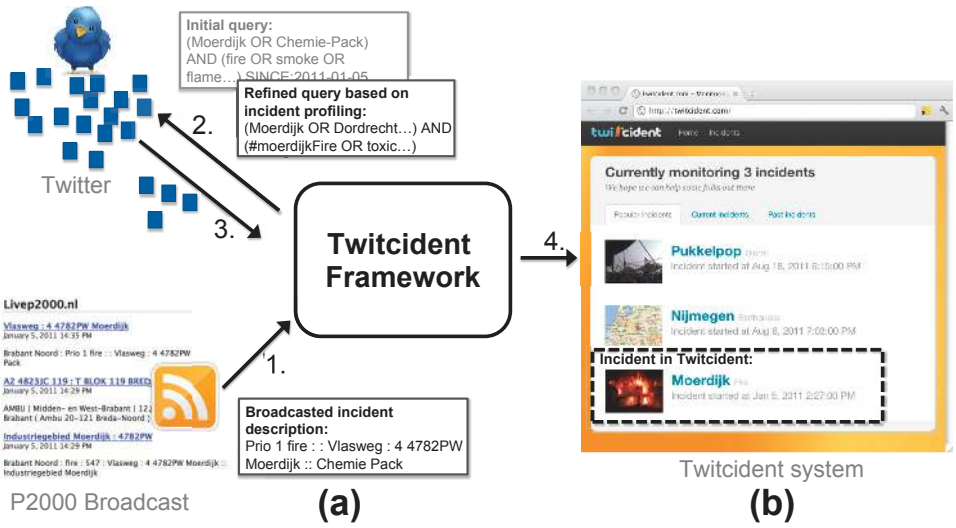


Figure 6.3: Incident detection: (1) as soon as an incident is broadcasted via the P2000 network, the Twitcident framework (2) transforms the encoded P2000 message into an initial incident query to (3) collect Twitter messages that are possibly relevant for the incident so that (4) information about the incident can be accessed via the Twitcident system. Over time, the incident profiling effects refinements of the queries that are used to collect tweets. The screenshot shows the dashboard of popular incidents that are (and have been) monitored by Twitcident.

Social Media Aggregation

Based on the incident profiling, the Twitcident system exploits the social media aggregation component to collect Twitter messages as well as related pictures and videos that are posted on platforms such as Twitpic[5] or Twitvid[6] respectively. Twitcident utilizes the data collection function provided by Twitter Analytical Platform (see Section 2.6.1). Therefore, we can use either search or streaming data, supported by the Twitter Analysis Language, to collect messages. The search function allows for querying Twitter messages that have been indexed by the Twitter Analytical Platform and therefore enable Twitcident to collect those incident-related tweets that have been posted before Twitcident detected the incident. The streaming acquisition function does not allow for querying previously published tweets but allows Twitcident to continuously listen for current tweets that mention keywords

[5]http://twitpic.com, accessed July 30th, 2014
[6]http://twitvid.com, accessed November 7th, 2011

related to an incident.

Semantic Enrichment

Based on the Twitter Analytical Platform, the aggregated Social Web content (Twitter messages) is processed by the semantic enrichment component (see Section 2.6.5) of Twitcident which features the following functionality.

NER The NER module assembles the different services supported by TAL for detecting entities such as persons, locations or organizations that are mentioned in tweets (see Section 2.6.5). As those entity recognition services only function for limited languages such as English, Twitcident provides the option to translate tweets into English, which is usually well supported by NER services[7]. Besides, one can also take advantage of the internationalization efforts of NER services [45]. In practice, we have found that translating Dutch into English worked well for our application purposes and thus were used in the rest of this chapter. The extracted entities are mapped to concepts in DBpedia [20], the RDF representation of Wikipedia, and the type of an entity is utilized to specify the facet of the corresponding facet-value pair. For example, given a Twitter message such as *"#txfire is approaching Austin, 50 houses destroyed already http://bit.ly/3r6fgt"*, the NER module allows for detecting the facet-value pair *"(location, dbpedia:Austin_Texas)"*[8].

Classification Twitcident classifies the content of Twitter messages into reports about casualties, damages or risks and also categorizes the type of experience that is reported in a tweet, e.g. whether the publisher of a tweet is seeing, feeling, hearing or smelling something. In TAL, the classification can be either defined as the hand-crafted rules that operates on both the facet-value pairs and the plain words that are mentioned in a tweet for constructing new attributes within tuples, or implemented by utilizing the machine learning function, which allows for more complex models.

Linkage By following links that are posted within messages, Twitcident further contextualizes the semantics of a message. Therefore, the semantic enrichment module extracts the main content of the Web resource that is referenced from a tweet by invoking the *External Link Crawler* of TAL (see Section 2.6.3) and processes it via the NER module to further enrich the Twitter

[7] http://code.google.com/apis/language/translate/overview.html, accessed July 30th, 2014

[8] The namespace abbreviation "dbpedia" points to: http://dbpedia.org/resource/

message with facet-value pairs that describe its content. For the aforementioned tweet which lists *"http://bit.ly/3r6fgt"*, one may extract additional facet-value pairs such as *"(location, dbpedia:Bastrop_Texas)"* or *"(organization, dbpedia:Texas_Forrest_Service)"*.

Metadata extraction Twitcident can also make use of the metadata that TAL collects and infers from Twitter messages, such as pictures referenced from the tweet or background information about the publisher of a tweet. Other types of metadata may include the profile picture, number of followers, number of tweets published during the incident or the location of the user when publishing her tweets. Such provenance data is important for end-users to assess the trustworthiness of a tweet and is moreover exploited by the Twitcident system when tweets that match the current query are sorted according to their relevance (see the search in Figure 6.2(a)).

Enriched Twitter messages can therefore also be represented by means of a set of weighted facet-value pairs. In line with Definition 2, the profile $P(t)$ of a Twitter message $t \in T$ can therefore specified as: $P(t) = \{((f,v), w(t,(f,v)))| (f,v) \in FVPs, t \in T, w(t,(f,v))) \in [0..1]\}$. From a implementation perspective, the profile $P(t)$ can be represented by customized attributes constructed by developers in TAL.

Filtering

The goal of the filtering step is to identify those tweets that are relevant to an incident. Therefore, the Twitcident filtering component first detects the language of a Twitter message and filters out all tweets that do not match the target language(s). In the deployed Twitcident system, as the incidents being monitored are located in the Netherlands, we only consider Dutch or English tweets as relevant and discard Twitter messages for which we detect another language. Based on this pre-processing, the Twitcident framework features two core filtering strategies: (i) semantic filtering and (ii) semantic filtering with news contextualization. In TAL, these strategies can be implemented by utilizing the *Filtering* function, which usually needs to make use of a series of analytical tools in the functionality stack for preparing calculation results and consequently a boolean value in advance.

Semantic Filtering Given the current incident profile $P(i)$ and the set of semantically enriched Twitter messages $P(t)$, the core challenge is to decide whether a tweet t is relevant for an incident i. The semantic filtering strategy

therefore exploits the set of alternative labels of a DBpedia URI v that is mentioned in the facet-value pairs (f, v) of $P(i)$. If an alternative label is mentioned in the content of a Twitter message t then the corresponding facet-value pair (f, v) is added to the tweet profile. Given the further enriched tweet profile—denoted as $\bar{P}(t)$—and $P(i)@k$, the top-k weighted facet-value pairs of the incident profile $P(i)$, the semantic filtering strategy computes the similarity between $P(i)@k$ and $\bar{P}(t)$ and considers a tweet t relevant to an incident i if $filter_{sem}(P(i), P(t)) = 1$:

$$filter_{sem}(P(i), P(t)) = \begin{cases} 1 & \text{if } sim(P(i)@k, \bar{P}(t)) > \delta \\ 0 & \text{otherwise} \end{cases} \tag{6.2}$$

In our experiments in Section 6.4, we use $P(i)@20$, apply the Jaccard similarity coefficient to compute $sim(P(i), \bar{P}(t))$ and set $\delta = 0$ as threshold. A Twitter message t is thus relevant if at least one facet-value pair of $P(i)@k$ also occurs in $\bar{P}(t)$.

Semantic Filtering with News Context As Twitter users might be influenced by public news media, Twitcident also monitors popular news agencies. The semantic filtering with news contextualization therefore extends the semantic filtering by enriching the incident profile $P(i)$ with information from mainstream news media before generating $\bar{P}(t)$. In particular, $P(i)$ is complemented with facet-value pairs that are extracted from related news articles. A news article is considered to be related to an incident if it matches the initial incident profile $P(i)$. The expanded incident profile $\bar{P}(i)$ is then used to perform the semantic filtering as described above. A tweet t is considered to be relevant to an incident i if $sim(\bar{P}(i)@k, \bar{P}(t)) > \delta$.

6.3.4 Faceted Search and Analytics

Incident detection, incident profiling, media aggregation, semantic enrichment and filtering are automatic processes that deliver information about an incident as reported by people on the Social Web. However, in order to find information in the filtered Social Web streams, appropriate functionality for search and analysis has to be engineered as well. The Twitcident framework approaches the challenge of retrieving relevant information from Social Web streams by means of faceted search as proposed in [1]. In this section, we re-visit the different faceted search strategies provided by the Twitcident framework and detail Twitcident analytics. In TAL, the ranking of these strategies can be implemented as assigning a score which is derived

from the attributes of the tuples representing the Twitter messages. In this way, Twitcident can render the items in descending order of their scores.

Faceted Search Strategies

The faceted search functionality allows users to further filter incident-related messages by selecting facet-value pairs that should be featured in the retrieved messages. A faceted query q thus may consist of several facet-value pairs. For example, one may narrow down to the Twitter messages that are collected for the Moerdijk fire and contain video posted in Dordrecht, from where another perspective on the incident could be perceived, with the following facted query $q_{example}$, *(location, Dordrecht), (hasVideo, true), (type, Fire)*. Only those tweets that match all the facet-value constraints will be returned to the user. The ranking of the tweets that match a query is a research problem of its own and is, in the context of microblogging systems, usually solved by ranking according to recency [163]. Twitcident ranks the matching tweets according to their (i) creation time or (ii) relevance. The relevance is computed by exploiting various features including provenance information such as the authority score of the user who published a tweet [153].

A key challenge in engineering a faceted search interface is to support the facet-value selection as well as possible. Hence, the facet-value pairs that are presented in the faceted search interface (see Figure 6.1(a)) have to be ranked so that users can quickly narrow down the search result lists until they find the tweets that fulfill their information needs. The Twitcident framework provides different strategies that allow for ranking facet-value pairs and therefore generating query recommendations.

Frequency-based Faceted Search. A straightforward approach is to rank the facet-value pairs $(f, v) \in FVPs$ based on their occurrence frequency in the current hit list H of Twitter messages that match the current query $q = \{(f, v) | (f, v) \in FVPs \text{ selected as filter}\}$, i.e. messages that contain all facet-value pairs in q:

$$rank_{frequency}((f, v), H) = |H_{(f,v)}| \tag{6.3}$$

$|H_{(f,v)}|$ is the number of (remaining) messages that contain the facet-value pair (f, v) which can be applied to further filter the given hit list H. By ranking those facet values high that appear in most of the messages, $rank_{frequency}$ minimizes the risk of ranking relevant facet values too low.

However, it might increase the effort that a user has to invest to narrow down the search result list: by selecting facet values which occur in most of the remaining tweets the size of the hit list is reduced slowly.

Time-sensitive Faceted Search. Topics that are reported and discussed on the Social Web about an incident may change over time [91, 93]. Hence, also the information demands of users who are seeking details about an incident are likely to shift. The time-sensitive faceted search strategy adapts to this behavior and promotes those trending facet-value pairs that are often mentioned in recent Social Web messages:

$$rank_{time}((f,v),H) = max(\{age(m)|m \in H\}) - \frac{\sum_{m \in H_{(f,v)}} age(m)}{|\{m \in H_{(f,v)}\}|} \quad (6.4)$$

Here, $age(m)$ is the age of a message $m \in H$ (and $m \in H_{(f,v)}$) with respect to the current time when the query is issued. $rank_{time}((f,v),H)$ thus calculates the temporal distance between the oldest message in the hit list and the average age of messages that contain the given facet-value pair (f,v). The younger the average age of messages that mention (f,v), the higher the ranking score.

Personalized Faceted Search. Individual users may have different information needs that are reflected by their personal interests. To adapt the faceted search to the individual demands of a user, the Twitcident framework infers a user's interests from her Twitter activities, i.e. from the tweets a user published herself. The interest profile $P(u)$ of a user $u \in U$ can therefore be represented in the same way as incident or tweet profiles (cf. Definition 2), hence as a set of weighted facet-value pairs.

$$\begin{aligned} P(u) =& \{((f,v), w(u,(f,v)))| \\ & (f,v) \in \bigcup_{t \in T_u} P(t), u \in U, w(u,(f,v)) \in [0..1]\} \end{aligned} \quad (6.5)$$

Twitcident analyzes the entire Twitter timeline of a user to construct a profile. It thus considers all the profiles $P(t)$ of tweets that the user published and weighs the facet-value pairs according to their occurrence frequency in the tweets. Given a facet-value pair (f,v), the personalized facet ranking strategy utilizes the weight $w(u,(f,v))$ in $P(u)$ to determine the ranking

score:

$$rank_{pers}((f,v),P(u)) = \begin{cases} w(u,(f,v)) & \text{if } (f,v) \in P(u) \\ 0 & \text{otherwise} \end{cases} \quad (6.6)$$

The Twitcident framework moreover allows to combine different faceted search strategies using their normalized ranking score so that the following condition should be satisfied: $rank((f,v),H) \in [0..1]$. In our experiments in Section 6.5, we combine the personalized and time-sensitive ranking strategy with the frequency-based strategy and set $\lambda = 0.5$:

$$\begin{aligned} rank_{combine}((f,v),H) = \lambda rank_{frequency}((f,v),H) \\ + (1-\lambda) rank_{personalized}((f,v),H) \end{aligned} \quad (6.7)$$

Realtime Analytics

Based on the semantic enrichment, the Twitcident framework provides functionality to analyze the current Social Web stream about an incident. Figure 6.2 shows some of the graphical widgets that are delivered to the users such as the evolution of topics over time or the geographical impact area of an incident. Twitcident exploits the incident and tweet profiles to generate these diagrams. For example, the impact area of an incident is deduced from the geographical location of Twitter messages that contain experiences of users, e.g. in which people state what they see, hear, or smell. The analytical tools adapt furthermore to the current context of a user: if a user filters the Social Web stream by means of faceted search then the diagrams summarize and visualize only that fraction of the information that matches the filter.

Having introduced the core functionality of the Twitcident framework and its implementation with TAL, we will, in the next sections, evaluate the two fundamental research challenges that we approach with the Twitcident framework: automated filtering of relevant information from Social Web streams (see Section 6.4) and search within Social Web streams (see Section 6.5).

6.4 Evaluation of Tweet Filtering

Given that people publish around 500 million messages per day on Twitter as mentioned in Chapter 1, automatically retrieving and filtering information about particular incidents from Twitter streams is thus a non-trivial

problem. In this section, we evaluate and compare the different strategies that Twitcident provides in order to solve this challenge and investigate the following research questions:

- Which filtering strategy performs best in retrieving messages that are relevant for a given incident? How do semantic filtering strategies perform in comparison to keyword-based approaches?
- How are the filtering strategies affected by the characteristics of the (initial) incident description?

6.4.1 Experimental Setup

We evaluate the filtering strategies, again making use of the context of the TREC Microblog track that was introduced in 2011. The dataset description has been given in Table 3.2. In our experiments, we interpret the topics that come together with the corpus, e.g. *Mexico drug war* or *Protests in Jordan*, as incidents and consider the topic string as the initial incident description which the Twitcident framework exploits to perform incident profiling and tweet filtering (see Figure 6.1).

For the top tweets returned by each filtering strategy for each topic, we utilized the relevance judgments to evaluate the filtering performance. Thus, we measure the performance via mean average precision (MAP), precision within the top-k returned items (P@k) and recall.

Baseline: Keyword Filtering

We compare the semantic filtering strategies provided by the Twitcident framework with a keyword-based filtering baseline that interprets the label of a topic as a keyword query. The baseline evaluates a query and generates a ranking of tweets using language modeling with relevance model RM2 [186]. Apart from filtering out non-English tweets, the baseline also filters out retweets, tweets with less than 100 characters and tweets with words that contain a single letter three or more times in sequence (e.g., "ooooooooh").

6.4.2 Experimental Results

Figure 6.4 summarizes the results of our filtering evaluation and demonstrates that the semantic strategies of the Twitcident framework clearly outperform the keyword-based filtering in all metrics.

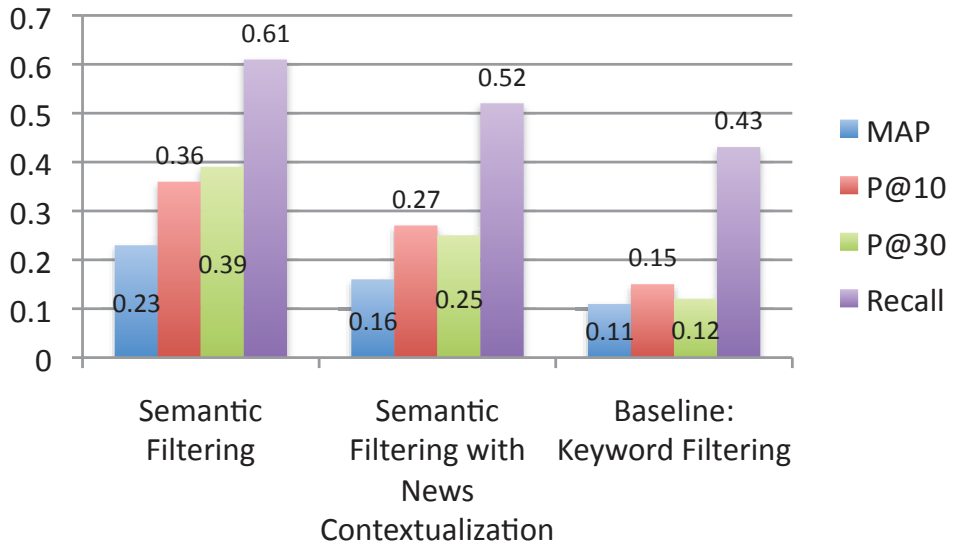
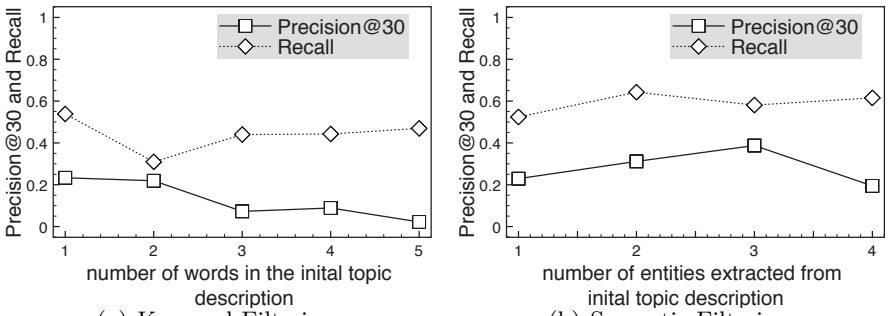


Figure 6.4: Result overview on the filtering strategies. Reported are the mean average precision (MAP), precision at k (P@10, P@30) and recall.

For example, the semantic filtering performs—with respect to MAP, P@10, and P@30—more than twice as good as the baseline and with respect to recall it improves the filtering performance by 41.8%. News-based contextualization also leads to major improvements in comparison to the keyword-based baseline. However, it performs worse than the semantic filtering which performs incident profiling solely based on tweets. This indicates that facet-value pairs that are extracted from news articles, which contain reports about the incident/topic, seem to add too much noise in the incident profiling and filtering process.

Figure 6.5 illustrates the impact of the initial topic description on the filtering. The x-axis specifies the number of (a) words and (b) facet-value pairs that are extracted from the initial description while the y-axis marks P@30 and recall. For keyword filtering, we observe that the precision almost gradually drops the more keywords are listed in the initial topic description so that for topics that feature six keywords, the average precision is just 0.03. In contrast, the semantic filtering, which does not consider all keywords from the topic description but considers only named entities for the topic profiling, is more robust and also achieves in the worst case a considerably higher average precision of 0.2. For both strategies, the recall increases slightly the more concepts are extracted from the initial topic description. Again, the semantic filtering performs better than the keyword-based filtering and features a more



(a) Keyword Filtering (b) Semantic Filtering

Figure 6.5: Robustness of (a) keyword-based filtering and (b) semantic filtering: correlation between the number of (a) words and (b) semantic concepts that can be extracted from the initial topic description and the filtering performance measured by means of Precision@30 and Recall.

stable behavior when characteristics of the topic description vary.

6.4.3 Synopsis

In conclusion, we can therefore answer the research questions raised at the beginning of this section as follows.

1. Semantic filtering allows for the best filtering performance. It clearly outperforms the keyword-based strategy and more than doubles mean average precision.

2. The complexity of a topic, measured by the number of concepts that can be extracted from the initial topic description, impacts the precision of the keyword-based strategy negatively: the higher the complexity the lower the precision. The semantic filtering strategy is more robust and also achieves high precisions for complex topics.

6.5 Evaluation of Faceted Search

Based on the automatic filtering of Social Web streams for detecting messages that are relevant for a given incident, the Twitcident framework provides faceted search functionality that allows users to filter the messages and retrieve information they are interested in. In line with the evaluations done

in [1], we now evaluate the quality of the faceted search strategies on top of the automatic filtering process and study the following research questions:

1. How well does faceted search supported by the Twitcident framework perform in comparison to keyword search?
2. What faceted search strategy supports users best in finding relevant Twitter messages?
3. What factors influence the performance of the faceted search strategies?

6.5.1 Experimental Setup

In order to answer the above research questions and evaluate the faceted search strategies (see Section 6.3.4), we applied an evaluation methodology introduced by Koren et al. [90] that simulates the clicking behavior of users in the context of faceted search interfaces. In a faceted search interface, a user can select a facet-value pair to refine the query and drill down the search result list until she finds a relevant document. We model the user's facet-value pair selection behavior by means of a *first-match user* that selects the first matching facet-value pair and continues to refine the query until no more appropriate facet-value pairs can be selected.

To evaluate the performance, we used again the TREC microblog dataset described in Section 3.3.3 and generated search settings by randomly selecting, for each of the 50 topics, 50 re-tweets which mention at least one hashtag—thus resulting in 2500 settings. Each search setting consists of (i) a target tweet (= the tweet that was re-tweeted), (ii) a user that is searching for the tweet (= the user who re-tweeted the tweet) and (iii) the timestamp of the search activity (= the time when the user re-tweeted the message). The set of candidate items is given by all those tweets which have been published before the search activity and are considered to be relevant to the corresponding topic based on the semantic filtering strategy of the Twitcident framework. We thus test—except for the incident detection—the entire pipeline of the Twitcident framework as depicted in Figure 6.1. The filtering delivered, on average, more than 5000 candidates per search setting while there is only exactly one Twitter message that is considered to be relevant, namely the Twitter message that was actually re-tweeted by the user.

For measuring the performance of the search strategies, we use mean reciprocal rank (MRR) of the target item in the search result ranking[9] when

[9]Tweets are ranked according to their creation time so that the latest tweets appear at

the user selects it. Furthermore, we utilize MRR of the first relevant facet-value pair and success at rank k (S@k) which is the probability that a relevant facet-value pair, that the user selects to narrow down the search result list, appears within the top-k of the facet-value pair ranking. Both metrics are direct indicators for the effort a user needs to spend using the search interface: the higher MRR and S@k, the faster the user will find a relevant facet-value pair when scanning the facet-value pair ranking.

Dataset Characteristics

In the faceted search evaluation, we moreover experiment with the link-based semantic enrichment that is provided by the Twitcident framework (see Section 6.3.3). As depicted in Figure 6.6, we observe that the extraction of facet-value pairs from Web resources that are linked from a Twitter message allows to further extend the profile of the corresponding tweet. It therefore reduces the level of sparsity. For example, for semantic enrichment, which is solely based on tweets, 41.2% of the messages feature at least two facet-value pairs while the additional link-based enrichment allows for representing 60.1% of the tweets with at least two facet-value pairs.

Baseline Strategies

We compare the faceted search strategies of the Twitcident framework (see Section 6.3.4) with two baseline strategies that exploit hashtags:

Hashtag-based Keyword Search For this baseline strategy, the user randomly selects one of the hashtags that is mentioned in the Twitter message the user is searching for[10]. Given the messages that match this keyword query, the user starts scanning the result list.

Hashtag-based Faceted Search This strategy interprets hashtags as facet values and therefore ranks the hashtag-based facet-value pairs in the same way as the frequency-based faceted search strategy (see Section 6.3.4), i.e. according to their occurrence frequency in the current search result list. The selection of hashtag-based facet-value pairs is simulated according to the aforementioned procedure.

---

the top of the ranking.

[10]To not discriminate the hashtag-based search strategies, we selected the search settings so that each target tweet contains at least one hashtag.

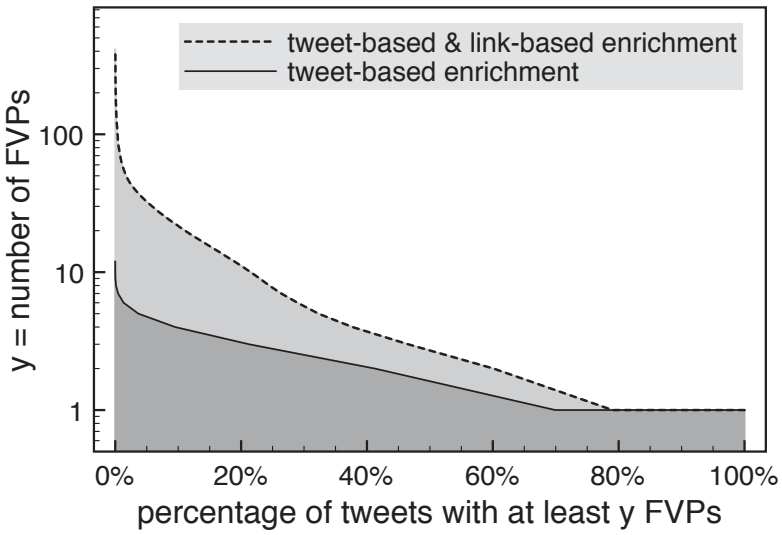


Figure 6.6: Impact of link-based semantic enrichment: the histogram shows the fraction of Twitter messages that feature at least y facet-value pairs (FVPs) for (i) semantic enrichment solely on tweets (tweet-based) and (ii) the link-based strategy that follows links which are posted in Twitter messages.

6.5.2 Experimental Results

Figure 6.7 compares the frequency-based faceted search strategy featured by the Twitcident framework with the hashtag-based search strategies. The comparison of the MRR scores reveals that the semantic faceted search strategy improves the search performance significantly by 34.8% and 22.4% over the hashtag-based keyword search and the hashtag-based faceted search strategy[11]. Interpreting hashtags as facet values leads to an improvement over the single keyword query as well. However, the semantic enrichment provided by the Twitcident framework proves to generate more valuable representations of the Twitter messages and therefore allows for faceted search functionality that clearly outperforms the two hashtag-based strategies.

The performance of the different faceted search strategies is listed in Figure 6.8. The performance of those strategies that benefit from the semantic enrichment significantly exceeds the performance of the hashtag-based strategy in predicting appropriate facet-value pairs. A detailed review of the results shows that a key success factor of the semantic faceted

[11]Statistical significance was tested with a two-tailed t-Test where the significance level was set to $\alpha = 0.01$.

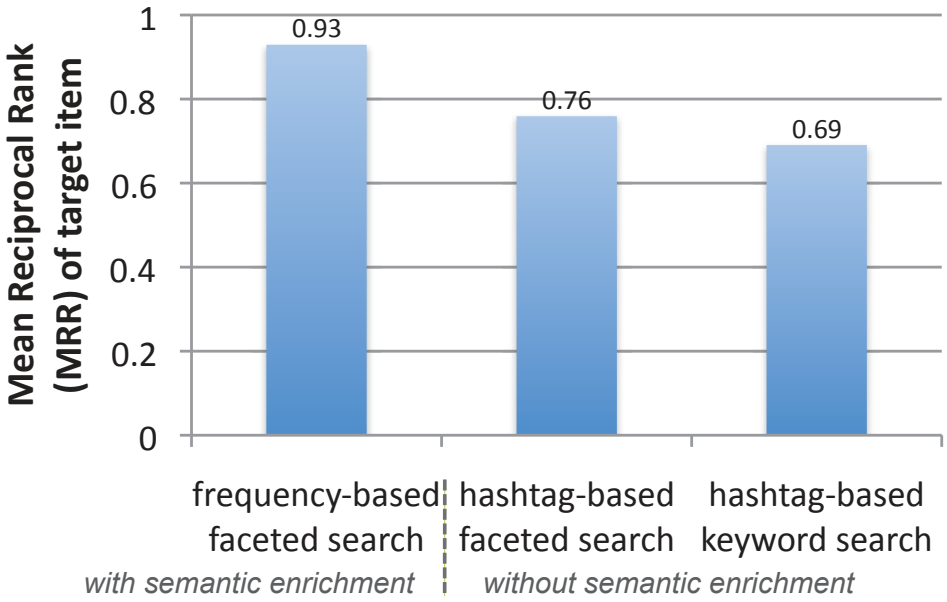


Figure 6.7: Result overview of search strategies: comparison of hashtag-based and semantic search.

search strategies is given by their ability of disambiguating facet-value pairs. While the hashtag-based strategy would, for example, treat *#Tahrir* and *#TahrirSquare* as different facet values, the semantic faceted search strategies would—in context of the *"Egyptian evacuation"* incident which is one of the TREC topics—map both values to the same concept (namely *dbpedia:Tahrir_Square*) and therefore facilitate the faceted search for the user.

Figure 6.8 furthermore shows that both personalization and temporal contextualization lead to significant improvements over the frequency-based strategy. In fact, regarding MRR the performance of the personalized and time-sensitive strategies is 39.7% and 36.8% better than the one of the faceted search strategy that ranks the facet-value pairs according to their occurrence frequency in the current search result set.

By enriching the tweet profiles with facet-value pairs extracted from external Web resources that are referenced from the Twitter messages (link-based semantic enrichment), one can further improve the performance of the semantic faceted search strategies (see Figure 6.9). The level of improvement depends on the characteristics of the tweet profiles. Those search settings where the target tweet contains exactly one facet-value pair benefit most from the link-based enrichment. For these settings, the performance increases by

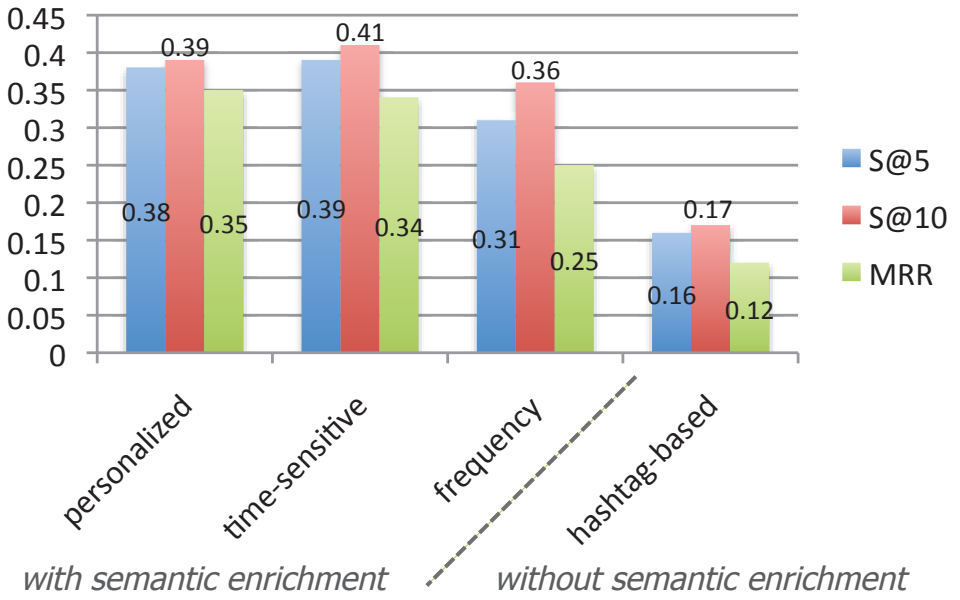


Figure 6.8: Result overview of the faceted search strategies. Reported are the mean reciprocal rank (MRR) of the first relevant facet-value pair (FVP) and success at k (S@5, S@10), i.e. the probability that a relevant FVP appears within the top k.

14.5% for the frequency-based strategy and around 7% for the personalized and time-sensitive strategies.

Figure 6.10 allows us to study how the performance of the personalized and time-sensitive search strategies depends on the characteristics of the user and incident profiles. Therefore, Figure 6.10(a) plots the MRR scores of the personalized strategy in relation to the size of the profile of the user who performed the corresponding search activity. It is interesting to observe how the average performance varies with changing profile sizes: the average MRR for profiles with less then 10 distinct FVPs is 0.328. The personalized strategy achieves its maximum average MRR performance for profiles that feature between 50 and 70 FVPs while for the few user profiles which feature more than 150 FVPs the performance drops—possibly because those profiles feature too much diversity.

The time-sensitive faceted search strategy, which promotes those facet-value pairs that are currently trending, performs best for those search settings that are performed within a topic that is characterized by strong temporal dynamics (see Figure 6.10(b)). Here, the dynamics of a topic are described by means of the standard deviation of the creation times of tweets which

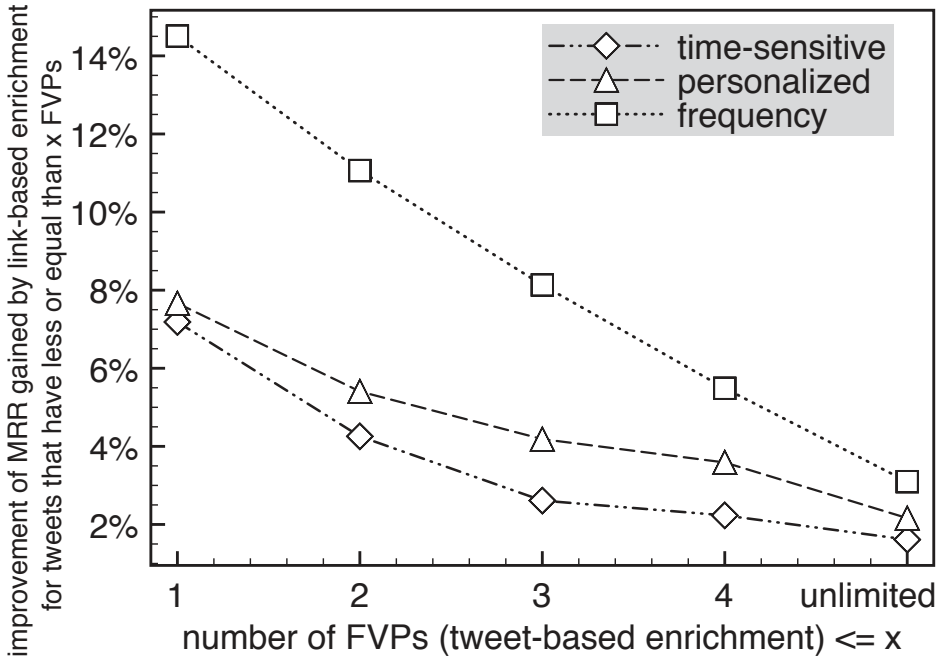


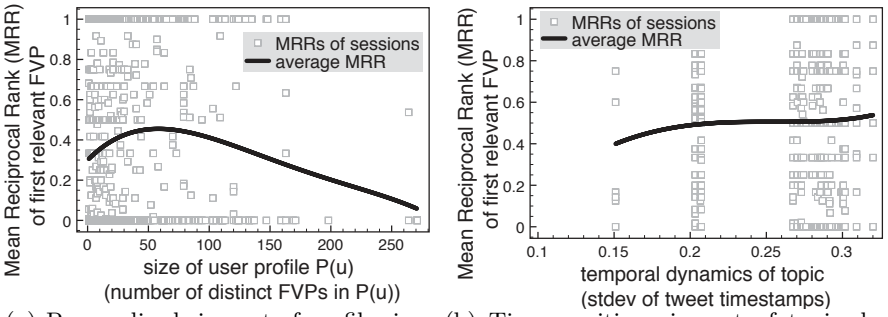
Figure 6.9: Impact of link-based semantic enrichment on faceted search performance. The y-axis shows the improvement with respect to the mean reciprocal rank (MRR) of the first relevant FVP that is gained when using link-based enrichment in addition to solely tweet-based enrichment averaged for those search settings where the target tweet features x or less than x FVPs.

are considered to be relevant for the topic. Figure 6.10(b) depicts that the performance slightly increases the more a topic underlies temporal changes. Hence, the more distributed the messages are posted over time the more important it is to adapt to the temporal context.

6.5.3 Synopsis

Given the experimental results, we can answer the research questions raised at the beginning of this section:

1. Faceted search strategies allow for significantly higher search performance than the hashtag-based keyword search strategies. They enable users to more precisely filter tweets and therefore retrieve relevant information.



(a) Personalized: impact of profile size (b) Time-sensitive: impact of topic dynamics

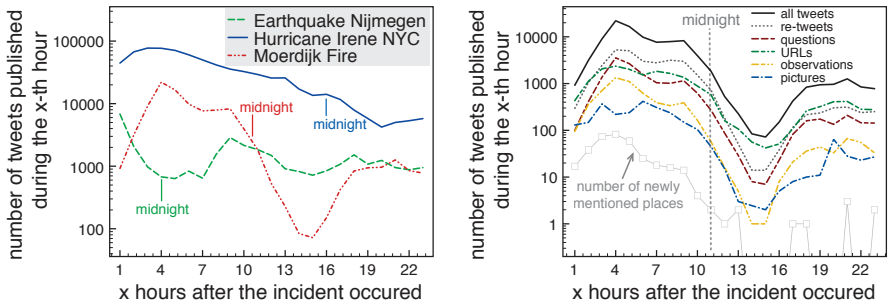
Figure 6.10: Impact of (a) profile size on the search performance of the personalized faceted search strategy and correlation between (b) search performance and temporal dynamics of the topic within which a user is searching. Temporal dynamics is measured by means of the standard deviation of the timestamps of Twitter messages that are published within one topic, i.e. a high standard deviation indicates strong temporal dynamics.

2. Personalized and time-sensitive faceted search strategies that adapt to the profile of a user and to the temporal context respectively allow for the best search performance and lead to significant improvements over the standard semantic faceted search strategy. Further exploitation of links posted in tweets allows us to further enrich the semantic representation of tweets and moreover induces additional improvements of the search performance.

3. The performance of the personalized faceted search strategy is influenced by the size of a user's profile and achieves the highest performance for medium-sized profiles. The quality of the time-sensitive strategy depends on the temporal dynamics within an incident: the more temporal changes the more important it is to adapt to the temporal context.

6.6 Discussion

With Twitcident we introduce a system that allows users to explore, search and analyze information about incidents available on the Social Web and Twitter in particular. Since January 2011, we have tested the Twitcident system in practice to monitor various incidents, specifically to support emergency services such as the Dutch police and fire department. Given these ex-



(a) Different incident types (b) Moerdijk fire

Figure 6.11: Posting behavior about incidents within the first 24 hours of an incident: (a) comparison of different types of incidents and (b) type of information posted during a fire incident in Moerdijk.

periences, we identify that different types of incidents imply different types of posting behavior on the Social Web. For example, Figure 6.11(a) compares the number of Twitter messages posted about three different types of incidents within the first 24 hours: a large-scale fire at a chemical factory in Moerdijk (Jan 5th 2011), an earthquake with its epicenter close to Nijmegen (Sep 8th 2011) and the so-called hurricane *Irene* which caused flooding in New York (Aug 28th 2011). One can see that all incidents reach their maximum peak within the first 4 hours after the incident occurred. For the fire and hurricane the amount of tweets gradually grows until it reaches its peak while for the unexpected earthquake most tweets are already published within the first hour after the incident. In fact, the hurricane Irene did not hit New York City unexpectedly, but was forecast already weeks ahead which caused Twitter traffic already before the hurricane appeared.

Twitcident thus has to process huge amounts of messages within the first hours of an incident. To handle tens of thousand of messages per hour, Twitcident parallelizes the semantic enrichment of Twitter messages which is the most time-intensive procedure. In particular, following URLs and processing the corresponding Web sites may take seconds. Therefore, Twitcident applies heuristics to decide whether the link of a tweet should be processed in realtime or marked for later processing (e.g. during the night when the amount of messages to be processed decreases; see Figure 6.11(a)). For example, URLs posted in tweets for which the tweet-based enrichment—which takes on average between 100 and 300 milliseconds—detects already two or more facet-value pairs are not processed immediately because for these tweets the link-based enrichment improves the search performance only slightly (see Figure 6.9).

Figure 6.11(b) illustrates for the fire at the chemical factory in Moerdijk the kind of information that is posted on Twitter within the first 24 hours after the fire started. It is interesting to see that the number of questions that are being asked is exceptionally high when the overall number of tweets reaches its maximum. At that point, questions such as *"What about the toxic cloud?"* or *"Is there a chance that the smoke is going to Leiden?"* are prominent and exceed the amount of URLs and pictures which may reveal answers to these questions. Emergency services are often interested in *new* information and question, for example, whether the impact area of an incident is increasing (cf. "number of newly mentioned places" in Figure 6.11(b)).

Twitcident allows people to find answers to such questions and allows emergency services to analyze the information that people publish on the Social Web.

6.7 Conclusions

In this chapter we introduced Twitcident, a framework for filtering, searching and analyzing information about incidents that people publish in their Social Web streams. Triggered by an incident detection module that monitors emergency broadcasting services, our framework automatically collects and filters relevant information from Twitter. It enriches the semantics of Twitter messages to adapt and improve the incident profiling and filtering over time. Semantic enrichment is also the foundation for faceted search and realtime analytics provided by the Twitcident framework. In our evaluations we proved that semantic enrichment boosts the performance of both the filtering of Twitter messages for a given incident and the search for relevant information about an incident within the filtered messages significantly.

Chapter 7

Conclusion

With its mechanism of motivating end-users to become active producers of content, the Social Web attracts a large community to contribute an immense amount of user-generated content about influential events and daily activities. Thus, it has become a rich resource for exploiting social insights as well as an emerging target to conduct interdisciplinary research. Performing analytics on Social Web data can be used to detect natural disasters, monitor the seasonal flu, derive personal favors, or fulfill a variety of other information needs in different scenarios. Given the characteristics of Social Web data, researchers and developers working to achieve these goals have encountered new and interesting challenges while conducting various analytical tasks. In this thesis, taking the microblogging service Twitter as a typical application of the Social Web, we proposed a novel platform to support various tasks of data analytics, gained deeper understanding of Twitter data from the perspective of three aspects: relevance, redundancy, and diversity. We further showed the applicability and value of our works by presenting Twinder, a search engine for Twitter streams, and Twitcident, an information exploration system for Twitter messages posted during emergency circumstances. This chapter concludes the work done in this thesis and sketches the possible future directions.

7.1 Summary of Contributions

In this section, we summarize the answers to the research questions that have been raised in the first chapter and sum up the contributions made by this thesis.

- **Social Web Data Analytical Platform.** In business intelligence cases [35], the data analytics tasks are often conducted on structured data stored in relational databases. The consequent data analytics results provide their consumers a data-centric approach to gain insights in terms of intelligence and "knowledge-to-act" [34, 168, 178]. The emerging development of Social Web applications opens new possibilities for data analytics in a wider range of application scenarios. The most typical platform being used for Social Web data analytics is Twitter because of its pervasiveness and data availability. However, there does not exist a systematic solution which allows for performing data analytics tasks with Twitter data effectively and efficiently for researchers and developers. Therefore, the following research questions had to be answered.

  - What are the characteristics of Social Web data, which make analytics a non-trivial challenge?
  - What are the common core procedures across Social Web data analytics?
  - How can we accommodate essential procedures for Social Web data analytics in a scalable platform?
  - How can we efficiently build workflows for Social Web data analytics?

  Based on studying use cases of Social Web data analytics, we provided our answer to the first two questions. This answer gave us the inspiration to design a unified solution to problems of a more general scope, as the answer to the third research question. Therefore, we proposed TAP (Twitter Analytical Platform), with which we can conduct Twitter data analytics tasks. For the last question, we proposed TAL (Twitter Analysis Language), a domain-specific language which allows for customizing the workflows that can be executed on TAP. We showcased the applicability of our platform by building a prototype search engine for Twitter with very little coding efforts.

- **Relevance Estimation for Microblog Search.** Searching for tweets that are relevant to a given topic is a non-trivial research challenge due to the high-volume of posts published every day, especially during trending events. Previous studies [163] showed that users exhibit a different search behaviour on Twitter compared to Web search. To enhance microblog search and move beyond keyword-based retrieval strategies, we thus answered the following research questions.

- How can we enrich search queries on Twitter with background knowledge in order to better understand the meaning behind them?
- Which micropost features allow us to best predict a micropost's relevance to a query?
- How can we put our analytical findings into our prototype Twinder so that the overall retrieval effectiveness of the system improves?

We answered these questions in Chapter 3 and provided a solution to identify the relevant Twitter messages for a given topic. We tried to expand the original queries with background knowledge from the Linked Open Data Cloud and external Web resources. Taking the positive results from query expansion as one feature of a tweet-query pair, we further extracted other features based on the hypotheses made for predicting the relevance relationship between tweets and queries. Besides query-sensitive features, we also focused on query-insensitive features of tweets which may be extracted before the query is given. In this way, we introduced syntactic, semantic, as well as contextual features and analyzed them with a standard microblog corpus. We evaluated the utility of these features with a machine learning approach that allowed us to gain insights into the importance of the different features for the relevance classification.

Our main discoveries about the factors that lead to relevant tweets are the following: (i) The learned models which take advantage of semantic and query-sensitive features outperform those which do not take the semantic and query-sensitive features into account. (ii) The social context of the user posting the message has little impact on the prediction. (iii) The importance of a feature differs depending on the characteristics of the topics. For example, the sentiment-based feature is more important for popular than for unpopular topics and the semantic similarity does not have a significant impact on entertainment topics.

We applied these results in practice by integrating them into the Twinder search engine.

- **Near-Duplicate Detection for Microblog Search.** The majority of tweets published every day are related to either trending or persistent news [91]. Besides retweets, users post short messages of similar contents, which are often not easily detected by simple automatic methods. The existing approaches of near-duplicate detection were designed for the Web. To understand the significance of the content redundancy

problem and solve the near-duplicate detection problem on Twitter, we answered the following research questions.

- How much duplicate content exists in typical microblog search results?
- How can we automatically detect the duplicate content along with the duplication level?
- How does removing or aggregating duplicate contents affect the quality of the search results with respect to diversity?

In Chapter 4, we answered the first research question by analyzing the duplicate contents from which we derived a 5-level model of near-duplicates for Twitter messages. We further developed a framework for identifying near-duplicate tweets by combining (i) syntactic features, (ii) semantic features and (iii) contextual features and by considering information from external Web resources that are linked from the microposts. We found that a higher performance had been achieved for topics of certain themes, popularity, locality, or temporal persistence. Our experiments showed that semantic features such as the overlap of WordNet concepts are of particular importance for detecting near-duplicates. We showed the effectiveness of classifying the duplicates into different levels according to our proposed model. Given our near-duplicate detection strategies, we additionally developed functionality to reduce the redundancy of search results. We integrated this functionality into the Twinder search engine and showed that our duplicate detection framework had improved the quality of top-k retrieval significantly since we had decreased the fraction of duplicate content that was delivered to the users.

- **Diversity Analytics of Microblog Search Results.** Previous studies [15, 40, 139] have been devoted to evaluation measures and approaches for search results diversification on the Web. The characteristics of microblog posts, e.g. shortness, make it hard to distinguish the subtopics among the messages on a general topic. To obtain insights of the diversity within the microblog search results, we had to conduct investigations on a dedicated corpus, which did not yet exist. Therefore, the following research questions needed to be answered.

  - How can we build a microblog corpus for search result diversification?
  - How suitable is the corpus that we created for research on search result diversification?

  - To what extent can we achieve diversity by applying the developed de-duplication strategies?

  Chapter 5 provided answers to these questions and presented our efforts in building a microblog corpus for search results diversification. We then showed its suitability for this purpose by conducting a comprehensive analysis of the corpus. The considerable subjectivity in the annotation process had been revealed in the analysis of the annotators' influence on subtopic creation and relevance judgments. However, we showed that the observed trends with respect to the different evaluation measures were largely independent of the specific annotator by carrying out de-duplication retrieval experiments. We also found that the importance of features varied according to the topic type by comparing the results as reported in Chapter 4.

- **Information Exploration System for Social Web Streams.** Existing studies show that the data from the Social Web, particularly Twitter, helps to detect incidents [133] by analyzing the information that people report about an incident [60]. The fundamental engineering challenges for building such systems, including automatically filtering relevant information from the Social Web and making the information accessible and findable in a given incident context had not been researched sufficiently yet. Hence, we raised the following research questions.

  - How can we build an information exploration system with the Twitter Analytical Platform?
  - How well do the proposed strategies for information exploration perform in fulfilling the information needs?

  In Chapter 6, we answered these research questions by introducing Twitcident, a framework for filtering, searching and, analyzing information about incidents that people publish in their Social Web streams. Twitcident makes use of the analysis tools provided by the Twitter Analytical Platform to collect Twitter streams about the given incident, which is then filtered, enriched, and prepared for rendering into analytics in the Web applications, including faceted search and visualized analytics. The incident profile can be refined as further information from the Twitter streams is continuously integrated. We evaluated the Twitcident framework and showed that semantic enrichment significantly improved the performance of both the filtering of Twitter

messages for a given incident and the search for relevant information about an incident within the filtered messages.

7.2 Future Work

As a Chinese proverb said, it is as impossible to find a perfect man as it is to find 100 percent pure gold. This thesis is no exception for having flaws and limitations. For example, the relevance model of RM2 [92] that we used in Chapter 3 was adopted for evaluating the impact of semantics on the effectiveness of our query extension strategies. However, we noticed that, with RM3 [77], a baseline run could compete with our query expansion strategy. Recently, numerous dataflow languages and tools have become popular, such as Pig [122], Scalding[1], Spark[185], as well as Storm [166] and Summingbird [22] which provide support for processing streaming data in real-time. Due to the lack of engineering efforts, the tool kits designed for experimental purposes in our Twitter Analytical Platform may not be competitive as aforementioned popular languages in terms of stability, throughput, maintainability, etc. There is still space for improvements in the diversification of microblog search results. We list the challenging work that can further complete this thesis as follows.

Twitter Analytical Platform. Given the data analytical platform introduced in Chapter 2 for Twitter with the tools in the functionality stack, the lists of supported Social Web applications and analysis tools could be extended continuously. The plan of supporting more Social Web applications leads to the possibility of *cross-platform Social Web data analytics* and the requirements of including media contents of more types, including images, videos, or even interactive content. The demands in tooling can be derived from the analytics practices with the current platform and surveys on other analytical use cases. The domain-specific language, Twitter Analysis Language, has the potential [83] to support a more complex syntax, e.g. conditional and loop statement, to make the workflow programming more expressive and efficient. As the Twitter Analysis Language and the implementation of the functionality stack are loosely coupled, we can make use of the popular dataflow processing languages and tools, e.g. Spark or Summingbird, to realize the analytical tools. In that case, our platform can benefit from the performances already achieved by these technologies while attracting more

[1] https://dev.twitter.com/blog/scalding, accessed October 14th, 2014.

scientists and developers to engage in this project. Currently, the analytical platform considers a single post from the Social Web as the unit for analytics. However, it would be a non-trivial challenge to accommodate the tasks of network analyses.

Social Web Search. As the three chapters (Chapters 3-5) in this thesis focus on the analytics in the context of information retrieval for Twitter, we would recommend two promising research directions, including *personalized retrieval* and *advanced search result diversification* with novel and specialised measures for Twitter. Since Abel et al. [3, 4, 155] have shown that the user preferences can be derived from Twitter activities, it is feasible to make the search results adaptive to personal preferences. Moreover, we have identified the importance of topic type for analytics in different aspects, hence the corresponding adaptation algorithms are expected to be beneficial for retrieval effectiveness. This also implies that we need to better understand the user intents behind the queries. Hence, further investigation of diversity among microblog search results could help. The characteristics of Twitter messages can make the problem quite challenging. Therefore, we suggest to consider temporal information to tackle this challenge as the discussion on Twitter shows strong features in timeliness. Furthermore, the meaning of diversity in the Social Web context could be redefined with the consideration of its characteristics. For instance, the factors of media types, temporal features, sentiments, or even geo-locations could be considered while forging a diversified search result that will be delivered to end-users.

Applications. As in Chapter 6 the efforts have been spent on the fundamental challenges of automatically filtering to the context of a given incident and real-time analytics, we have noticed the potential value in the research of *early signal detection* and *comprehensive real-time visualization support* for Social Web data. The existing works show that the disease reporting events and severity levels can be detected and tracked from the Web content [143, 152]. Hence, it would be exciting to study the methodology for detecting events of certain types, such as possible unplanned mass events, traffic accidents that will potentially have impact on transportation systems, and extreme natural hazards. In the Twitcident system as presented in this thesis, we have provided numerous visualization tools for end-users to consume the results of analytics from the Twitter data. It would be challenging and valuable to explore the problems that one would have to solve while designing a workbench for crafting the visualization products with user-generated unstructured data on the Web. While there have been research efforts focusing on interactive data visualization environments [141] and the strategies to tackle the chal-

lenges introduced by Big Data characteristics [102], i.e. Volume, Velocity, Variety etc., we would recommend to investigate the methodologies for engineering visualization products with the Social Web data so that the outcomes could be integrated into our platform to make the analytics designing more efficient, e.g. providing programmable or interactive interface for the visualization design.

Besides the efforts that can be spent to make this thesis perfect, the existing contributions in this thesis for Social Web data analytics create create a number of thought-provoking research directions that are worth to be investigated in the future. To conclude this thesis, we sketch a couple of them as follows.

Temporal Summerizations. The overwhelming volume of messages generated on Social Web during popular events leads to information overload problem. To tackle it, one solution is to make a temporal summarization [9]. The contributions made in Chapter 4 and Chapter 5 studied redundancy and diversity within Twitter search results. These results are related to redundancy detection and topic discovery, which are two main issues that one needs to tackle for generating a chronological summerization of microposts retrieved for a given query. We also noticed the recent emerging efforts towards the similar goals. For example, in 2013 TREC introduced "Temporal Summarization Track" [12] which aims at developing systems that can make broadcast concise updates in short sentences about an on-going event and track the important event-related attributes, e.g. the number of casualties. Specifically for microposts, there is also a new "Tweet Timeline Generation" task introduced by TREC 2014 Microblog Track[2].

Daily Activity Analytics. Recently, there are numerous activity trackers introduced by either start-ups or big companies, such as FitBit[3], Jawbone[4], Nike+ FuelBand[5] and Apple Watch[6]. Furthermore, the corresponding platforms for sharing such data are released to provide various and increasingly rich analytics widgets. This leads to the feasibility of gaining deeper understanding of the pattern of users' daily activities [65]. However, the privacy issues have already been noticed [86] so that a trade-off between privacy

[2]https://github.com/lintool/twitter-tools/wiki/TREC-2014-Track-Guidelines#4-tweet-timeline-generation-task, accessed October 14th, 2014.

[3]http://www.fitbit.com/, accessed October 13th, 2014.

[4]https://jawbone.com/, accessed October 13th, 2014.

[5]http://www.nike.com/us/en_us/c/nikeplus-fuelband, accessed October 13th, 2014.

[6]https://www.apple.com/watch/, accessed October 13th, 2014.

preservation and benefits of the analytical results need to be found. In this thesis, we investigate the methodologies to analyze the traces that users left on Social Web applications mainly in the form of textual messages. Nevertheless, the platform can be extended to conduct analytics of people's activity traces. As a result, the application supported by our analytical platform can benefit from the more comprehensive user profiles or stereotypes built for people from different culture, industries, ages, or areas.

Making Sense of Contexts. The contextual information has already been utilized in the Web search to improve query suggestions [132]. Moreover, as mobile devices become increasingly popular, it is possible to proactively predict users' information needs based on the contextual information and their historical activities [7, 47, 183]. The analytical results from Chapter 3 and Chapter 4 showed that the contextual features had limitations in boosting the effectiveness of relevant estimation and duplicate detection. However, this can be caused by the simplicity of those features. Our Twitter Analytical Platform provides us with the possibility of profiling users [155] and the ability of referencing the external knowledge base. We speculate that making good use of external evidences linked from the contextual information may lead to better results.

Towards Knowledge Generation. Besides proactively making contextual suggestions with users' historical information, there are also emerging research efforts spent on assimilating wisdom from the crowd into knowledge bases [13]. There is a dedicated track, which is known as *Knowledge Base Acceleration* [59], in TREC since 2012. Our analytical platform can help in filtering and identifying the vital and timely information from the Social Web applications. This can help in various aspects for knowledge generation. For example, the filtered vital information can help users with the authoring process on platforms such as Wikipedia [84]. The more comprehensive results can prevent users from being bounded in the biases and may enhance the diversity and the objectivity. Apart from providing valuable information to help content authoring, the process of knowledge base acceleration can also benefit from inquiring the experts on the Social Web. Specifically, matching the questions or the missing knowledge with the users that know the answers is a non-trivial challenge [181, 182].

Bibliography

[1] F. Abel, I. Celik, and P. Siehndel. Leveraging the Semantics of Tweets for Adaptive Faceted Search on Twitter. In L. Aroyo, N. Noy, and C. Welty, editors, *Proceedings of the 10th International Semantic Web Conference (ISWC), Bonn, Germany*. Springer, October 2011.

[2] F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Analyzing Temporal Dynamics in Twitter Profiles for Personalized Recommendations in the Social Web. In *Proceedings of ACM WebSci '11, 3rd International Conference on Web Science, Koblenz, Germany*. ACM, June 2011.

[3] F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Analyzing User Modeling on Twitter for Personalized News Recommendations. In J. A. Konstan, R. Conejo, J. L. Marzo, and N. Oliver, editors, *Proceedings of the 19th International Conference on User Modeling, Adaption and Personalization (UMAP), Girona, Spain*, volume 6787 of *Lecture Notes in Computer Science*, pages 1–12, Girona, Spain, July 2011. Springer.

[4] F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Semantic Enrichment of Twitter Posts for User Profile Construction on the Social Web. In *Proceeding of 8th Extended Semantic Web Conference (ESWC '11)*, Heraklion, Greece, May 2011.

[5] F. Abel, C. Hauff, G.-J. Houben, R. Stronkman, and K. Tao. Semantics + Filtering + Search = Twitcident. Exploring Information in Social Web Streams. In *Proceedings of the 23rd ACM Conference on Hypertext and Social Media*, HT '12, pages 285–294, New York, NY, USA, 2012. ACM.

[6] F. Abel, C. Hauff, G.-J. Houben, R. Stronkman, and K. Tao. Twitcident: Fighting Fire with Information from Social Web Stream. In *International Conference on World Wide Web (WWW), Lyon, France*. ACM, 2012.

[7] F. Abel, C. Hauff, G.-J. Houben, and K. Tao. Leveraging User Modeling on the Social Web with Linked Data. In M. Brambilla, T. Tokuda, and R. Tolksdorf, editors, *Proceedings of the 12th International Conference on Web Engineering (ICWE '12)*, volume 7387 of *Lecture Notes in Computer Science*, pages 378–385. Springer, 2012.

[8] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying Search Results. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining (WSDM)*, pages 5–14. ACM, February 2009.

[9] J. Allan, R. Gupta, and V. Khandelwal. Temporal Summaries of New Topics. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 10–18, New York, NY, USA, 2001. ACM.

[10] D. Antoniades, I. Polakis, G. Kontaxis, E. Athanasopoulos, S. Ioannidis, E. P. Markatos, and T. Karagiannis. we.b: The web of Short URLs. In *Proceedings of the 20th International Conference on World Wide Web (WWW)*, pages 715–724. ACM, April 2011.

[11] Y. Arase, X. Xie, T. Hara, and S. Nishio. Mining People's Trips from Large Scale Geo-tagged Photos. In *Proceedings of the International Conference on Multimedia*, MM '10, pages 133–142, New York, NY, USA, 2010. ACM.

[12] A. Aslam, F. Diaz, M. Ekstrand-Abueg, V. Pavlu, and T. Sakai. TREC 2013 Temporal Summarization. *Proceeddings of the 22nd Text REtrieval Conference (TREC 2013)*, 2013.

[13] K. Balog, H. Ramampiaro, and K. Nørvåg. KBAAA: A Web-based Toolkit for the Assessment and Analysis of Knowledge Base Acceleration Systems. In *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval*, OAIR '13, pages 215–216, Paris, France, France, 2013. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

[14] A. Bandyopadhyay, K. Ghosh, P. Majumder, and M. Mitra. Query Expansion for Microblog Retrieval. *International Journal of Web Science*, 1(4):368–380, 2012.

[15] P. N. Bennett, B. Carterette, O. Chapelle, and T. Joachims. Beyond Binary Relevance: Preferences, Diversity, and Set-level Judgments. *SIGIR Forum*, 42(2):53–58, 2008.

[16] T. Berners-Lee, R. Cailliau, J.-F. Groff, and B. Pollermann. World-Wide Web: the Information Universe. *Internet Research*, 2(1):52–58, 1992.

[17] M. S. Bernstein, B. Suh, L. Hong, J. Chen, S. Kairam, and E. H. Chi. Eddi: Interactive Topic-based Browsing of Social Status Streams. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, UIST '10, pages 303–312, New York, NY, USA, 2010. ACM.

[18] Y. Bernstein and J. Zobel. Redundant Documents and Search Effectiveness. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pages 736–743, New York, NY, USA, 2005. ACM.

[19] C. Bizer, T. Heath, K. Idehen, and T. Berners-Lee. Linked Data on the Web (LDOW2008). In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 1265–1266, New York, NY, USA, 2008. ACM.

[20] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. DBpedia - A Crystallization Point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, July 2009.

[21] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, Mar. 2003.

[22] P. O. Boykin, S. Ritchie, I. O'Connell, and J. Lin. Summingbird: A Framework for Integrating Batch and Online MapReduce Computations. *PVLDB*, 7(13):1441–1451, 2014.

[23] M. Bravenboer, K. T. Kalleberg, R. Vermaas, and E. Visser. Stratego/XT 0.17. A language and toolset for program transformation. *Science of Computer Programming*, 72(1–2):52 – 70, 2008. Special Issue on Second issue of experimental software and toolkits (EST).

[24] S. Brin and L. Page. The Anatomy of a Large-scale Hypertextual Web Search Engine. *Computer networks and ISDN systems*, 30(1):107–117, 1998.

[25] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig. Syntactic Clustering of the Web. In *Selected Papers from the Sixth International Conference on World Wide Web*, pages 1157–1166, Essex, UK, 1997. Elsevier Science Publishers Ltd.

[26] M. Busch, K. Gade, B. Larson, P. Lok, S. Luckenbill, and J. Lin. Earlybird: Real-Time Search at Twitter. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, ICDE '12, pages 1360–1369, Washington, DC, USA, 2012. IEEE Computer Society.

[27] A. E. Cano, M. Rowe, M. Stankovic, and A. Dadzie, editors. *Proceedings of the Concept Extraction Challenge at the Workshop on 'Making Sense of Microposts', Rio de Janeiro, Brazil, May 13, 2013*, volume 1019 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2013.

[28] J. Carbonell and J. Goldstein. The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries. In *SIGIR '98*, pages 335–336, 1998.

[29] C. Carpineto, S. Osiński, G. Romano, and D. Weiss. A Survey of Web Clustering Engines. *ACM Comput. Surv.*, 41(3):17:1–17:38, July 2009.

[30] B. Carterette and P. Chandar. Probabilistic Models of Ranking Novel Documents for Faceted Topic Retrieval. In *CIKM '09*, pages 1287–1296, 2009.

[31] S. Chakrabarti, B. Dom, and M. van den Berg. System and Method for Focussed Web Crawling, July 9 2002. US Patent 6,418,433.

[32] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan. Expected Reciprocal Rank for Graded Relevance. In *CIKM '09*, pages 621–630, 2009.

[33] M. S. Charikar. Similarity Estimation Techniques from Rounding Algorithms. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC)*, pages 380–388. ACM, May 2002.

[34] S. Chaudhuri, U. Dayal, and V. Narasayya. An Overview of Business Intelligence Technology. *Commun. ACM*, 54(8):88–98, Aug. 2011.

[35] H. Chen, R. H. Chiang, and V. C. Storey. Business Intelligence and Analytics: From Big Data to Big Impact. *MIS quarterly*, 36(4):1165–1188, 2012.

[36] J. Chen, R. Nairn, and E. H. Chi. Speak Little and Well: Recommending Conversations in Online Social Streams. In *Proceedings of the 29th International Conference on Human Factors in Computing Systems (CHI)*, New York, NY, USA, 2011. ACM.

[37] E. Cho, S. A. Myers, and J. Leskovec. Friendship and Mobility: User Movement in Location-based Social Networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 1082–1090, New York, NY, USA, 2011. ACM.

[38] J. Cho, H. Garcia-Molina, and L. Page. Efficient Crawling Through URL Ordering. In *Proceedings of the Seventh International Conference on World Wide Web 7*, WWW7, pages 161–172, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V.

[39] Y. Choi, Y. Jung, and S.-H. Myaeng. Identifying Controversial Issues and Their Sub-topics in News Articles. In *Proceedings of the 2010 Pacific Asia Conference on Intelligence and Security Informatics*, PAISI'10, pages 140–153, Berlin, Heidelberg, 2010. Springer-Verlag.

[40] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and Diversity in Information Retrieval Evaluation. In *SIGIR '08*, pages 659–666, 2008.

[41] C. L. A. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 Web Track. In *TREC '09*, 2009.

[42] C. L. A. Clarke, M. Kolla, and O. Vechtomova. An Effectiveness Measure for Ambiguous and Underspecified Queries. In L. Azzopardi, G. Kazai, S. E. Robertson, S. M. Rüger, M. Shokouhi, D. Song, and E. Yilmaz, editors, *Proceedings of the 2009 Conference on the Theory of Information Retrieval*, ICTIR '09, pages 188–199, 2009.

[43] J. W. Cooper, A. R. Coden, and E. W. Brown. Detecting Similar Documents Using Salient Terms. In *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, CIKM '02, pages 245–251, New York, NY, USA, 2002. ACM.

[44] S. Cronen-Townsend and W. B. Croft. Quantifying Query Ambiguity. In *HLT '02*, pages 104–109, 2002.

[45] J. Daiber, M. Jakob, C. Hokamp, and P. N. Mendes. Improving Efficiency and Accuracy in Multilingual Entity Extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*, 2013.

[46] J. Dean and M. R. Henzinger. Finding Related Pages in the World Wide Web. In *Proceedings of the Eighth International Conference on*

World Wide Web, WWW '99, pages 1467–1479, New York, NY, USA, 1999. Elsevier North-Holland, Inc.

[47] A. Dean-Hall, C. L. Clarke, J. Kamps, P. Thomas, N. Simone, and E. Voorhees. Overview of the TREC 2013 Contextual Suggestion Track. *Proceeddings of the 22nd Text REtrieval Conference (TREC 2013)*, 2013.

[48] A. Dong, R. Zhang, P. Kolari, J. Bai, F. Diaz, Y. Chang, Z. Zheng, and H. Zha. Time is of the Essence: Improving Recency Ranking using Twitter Data. In *WWW '10: Proceedings of the 19th International Conference on World Wide Web*, pages 331–340, New York, NY, USA, 2010. ACM.

[49] W. Dong, Z. Wang, M. Charikar, and K. Li. High-confidence Near-duplicate Image Detection. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, ICMR '12, pages 1:1–1:8, New York, NY, USA, 2012. ACM.

[50] Y. Duan, L. Jiang, T. Qin, M. Zhou, and H.-Y. Shum. An Empirical Study on Learning to Rank of Tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 295–303, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[51] P. S. Earle, D. C. Bowden, and M. Guy. Twitter Earthquake Detection: Earthquake Monitoring in a Social World. *Annals of Geophysics*, 54(6), 2012.

[52] M. Efron. Information Search and Retrieval in Microblogs. *Journal of the American Society for Information Science and Technology*, 62(6):996–1008, June 2011.

[53] M. Efron and G. Golovchinsky. Estimation Methods for Ranking Recent Information. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 495–504, New York, NY, USA, 2011. ACM.

[54] M. Efron, J. Lin, J. He, and A. de Vries. Temporal Feedback for Tweet Search with Non-parametric Density Estimation. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, pages 33–42, New York, NY, USA, 2014. ACM.

[55] M. V. Erp, G. Rizzo, and R. Troncy. Learning with the Web: Spotting Named Entities on the Intersection of NERD and Machine Learning. In *Making Sense of Microposts (#MSM2013) Concept Extraction Challenge*, pages 27–30, May 2013.

[56] J. Fernquist and E. H. Chi. Perception and Understanding of Social Annotations in Web Search. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, pages 403–412, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.

[57] D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. Prager, et al. Building Watson: An Overview of the DeepQA Project. *AI magazine*, 31(3):59–79, 2010.

[58] T. Finin, W. Murnane, A. Karandikar, N. Keller, J. Martineau, and M. Dredze. Annotating Named Entities in Twitter Data with Crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, CSLDAMT '10, pages 80–88, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[59] J. R. Frank, S. J. Bauer, M. Kleiman-Weiner, D. A. Roberts, N. Tripuraneni, C. Zhang, C. Ré, E. Voorhees, and I. Soboroff. TREC KBA 2013 Overview Notebook Paper. *Proceeddings of the 22nd Text REtrieval Conference (TREC 2013)*, 2013.

[60] D. Gaffney. #iranElection: Quantifying Online Activism. In *WebSci10: Extending the Frontiers of Society On-Line*, 2010.

[61] Q. Gao, F. Abel, and G.-J. Houben. GeniUS: Generic User Modeling Library for the Social Semantic Web. In *Joint International Technology Conference (JIST), Hangzhou, China*. Springer, December 2011.

[62] Q. Gao, F. Abel, G.-J. Houben, and Y. Yu. A Comparative Study of Users' Microblogging Behavior on Sina Weibo and Twitter. In *User Modeling, Adaptation, and Personalization - 20th International Conference, UMAP 2012, Montreal, Canada, July 16-20, 2012. Proceedings*, volume 7379 of *Lecture Notes in Computer Science*, pages 88–101. Springer, 2012.

[63] F. Godin, P. Debevere, E. Mannens, W. D. Neve, and R. V. de Walle. Leveraging Existing Tools for Named Entity Recognition in Microp-

osts. In *Making Sense of Microposts (#MSM2013) Concept Extraction Challenge*, pages 36–39, May 2013.

[64] P. Golbus, J. Aslam, and C. Clarke. Increasing Evaluation Sensitivity to Diversity. *Information Retrieval*, pages 1–26, 2013.

[65] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding Individual Human Mobility Patterns. *Nature*, 453(7196):779–782, 2008.

[66] T. Götz and O. Suhre. Design and Implementation of the UIMA Common Analysis System. *IBM Syst. J.*, 43(3):476–489, July 2004.

[67] S. Gouws, D. Metzler, C. Cai, and E. Hovy. Contextual Bearing on Linguistic Variation in Social Media. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 20–29, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[68] J. Guo, G. Xu, X. Cheng, and H. Li. Named Entity Recognition in Query. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 267–274, New York, NY, USA, 2009. ACM.

[69] H. Hajishirzi, W.-t. Yih, and A. Kolcz. Adaptive Near-duplicate Detection via Similarity Learning. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, pages 419–426, New York, NY, USA, 2010. ACM.

[70] T. H. Haveliwala, A. Gionis, D. Klein, and P. Indyk. Evaluating Strategies for Similarity Search on the Web. In *Proceedings of the 11th International Conference on World Wide Web*, WWW '02, pages 432–442, New York, NY, USA, 2002. ACM.

[71] M. A. Hearst. Design Recommendations for Hierarchical Faceted Search Interfaces. In *Proc. SIGIR 2006, Workshop on Faceted Search*, pages 26–30, August 2006.

[72] J. Hendler, N. Shadbolt, W. Hall, T. Berners-Lee, and D. Weitzner. Web Science: An Interdisciplinary Approach to Understanding the Web. *Commun. ACM*, 51(7):60–69, July 2008.

[73] M. Henzinger. Finding Near-Duplicate Web Pages: A Large-Scale Evaluation of Algorithms. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 284–291. ACM, August 2006.

[74] L. Huang, L. Wang, and X. Li. Achieving Both High Precision and High Recall in Near-duplicate Detection. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pages 63–72, New York, NY, USA, 2008. ACM.

[75] B. A. Huberman and L. A. Adamic. Internet: Growth Dynamics of the World-Wide Web. *Nature*, 401(6749):131–131, 1999.

[76] A. Jadhav, H. Purohit, P. Kapanipathi, P. Ananthram, A. Ranabahu, V. Nguyen, P. N. Mendes, A. G. Smith, M. Cooney, and A. Sheth. Twitris 2.0 : Semantically Empowered System for Understanding Perceptions From Social Data. In *Semantic Web Challenge*, 2010.

[77] N. A. Jaleel, J. Allan, W. B. Croft, F. Diaz, L. S. Larkey, X. Li, M. D. Smucker, and C. Wade. UMass at TREC 2004: Novelty and HARD. In *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, November 16-19, 2004*, volume Special Publication 500-261. National Institute of Standards and Technology (NIST), 2004.

[78] K. Järvelin and J. Kekäläinen. Cumulated Gain-based Evaluation of IR Techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, Oct. 2002.

[79] G. Jeh and J. Widom. Scaling Personalized Web Search. In *Proceedings of the 12th International Conference on World Wide Web*, WWW '03, pages 271–279, New York, NY, USA, 2003. ACM.

[80] S. Joshi, N. Agrawal, R. Krishnapuram, and S. Negi. A Bag of Paths Model for Measuring Structural Similarity in Web Documents. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, pages 577–582, New York, NY, USA, 2003. ACM.

[81] J.-Y. Jung. Social Media Use and Goals after the Great East Japan Earthquake. *First Monday*, 17(8), 2012.

[82] A. Jungherr. Tweets and Votes, a Special Relationship: The 2009 Federal Election in Germany. In *Proceedings of the 2nd Workshop on Politics, Elections and Data*, PLEAD '13, pages 5–14, New York, NY, USA, 2013. ACM.

[83] L. C. Kats and E. Visser. The Spoofax Language Workbench: Rules for Declarative Specification of Languages and IDEs. In *Proceedings of the ACM International Conference on Object Oriented Programming*

Systems Languages and Applications, OOPSLA '10, pages 444–463, New York, NY, USA, 2010. ACM.

[84] F. L. Keppmann, F. Flöck, A. Adam, E. Simperl, D. Rusu, and A. Metzger. A Knowledge Diversity Dashboard for Wikipedia. In *Proceedings of ACM WebSci '12, 4th International Conference on Web Science*, Evanston, IL, USA, 2012.

[85] J. W. Kim, K. S. Candan, and J. Tatemura. Efficient Overlap and Content Reuse Detection in Blogs and Online News Articles. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 81–90, New York, NY, USA, 2009. ACM.

[86] B. P. Knijnenburg, A. Kobsa, and H. Jin. Preference-based Location Sharing: Are More Privacy Options Really Better? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2667–2676, New York, NY, USA, 2013. ACM.

[87] A. Kołcz, A. Chowdhury, and J. Alspector. Improved Robustness of Signature-based Near-replica Detection via Lexicon Randomization. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 605–610, New York, NY, USA, 2004. ACM.

[88] G. D. P. Konat, L. C. L. Kats, G. Wachsmuth, and E. Visser. Declarative Name Binding and Scope Rules. In *Software Language Engineering, 5th International Conference, SLE 2012, Dresden, Germany, September 26-28, 2012, Revised Selected Papers*, volume 7745 of *Lecture Notes in Computer Science*, pages 311–331. Springer, 2012.

[89] H. S. Koppula, K. P. Leela, A. Agarwal, K. P. Chitrapura, S. Garg, and A. Sasturkar. Learning URL Patterns for Webpage De-duplication. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, pages 381–390, New York, NY, USA, 2010. ACM.

[90] J. Koren, Y. Zhang, and X. Liu. Personalized Interactive Faceted Search. In *WWW '08: Proceeding of the 17th International Conference on World Wide Web*, pages 477–486, New York, NY, USA, 2008. ACM.

[91] H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a Social Network or a News Media? In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 591–600, New York, NY, USA, 2010. ACM.

[92] V. Lavrenko and W. B. Croft. Relevance Based Language Models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 120–127, New York, NY, USA, 2001. ACM.

[93] K. Lerman and R. Ghosh. Information Contagion: an Empirical Study of Spread of News on Digg and Twitter Social Networks. In *Proceedings of 4th International Conference on Weblogs and Social Media (ICWSM)*, May 2010.

[94] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting Positive and Negative Links in Online Social Networks. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 641–650, New York, NY, USA, 2010. ACM.

[95] J. Letierce, A. Passant, J. G. Breslin, and S. Decker. Using Twitter During an Academic Conference: The #iswc2009 Use-Case. In *ICWSM*, 2010.

[96] X. Li and W. B. Croft. Time-based Language Models. In *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, CIKM '03, pages 469–475, New York, NY, USA, 2003. ACM.

[97] H. Lietz, C. Wagner, A. Bleier, and M. Strohmaier. When Politicians Talk: Assessing Online Conversational Practices of Political Parties on Twitter. In *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014*. The AAAI Press, 2014.

[98] D. Lin. An Information-Theoretic Definition of Similarity. In *Proceedings of the 15th International Conference on Machine Learning (ICML)*, pages 296–304. Morgan Kaufmann, July 1998.

[99] J. Lin and M. Efron. Overview of the TREC-2013 Microblog Track. *Proceeddings of the 22nd Text REtrieval Conference (TREC 2013)*, 2013.

[100] J. Lin and A. Kolcz. Large-scale Machine Learning at Twitter. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 793–804, New York, NY, USA, 2012. ACM.

[101] X. Liu, F. Wei, S. Zhang, and M. Zhou. Named Entity Recognition for Tweets. *ACM Trans. Intell. Syst. Technol.*, 4(1):3:1–3:15, Feb. 2013.

[102] Z. Liu, B. Jiang, and J. Heer. imMens: Real-time Visual Querying of Big Data. *Computer Graphics Forum (Proc. EuroVis)*, 32, 2013.

[103] A. Livne, M. P. Simmons, E. Adar, and L. A. Adamic. The Party is Over Here: Structure and Content in the 2010 Election. In *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*, 2011.

[104] C. Lu, W. Lam, and Y. Zhang. Twitter User Modeling and Tweets Recommendation based on Wikipedia Concept Graph. In *The Workshop on Intelligent Techniques for Web Personalization and Recommender Systems (ITWP), co-located with the 20th AAAI Conference on Artificial Intelligence (AAAI'12) Toronto, Ontario, Canada, July 22–26, 2012*. AAAI Press, 2012.

[105] R. F. Lusch, Y. Liu, and Y. Chen. The Phase Transition of Markets and Organizations. *Intelligent Systems, IEEE*, 25(1):71–75, Jan 2010.

[106] Y. Lv and C. Zhai. Adaptive relevance feedback in information retrieval. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 255–264, New York, NY, USA, 2009. ACM.

[107] U. Manber. Finding Similar Files in a Large File System. In *Proceedings of the USENIX Winter 1994 Technical Conference on USENIX Winter 1994 Technical Conference*, WTEC'94, pages 2–2, Berkeley, CA, USA, 1994. USENIX Association.

[108] G. S. Manku, A. Jain, and A. Das Sarma. Detecting Near-Duplicates for Web Crawling. In *Proceedings of the 16th International Conference on World Wide Web (WWW)*, pages 141–150. ACM, May 2007.

[109] A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller. TwitInfo: Aggregating and Visualizing Microblogs for Event Exploration. In *CHI '11: Proceedings of the 29th International Conference on Human Factors in Computing Systems*, New York, NY, USA, 2011. ACM.

[110] K. Massoudi, M. Tsagkias, M. Rijke, and W. Weerkamp. Incorporating Query Expansion and Quality Indicators in Searching Microblog Posts. In P. Clough, C. Foley, C. Gurrin, G. Jones, W. Kraaij, H. Lee, and V. Mudoch, editors, *Advances in Information Retrieval*, volume 6611 of *Lecture Notes in Computer Science*, pages 362–367. Springer Berlin Heidelberg, 2011.

[111] M. Mathioudakis and N. Koudas. TwitterMonitor: Trend Detection over the Twitter Stream. In *SIGMOD '10: Proceedings of the 2010 International Conference on Management of Data*, pages 1155–1158, New York, NY, USA, 2010. ACM.

[112] R. McCreadie and C. Macdonald. Relevance in Microblogs: Enhancing Tweet Retrieval Using Hyperlinked Documents. In *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval*, OAIR '13, pages 189–196, Paris, France, France, 2013. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

[113] R. McCreadie, I. Soboroff, J. Lin, C. Macdonald, I. Ounis, and D. McCullough. On Building a Reusable Twitter Corpus. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 1113–1114, New York, NY, USA, 2012. ACM.

[114] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. DBpedia Spotlight: Shedding Light on the Web of Documents. In *Proceedings of the 7th International Conference on Semantic Systems (I-SEMANTICS)*, pages 1–8. ACM, September 2011.

[115] P. T. Metaxas, E. Mustafaraj, and D. Gayo-Avello. How (Not) to Predict Elections. In *PASSAT/SocialCom 2011, Privacy, Security, Risk and Trust (PASSAT), 2011 IEEE Third International Conference on and 2011 IEEE Third International Conference on Social Computing (SocialCom), Boston, MA, USA, 9-11 Oct., 2011*, pages 165–171. IEEE, 2011.

[116] G. Miller et al. WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41, November 1995.

[117] M. Mitzenmacher, R. Pagh, and N. Pham. Efficient Estimation for High Similarities Using Odd Sketches. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 109–118, Republic and Canton of Geneva, Switzerland, 2014. International World Wide Web Conferences Steering Committee.

[118] M. Nagarajan, H. Purohit, and A. P. Sheth. A Qualitative Examination of Topical Tweet and Retweet Practices. In *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010*. The AAAI Press, 2010.

[119] N. Naveed, T. Gottron, J. Kunegis, and A. C. Alhadi. Bad News Travel Fast: A Content-based Analysis of Interestingness on Twitter. In *WebSci '11: Proceedings of the 3rd International Conference on Web Science*, 2011.

[120] A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. Exploiting Semantic Annotations for Clustering Geographic Areas and Users in Location-based Social Networks. In *The Social Mobile Web, Papers from the 2011 ICWSM Workshop, Barcelona, Catalonia, Spain, July 21, 2011*. AAAI, 2011.

[121] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. In *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010*. The AAAI Press, 2010.

[122] C. Olston, B. Reed, U. Srivastava, R. Kumar, and A. Tomkins. Pig Latin: A Not-so-foreign Language for Data Processing. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 1099–1110, New York, NY, USA, 2008. ACM.

[123] I. Ounis, C. Macdonald, J. Lin, and I. Soboroff. Overview of the TREC-2011 Microblog Track. *Proceeddings of the 20th Text REtrieval Conference (TREC 2011)*, 2011.

[124] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66, Stanford InfoLab, November 1999.

[125] R. Procter, F. Vis, and A. Voss. Reading the Riots on Twitter: Methodological Innovation for the Analysis of Big Data. *International Journal of Social Research Methodology*, 16(3):197–214, 2013.

[126] D. Rafiei, K. Bharat, and A. Shukla. Diversifying Web Search Results. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, pages 781–790. ACM, April 2010.

[127] E. Rahm and H. H. Do. Data cleaning: Problems and Current Approaches. *IEEE Data Eng. Bull.*, 23(4):3–13, 2000.

[128] M. Rios and J. Lin. Distilling Massive Amounts of Data into Simple Visualizations: Twitter Case Studies. In *Proceedings of the Workshop on Social Media Visualization (SocMedVis) at ICWSM 2012*, 2012.

[129] A. Ritter, S. Clark, Mausam, and O. Etzioni. Named Entity Recognition in Tweets: An Experimental Study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[130] D. M. Romero, B. Meeder, and J. Kleinberg. Differences in the Mechanics of Information Diffusion Across Topics: Idioms, Political Hashtags, and Complex Contagion on Twitter. In *Proceedings of the 20th International Conference on World Wide Web (WWW '11)*, New York, NY, USA, 2011. ACM.

[131] D. E. Rose and D. Levinson. Understanding User Goals in Web Search. In *Proceedings of the 13th International Conference on World Wide Web*, WWW '04, pages 13–19, New York, NY, USA, 2004. ACM.

[132] M. Sahami and T. Heilman. Generating Query Suggestions Using Contextual Information, May 2010. US Patent 7,725,485.

[133] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In *Proceedings of the 19th International Conference on World Wide web*, WWW '10, pages 851–860, New York, NY, USA, 2010. ACM.

[134] J. Sankaranarayanan, H. Samet, B. E. Teitler, M. D. Lieberman, and J. Sperling. TwitterStand: News in Tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, pages 42–51, New York, NY, USA, 2009. ACM.

[135] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting Query Reformulations for Web Search Result Diversification. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 881–890, New York, NY, USA, 2010. ACM.

[136] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 881–890, New York, NY, USA, 2010. ACM.

[137] R. L. Santos, C. Macdonald, and I. Ounis. Selectively Diversifying Web Search Results. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 1179–1188, New York, NY, USA, 2010. ACM.

[138] R. L. T. Santos, C. Macdonald, and I. Ounis. Aggregated Search Result Diversification. In *Proceedings of the 2011 Conference on the Theory of Information Retrieval*, ICTIR '11, pages 250–261, 2011.

[139] R. L. T. Santos, C. Macdonald, and I. Ounis. Intent-aware Search Result Diversification. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 595–604, 2011.

[140] R. L. T. Santos, J. Peng, C. Macdonald, and I. Ounis. Explicit Search Result Diversification Through Sub-queries. In *Proceedings of the 32nd European Conference on Advances in Information Retrieval*, ECIR'2010, pages 87–99, Berlin, Heidelberg, 2010. Springer-Verlag.

[141] A. Satyanarayan and J. Heer. Lyra: An Interactive Visualization Design Environment. *Computer Graphics Forum (Proc. EuroVis)*, 2014.

[142] S. Siersdorfer, J. San Pedro, and M. Sanderson. Automatic Video Tagging Using Content Redundancy. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 395–402, New York, NY, USA, 2009. ACM.

[143] A. Signorini, A. M. Segre, and P. M. Polgreen. The Use of Twitter to Track Levels of Disease Activity and Public Concern in the U.S. during the Influenza A H1N1 Pandemic. *PLoS ONE*, 6(5):e19467, 05 2011.

[144] A. Slivkins, F. Radlinski, and S. Gollapudi. Learning Optimally Diverse Rankings over Large Document Collections. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, ICML '10, pages 983–990, 2010.

[145] I. Soboroff, I. Ounis, C. Macdonald, and J. Lin. Overview of the TREC-2012 Microblog Track. *Proceeddings of the 21st Text REtrieval Conference (TREC 2012)*, 2012.

[146] B. Solis and D. Breakenridge. *Putting the Public Back in Public Relations*. Pearson Education, 2009.

[147] B. Song. Weak Signal Detection on Twitter Datasets. Master thesis, TU Delft, 2012. http://repository.tudelft.nl/view/ir/uuid:d82980e2-b9e1-497c-a1f0-be1d379f081b/.

[148] S. Sood and D. Loguinov. Probabilistic Near-duplicate Detection Using Simhash. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 1117–1126, New York, NY, USA, 2011. ACM.

[149] T. Steiner. Telling Breaking News Stories from Wikipedia with Social Multimedia: A Case Study of the 2014 Winter Olympics. *CoRR*, 2014.

[150] T. Steiner and C. 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, pages 201–202, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.

[151] T. Steiner, S. van Hooland, and E. 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, pages 791–794, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.

[152] A. Stewart, M. Smith, and W. Nejdl. A Transfer Approach to Detecting Disease Reporting Events in Blog Social Media. In *Proceedings of the 22nd ACM Conference on Hypertext and Hypermedia*, HT '11, pages 271–280, New York, NY, USA, 2011. ACM.

[153] R. Stronkman. Exploiting Twitter to Fulfill Information Needs during Incidents. Master thesis, TU Delft, 2011. http://repository.tudelft.nl/view/ir/uuid:dcb2afa9-ee38-4fb6-94e8-f2cfd7977358/.

[154] N. Sundaram, A. Turmukhametova, N. Satish, T. Mostak, P. Indyk, S. Madden, and P. Dubey. Streaming Similarity Search over One Billion Tweets Using Parallel Locality-sensitive Hashing. *Proc. VLDB Endow.*, 6(14):1930–1941, Sept. 2013.

[155] K. Tao, F. Abel, Q. Gao, and G.-J. Houben. TUMS: Twitter-Based User Modeling Service. In R. Garcia-Castro, D. Fensel, and G. Antoniou, editors, *Workshops at the 8th Extended Semantic Web Conference, ESWC 2011, Revised Selected Papers*, volume 7117 of *Lecture Notes in Computer Science*, pages 269–283. Springer, 2011.

[156] K. Tao, F. Abel, and C. Hauff. WISTUD at TREC 2011: Microblog Track: Exploiting Background Knowledge from DBpedia and News

Articles for Search on Twitter. In E. M. Voorhees and L. P. Buckland, editors, *Proceedings of The Twentieth Text REtrieval Conference, TREC 2011, Gaithersburg, Maryland, November 15-18, 2011*. National Institute of Standards and Technology (NIST), 2011.

[157] K. Tao, F. Abel, C. Hauff, and G.-J. Houben. Twinder: A Search Engine for Twitter Streams. In M. Brambilla, T. Tokuda, and R. Tolksdorf, editors, *Proceedings of the 12th International Conference on Web Engineering (ICWE '12)*, volume 7387 of *Lecture Notes in Computer Science*, pages 153–168. Springer, 2012.

[158] K. Tao, F. Abel, C. Hauff, and G.-J. Houben. What Makes a Tweet Relevant for a Topic? In *Proceedings, 2nd Workshop on Making Sense of Microposts (#MSM2012): Big things come in small packages, Lyon, France, 16 April 2012*, pages 49–56, April 2012.

[159] K. Tao, F. Abel, C. Hauff, G.-J. Houben, and U. Gadiraju. Groundhog Day: Near-duplicate Detection on Twitter. In *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13, pages 1273–1284, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.

[160] K. Tao, F. Abel, C. Hauff, G.-J. Houben, and U. Gadiraju. Twinder: Enhancing Twitter Search. In N. Ferro, editor, *Bridging Between Information Retrieval and Databases*, volume 8173 of *Lecture Notes in Computer Science*, pages 208–217. Springer Berlin Heidelberg, 2014.

[161] K. Tao, C. Hauff, and G.-J. Houben. Building a Microblog Corpus for Search Result Diversification. In R. E. Banchs, F. Silvestri, T.-Y. Liu, M. Zhang, S. Gao, and J. Lang, editors, *Information Retrieval Technology*, volume 8281 of *Lecture Notes in Computer Science*, pages 251–262. Springer Berlin Heidelberg, 2013.

[162] K. Tao, C. Hauff, G.-J. Houben, F. Abel, and G. Wachsmuth. Facilitating Twitter Data Analytics: Platform, Language and Functionality. In *Proceedings of the 2014 IEEE International Conference on Big Data, 27-30 October 2014, Washington DC, USA*. IEEE, 2014.

[163] J. Teevan, D. Ramage, and M. R. Morris. #TwitterSearch: A Comparison of Microblog Search and Web Search. In *Proceedings of the international conference on Web Search and Web Data Mining (WSDM '11)*, pages 35–44, New York, NY, USA, 2011. ACM.

[164] M. Theobald, J. Siddharth, and A. Paepcke. SpotSigs: Robust and Efficient Near Duplicate Detection in Large Web Collections. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 563–570, New York, NY, USA, 2008. ACM.

[165] A. Tomasic, H. García-Molina, and K. Shoens. Incremental Updates of Inverted Lists for Text Document Retrieval. In *Proceedings of the 1994 ACM SIGMOD International Conference on Management of Data*, SIGMOD '94, pages 289–300, New York, NY, USA, 1994. ACM.

[166] A. Toshniwal, S. Taneja, A. Shukla, K. Ramasamy, J. M. Patel, S. Kulkarni, J. Jackson, K. Gade, M. Fu, J. Donham, N. Bhagat, S. Mittal, and D. Ryaboy. Storm@Twitter. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, pages 147–156, New York, NY, USA, 2014. ACM.

[167] M. Tsagkias, M. de Rijke, and W. Weerkamp. Hypergeometric Language Models for Republished Article Finding. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 485–494, New York, NY, USA, 2011. ACM.

[168] E. Turban, R. Sharda, D. Delen, D. King, and J. Aronson. *Business Intelligence: A Managerial Approach*. Prentice Hall PTR, 2010.

[169] F. Ture, T. Elsayed, and J. Lin. No Free Lunch: Brute Force vs. Locality-sensitive Hashing for Cross-lingual Pairwise Similarity. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 943–952, New York, NY, USA, 2011. ACM.

[170] E. Vallés and P. Rosso. Detection of Near-duplicate User Generated Contents: The SMS Spam Collection. In *Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents*, SMUC '11, pages 27–34, New York, NY, USA, 2011. ACM.

[171] A. van Den Bosch and T. Bogers. Memory-based Named Entity Recognition in Tweets. In *Making Sense of Microposts (#MSM2013) Concept Extraction Challenge*, pages 40–43, May 2013.

[172] B. van der Zee. Twitter triumphs. *Index on Censorship*, 38(4):97–102, 2009.

[173] E. Varol, F. Can, C. Aykanat, and O. Kaya. CoDet: Sentence-based Containment Detection in News Corpora. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 2049–2052, New York, NY, USA, 2011. ACM.

[174] G. Vickery and S. Wunsch-Vincent. *Participative Web And User-Created Content: Web 2.0 Wikis and Social Networking*. Organization for Economic Cooperation and Development (OECD), Paris, France, France, 2007.

[175] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen. Microblogging During Two Natural Hazards Events: What Twitter May Contribute to Situational Awareness. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1079–1088, New York, NY, USA, 2010. ACM.

[176] T. Vollebregt, L. C. L. Kats, and E. Visser. Declarative specification of template-based textual editors. In *Proceedings of the Twelfth Workshop on Language Descriptions, Tools, and Applications*, LDTA '12, pages 8:1–8:7, New York, NY, USA, 2012. ACM.

[177] Štefan Dlugolinský, P. Krammer, M. Ciglan, M. Laclavík, and L. Hluchý. Combining Named Entity Recognition Methods for Concept Extraction in Microposts. In *Proceedings, 4th Workshop on Making Sense of Microposts (#Microposts2014): Big things come in small packages, Seoul, Korea, 7th April 2014*, pages 34–41, April 2014.

[178] H. J. Watson and B. H. Wixom. The Current State of Business Intelligence. *Computer*, 40(9):96–99, 2007.

[179] L. Weber. *Marketing to the Social Web: How Digital Customer Communities Build Your Business*. John Wiley & Sons, 2009.

[180] J. Weng and B.-S. Lee. Event Detection in Twitter. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*. AAAI Press, July 2011.

[181] J. Yang, C. Hauff, A. Bozzon, and G.-J. Houben. Asking the Right Question in Collaborative Q&A Systems. In *Proceedings of the 25th ACM Conference on Hypertext and Social Media*, HT '14, pages 179–189, New York, NY, USA, 2014. ACM.

[182] J. Yang, K. Tao, A. Bozzon, and G. Houben. Sparrows and Owls: Characterisation of Expert Behaviour in StackOverflow. In *User Modeling, Adaptation, and Personalization - 22nd International Conference, UMAP 2014, Aalborg, Denmark, July 7-11, 2014. Proceedings*, volume 8538 of *Lecture Notes in Computer Science*, pages 266–277. Springer, 2014.

[183] P. Yang and H. Fang. Opinion-based User Profile Modeling for Contextual Suggestions. In *Proceedings of the 2013 Conference on the Theory of Information Retrieval*, ICTIR '13, pages 18:80–18:83, New York, NY, USA, 2013. ACM.

[184] Y. Yue and T. Joachims. Predicting Diverse Subsets using Structural SVMs. In *Proceedings of the 25th International Conference on Machine Learning (ICML)*, ICML '08, pages 1224–1231, 2008.

[185] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, M. J. Franklin, S. Shenker, and I. Stoica. Resilient Distributed Datasets: A Fault-tolerant Abstraction for In-memory Cluster Computing. In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, NSDI'12, pages 2–2, Berkeley, CA, USA, 2012. USENIX Association.

[186] C. Zhai and J. Lafferty. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 334–342, New York, NY, USA, 2001. ACM.

[187] C. Zhai and J. Lafferty. A Risk Minimization Framework for Information Retrieval. *Inf. Process. Manage.*, 42(1):31–55, 2006.

[188] C. X. Zhai, W. W. Cohen, and J. Lafferty. Beyond Independent Relevance: Methods and Evaluation Metrics for Subtopic Retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '03, pages 10–17, 2003.

[189] Q. Zhang, Y. Zhang, H. Yu, and X. Huang. Efficient Partial-duplicate Detection Based on Sequence Matching. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, pages 675–682, New York, NY, USA, 2010. ACM.

[190] X. Zhang, B. He, T. Luo, and B. Li. Query-biased Learning to Rank for Real-time Twitter Search. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 1915–1919, New York, NY, USA, 2012. ACM.

[191] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li. Comparing Twitter and Traditional Media Using Topic Models. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, ECIR'11, pages 338–349, Berlin, Heidelberg, 2011. Springer-Verlag.

[192] L. Zighelnic and O. Kurland. Query-drift Prevention for Robust Query Expansion. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, pages 825–826, New York, NY, USA, 2008. ACM.

List of Figures

List of Tables

Summary

Social Web Data Analytics: Relevance, Redundancy, Diversity

In the past decade, the Social Web has evolved into both an essential channel for people to exchange information and a new type of mass media. The immense amount of data produced presents new possibilities and challenges: algorithms and technologies need to be developed to extract and infer useful information from the Social Web. One of the main issues on the (Social) Web is the impurity of the data – not all content produced is meaningful or useful. (1) How can we predict the relevance of messages on the Social Web to users' information needs? (2) How can we reduce the redundancy among a list of messages retrieved in response to a user query? (3) How can we boost the diversity of such a ranked list in order to provide a more comprehensive coverage of the aspects pertinent to an information need? In this thesis, we answer these questions through Social Web data analytics on microblog data.

The first part of the thesis introduces the Twitter Analytical Platform (TAP), which is an analytical platform for Twitter data. It aims at providing an easy-to-use platform for data scientists and software developers to efficiently conduct analytical tasks. The tasks can be customized with the Twitter Analysis Language (TAL), which is a language for designing data analytical workflows. In order to conduct the research presented in this thesis, a number of tools and components were implemented in TAP and are broadly applicable to typical Social Web analytics use cases.

Taking search on Twitter as one of the main use cases for this research, the second part of the thesis presents our results for answering the aforementioned three questions. We first propose a query expansion framework that utilizes information from external knowledge bases. We integrate our research findings into a relevance estimation framework, which aims at an-

alyzing the importance of different tweet-based features in predicting their relevance to an information need. Our second contribution is based on the insight that microblog search result rankings often contain a considerable amount of redundancy. We propose a near-duplicate detection framework designed to tackle this issue. Since a reduction in redundancy does not necessarily lead to increased diversity in the search result ranking, we also build a corpus specifically to investigate issues of novelty and diversity. Finally, we put the analytical results derived from investigating relevance, redundancy and diversity into practice and introduce Twinder, a search engine for Twitter streams. Twinder demonstrates the applicability of both our analytical platform TAP as well as our analytical findings.

Inspired by real-life use cases, the last part of the thesis focuses on the development of Twitcident, an application aimed at fulfilling the information need from (semi-)public sectors during emergency or potentially dangerous circumstances. Based on TAP, we develop an interface of semantic-based faceted search and multiple widgets of visualized analytics for Twitcident. These components allow users to explore Twitter messages more efficiently. The application and the evaluation results show the validity of TAP as well as the effectiveness of exploiting semantics for filtering Twitter messages.

Samenvatting

Social Web Data Analytics: Relevance, Redundancy, Diversity

In het afgelopen decennium heeft het Sociale Web zich ontwikkeld tot zowel een essentieel kanaal voor mensen om informatie uit te wisselen als een nieuw type van massamedium. Het immense volume van zo geproduceerde data biedt nieuwe mogelijkheden en uitdagingen: er moeten nieuwe algoritmen en technologie worden ontwikkeld om nuttige informatie uit het Sociale Web te extraheren en af te leiden. Een van de belangrijkste kwesties op het (Sociale) Web betreft de onvolkomenheden van de data – niet alle geproduceerde content is betekenisvol of nuttig. (1) Hoe kunnen we de relevantie voorspellen van berichten op het Sociale Web voor de informatiebehoeften van gebruikers? (2) Hoe kunnen we de redundantie verminderen binnen een lijst van berichten opgehaald als antwoord op een vraag van een gebruiker? (3) Hoe kunnen we de diversiteit versterken van zo' n geordende lijst om een uitgebreidere overdekking te geven van de aspecten die betrekking hebben op een informatiebehoefte? In dit proefschrift beantwoorden we deze vragen met behulp van data-analyse voor het Sociale Web op data van microblogs.

Het eerste deel van het proefschrift introduceert het Twitter Analytical Platform (TAP), een analyse-platform voor Twitter data. Het beoogt een gemakkelijk te gebruiken platform ter beschikking te stellen aan datawetenschappers en softwareontwikkelaars om efficiënt analysetaken uit te voeren. De taken kunnen worden geconfigureerd met de Twitter Analysis Language (TAL), een taal voor het ontwerpen van werkstromen voor data-analyse. Om het in dit proefschrift gepresenteerde onderzoek uit te voeren, zijn een aantal gereedschappen en componenten geïmplementeerd in TAP en breed toepasbaar voor typische casussen van Sociale Web-analyse.

Met het zoeken op Twitter als een van de belangrijkste casussen voor

dit onderzoek, presenteert het tweede deel van dit proefschrift onze resultaten van het beantwoorden van de drie hier boven genoemde vragen. We presenteren eerst een raamwerk voor het uitbreiden van zoekvragen dat gebruik maakt van informatie van externe kennisbronnen. We integreren onze onderzoeksresultaten in een raamwerk voor het schatten van relevantie, dat beoogt het belang van verschillende kenmerken van tweets te analyseren voor het voorspellen van hun relevantie voor een gebruikersbehoefte.

Onze tweede bijdrage is gebaseerd op het inzicht dat geordende zoekresultaten voor microblogs vaak een aanzienlijke hoeveelheid redundantie bevatten. We presenteren een raamwerk voor de detectie van bijna-duplicaten dat ontworpen om dit issue aan te pakken. Omdat een reductie in redundantie niet noodzakelijk leidt tot een hogere diversiteit in de ordening van zoekresultaten, bouwen we ook een specifiek corpus om onderzoek te doen naar vernieuwing en diversiteit. Tenslotte, zetten we de analyseresultaten afgeleid van het onderzoek naar relevantie, redundantie en diversiteit in in de praktijk en introduceren Twinder, een zoekmachine voor Twitter-stromen. Twinder toont de toepasbaarheid van zowel ons analyseplatform TAP als onze analyse-uitkomsten.

Geïnspireerd door echte casussen uit de praktijk, legt het laatste deel van het proefschrift de nadruk op de ontwikkeling van Twitcident, een toepassing gericht op het vervullen van informatiebehoeften van (semi)publieke sectoren tijdens noodgevallen of mogelijk gevaarlijke omstandigheden. Gebaseerd op TAP ontwikkelen we een interface voor zoeken op basis van semantiek en facetten en daarnaast verscheidene widgets voor visuele analyse voor Twitcident. Deze componenten staan gebruikers toe om Twitter-berichten efficiënter te verkennen. De toepassing en de evaluatieresultaten tonen de geldigheid van TAP en de effectiviteit van het benutten van semantiek voor het filteren van Twitter-berichten.

Curriculum Vitae

Ke Tao was born in Beijing, China on October 19, 1988. He obtained his Bachelor degree and Master degree at National University of Defense University in 2007 and 2009, Changsha, China. In 2010, he was a research assistant at Parallel and Distributed Processing Laboratory of National University of Defense Technology.

From Oct. 2010 to Dec. 2014, he was a Ph.D. student in the Web Information Systems group at Delft University of Technology, the Netherlands, supervised by Prof.dr.ir. Geert-Jan Houben. During his Ph.D., he conducted research on Social Web data analytics.

Publications

- Ke Tao, Claudia Hauff, Geert-Jan Houben, Fabian Abel, Guido Wachsmuth. *Facilitating Twitter Data Analytics: Platform, Language, Functionality.* In Proceedings of 2014 IEEE International Conference on Big Data, Washington DC, USA, 2014.
- Elaheh Momeni, Bernhard Haslhofer, Ke Tao, Geert-Jan Houben. *Sifting Useful Comments from Flickr Commons and YouTube.* International Journal on Digital Libraries, 2014.
- Jie Yang, Ke Tao, Alessandro Bozzon and Geert-Jan Houben. *Sparrows and Owls: Characterization of Expert Behaviour in StackOverflow.* In Proceedings of International Conference on User Modeling, Adaptation and Personalization (UMAP), Aalborg, Denmark, 2014.
- Ke Tao, Claudia Hauff, Geert-Jan Houben. *Building a Microblog Corpus for Search Result Diversification.* In Proceedings of Asia Information Retrieval Societies Conference (AIRS), Singapore, 2013.
- Fabian Abel, Qi Gao, Geert-Jan Houben, Ke Tao. *Twitter-Based User Modeling for News Recommendations.* In Proceedings of Internatio-

nal Joint Conference on Artificial Intelligence (IJCAI), Beijing, China, 2013.

- Ke Tao, Fabian Abel, Claudia Hauff, Geert-Jan Houben. *Groundhog Day: Near-Duplicate Detection on Twitter*. In Proceedings of International World Wide Web Conference, Rio de Janeiro, Brazil, 2013.
- Elaheh Momeni, Ke Tao, Bernhard Haslhofer, Geert-Jan Houben. *Identification of Useful User Comments in Social Media: A Case Study on Flickr Commons*. In Proceedings of ACM/IEEE-CS joint conference on Digital libraries (JCDL), Indianapolis, Indiana, USA, 2013.
- Ke Tao, Fabian Abel, Claudia Hauff, Geert-Jan Houben, Ujwal Gadiraju. *Twinder: Enhancing Twitter Search*. In Proceedings of PROMISE Winter School. Bressanone, Italy, 2013.
- Ke Tao, Claudia Hauff, Fabian Abel, Geert-Jan Houben. *Information Retrieval for Twitter*. Book chapter in *Twitter and Society*, Peter Lang Press, 2013.
- Ke Tao, Fabian Abel, Claudia Hauff, Geert-Jan Houben. *Twinder: A Search Engine for Twitter Streams*. In Proceedings of International Conference on Web Engineering (ICWE), Berlin, Germany, 2012.
- Fabian Abel, Claudia Hauff, Geert-Jan Houben, Ke Tao. *Leveraging User Modeling on the Social Web with Linked Data*. In Proceedings of International Conference on Web Engineering (ICWE), Berlin, Germany, 2012.
- Fabian Abel, Claudia Hauff, Geert-Jan Houben, Richard Stronkman, Ke Tao. *Semantics + Filtering + Search = Twitcident. Exploring Information in Social Web Streams*. In Proceedings of International Conference on Hypertext and Social Media (Hypertext), Milwaukee, USA, 2012.
- Fabian Abel, Claudia Hauff, Geert-Jan Houben, Richard Stronkman, Ke Tao. *Twitcident: fighting fire with information from social web streams*. In Companion Proceedings of International Conference on World Wide Web (WWW), Lyon, France, 2012.
- Ke Tao, Fabian Abel, Claudia Hauff, Geert-Jan Houben. *What makes a tweet relevant for a topic?*. In Additional Companion Proceedings of International Conference on World Wide Web (WWW), Lyon, France, 2012.
- Ke Tao, Fabian Abel, Claudia Hauff. *WISTUD at TREC 2011 Microblog Track: Exploiting Background Knowledge from DBpedia and*

News Articles for Search on Twitter. In Proceedings of Text REtrieval Conference (TREC), Gaithersburg, Maryland, USA, 2011

- Fabian Abel, Qi Gao, Geert-Jan Houben and Ke Tao. *Analyzing User Modeling on Twitter for Personalized News Recommendations.* In Proceedings of International Conference on User Modeling, Adaptation and Personalization (UMAP), Girona, Spain, 2011.
- Qi Gao, Fabian Abel, Geert-Jan Houben and Ke Tao. *Interweaving Trend and User Modeling for Personalized News Recommendation.* In Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence, Lyon, France, 2011.
- Fabian Abel, Qi Gao, Geert-Jan Houben and Ke Tao. *Analyzing Temporal Dynamics in Twitter Profiles for Personalized Recommendations in the Social Web.* In Proceedings of International Conference on Web Science (WebSci), Koblenz, Germany, 2011.
- Fabian Abel, Qi Gao, Geert-Jan Houben and Ke Tao. *Semantic Enrichment of Twitter Posts for User Profile Construction on the Social Web.* In Proceedings of Extended Semantic Web Conference (ESWC), Heraklion, Greece, 2011.
- Ke Tao, Fabian Abel, Qi Gao and Geert-Jan Houben. *TUMS : Twitter-based User Modeling Service.* In Proceedings of International Workshop on User Profile Data on the Social Semantic Web (UWeb) at ESWC2011, Heraklion, Greece, 2011.

SIKS Dissertation Series

Since 1998, all dissertations written by Ph.D. students who have conducted their research under auspices of a senior research fellow of the SIKS research school are published in the SIKS Dissertation Series.

2014-46 Ke Tao (TUD), *Social Web Data Analytics: Relevance, Redundancy, Diversity*
2014-45 Birgit Schmitz (OU), *Mobile Games for Learning: A Pattern-Based Approach*
2014-44 Paulien Meesters (UvT), *Intelligent Blauw. Intelligence-gestuurde politiezorg in gebiedsgebonden eenheden*
2014-43 Kevin Vlaanderen (UU), *Supporting Process Improvement using Method Increments*
2014-42 Carsten Eickhoff (CWI/TUD), *Contextual Multidimensional Relevance Models*
2014-41 Frederik Hogenboom (EUR), *Automated Detection of Financial Events in News Text*
2014-40 Walter Oboma (RUN), *A Framework for Knowledge Management Using ICT in Higher Education*
2014-39 Jasmina Maric (UvT), *Web Communities, Immigration and Social Capital*
2014-38 Danny Plass-Oude Bos (UT), *Making brain-computer interfaces better: improving usability through post-processing*
2014-37 Maral Dadvar (UT), *Experts and Machines United Against Cyberbullying*
2014-36 Joos Buijs (TUE), *Flexible Evolutionary Algorithms for Mining Structured Process Models*
2014-35 Joost van Oijen (UU), *Cognitive Agents in Virtual Worlds: A Middleware Design Approach*
2014-34 Christina Manteli (VU), *The Effect of Governance in Global Software Development: Analyzing Transactive Memory Systems*
2014-33 Tesfa Tegegne Asfaw (RUN), *Service Discovery in eHealth*
2014-32 Naser Ayat (UVA), *On Entity Resolution in Probabilistic Data*
2014-31 Leo van Moergestel (UU), *Agent Technology in Agile Multiparallel Manufacturing and Product Support*
2014-30 Peter de Kock Berenschot (UvT), *Anticipating Criminal Behaviour*
2014-29 Jaap Kabbedijk (UU), *Variability in Multi-Tenant Enterprise Software*
2014-28 Anna Chmielowiec (VU), *Decentralized k-Clique Matching*
2014-27 Rui Jorge Almeida (EUR), *Conditional Density Models Integrating Fuzzy and Probabilistic Representations of Uncertainty*
2014-26 Tim Baarslag (TUD), *What to Bid and When to Stop*
2014-25 Martijn Lappenschaar (RUN), *New network models for the analysis of disease interaction*
2014-24 Davide Ceolin (VU), *Trusting Semi-structured Web Data*
2014-23 Eleftherios Sidirourgos (UvA/CWI), *Space Efficient Indexes for the Big Data Era*
2014-22 Marieke Peeters (UU), *Personalized Educational Games - Developing agent-supported scenario-based training*
2014-21 Kassidy Clark (TUD), *Negotiation and Monitoring in Open Environments*
2014-20 Mena Habib (UT), *Named Entity Extraction and Disambiguation for Informal Text: The Missing Link*
2014-19 Vincius Ramos (TUE), *Adaptive Hypermedia Courses: Qualitative and Quantitative Evaluation and Tool Support*
2014-18 Mattijs Ghijsen (VU), *Methods and Models for the Design and Study of Dynamic Agent Organizations*
2014-17 Kathrin Dentler (VU), *Computing healthcare quality indicators automatically: Secondary Use of Patient Data and Semantic Interoperability*
2014-16 Krystyna Milian (VU), *Supporting trial recruitment and design by automatically interpreting eligibility criteria*
2014-15 Natalya Mogles (VU), *Agent-Based Analysis and Support of Human Functioning in Complex Socio-Technical Systems: Applications in Safety and Healthcare*
2014-14 Yangyang Shi (TUD), *Language Models With Meta-information*
2014-13 Arlette van Wissen (VU), *Agent-Based Support for Behavior Change: Models and Applications in Health and Safety Domains*
2014-12 Willem van Willigen (VU), *Look Ma, No Hands: Aspects of Autonomous Vehicle Control*
2014-11 Janneke van der Zwaan (TUD), *An Empathic Virtual Buddy for Social Support*
2014-10 Ivan Salvador Razo Zapata (VU), *Service Value Networks*
2014-09 Philip Jackson (UvT), *Toward Human-Level Artificial Intelligence: Representation and Computation of Meaning in Natural Language*
2014-08 Samur Araujo (TUD), *Data Integration over Distributed and Heterogeneous Data Endpoints*
2014-07 Arya Adriansyah (TUE), *Aligning Observed and Modeled Behavior*
2014-06 Damian Tamburri (VU), *Supporting Networked Software Development*
2014-05 Jurriaan van Reijsen (UU), *Knowledge Perspectives on Advancing Dynamic Capability*
2014-04 Hanna Jochmann-Mannak (UT), *Websites for children: search strategies and interface design - Three studies on children's search performance and evaluation*
2014-03 Sergio Raul Duarte Torres (UT), *Information Retrieval for Children: Search Behavior and Solutions*
2014-02 Fiona Tuliyano (RUN), *Combining System Dynamics with a Domain Modeling Method*
2014-01 Nicola Barile (UU), *Studies in Learning Monotone Models from Data*
2013-43 Marc Bron (UVA), *Exploration and Contextualization through Interaction and Concepts*
2013-42 Léon Planken (TUD), *Algorithms for Simple Temporal Reasoning*
2013-41 Jochem Liem (UVA), *Supporting the Concep-*

tual Modelling of Dynamic Systems: A Knowledge Engineering Perspective on Qualitative Reasoning
2013-40 Pim Nijssen (UM), *Monte-Carlo Tree Search for Multi-Player Games*
2013-39 Joop de Jong (TUD), *A Method for Enterprise Ontology based Design of Enterprise Information Systems*
2013-38 Eelco den Heijer (VU), *Autonomous Evolutionary Art*
2013-37 Dirk Börner (OUN), *Ambient Learning Displays*
2013-36 Than Lam Hoang (TUe), *Pattern Mining in Data Streams*
2013-35 Abdallah El Ali (UvA), *Minimal Mobile Human Computer Interaction*
2013-34 Kien Tjin-Kam-Jet (UT), *Distributed Deep Web Search*
2013-33 Qi Gao (TUD), *User Modeling and Personalization in the Microblogging Sphere*
2013-32 Kamakshi Rajagopal (OUN), *Networking For Learning; The role of Networking in a Lifelong Learner's Professional Development*
2013-31 Dinh Khoa Nguyen (UvT), *Blueprint Model and Language for Engineering Cloud Applications*
2013-30 Joyce Nakatumba (TUE), *Resource-Aware Business Process Management: Analysis and Support*
2013-29 Iwan de Kok (UT), *Listening Heads*
2013-28 Frans van der Sluis (UT), *When Complexity becomes Interesting: An Inquiry into the Information eXperience*
2013-27 Mohammad Huq (UT), *Inference-based Framework Managing Data Provenance*
2013-26 Alireza Zarghami (UT), *Architectural Support for Dynamic Homecare Service Provisioning*
2013-25 Agnieszka Anna Latoszek-Berendsen (UM), *Intention-based Decision Support. A new way of representing and implementing clinical guidelines in a Decision Support System*
2013-24 Haitham Bou Ammar (UM), *Automated Transfer in Reinforcement Learning*
2013-23 Patricio de Alencar Silva(UvT), *Value Activity Monitoring*
2013-22 Tom Claassen (RUN), *Causal Discovery and Logic*
2013-21 Sander Wubben (UvT), *Text-to-text generation by monolingual machine translation*
2013-20 Katja Hofmann (UvA), *Fast and Reliable Online Learning to Rank for Information Retrieval*
2013-19 Renze Steenhuizen (TUD), *Coordinated Multi-Agent Planning and Scheduling*
2013-18 Jeroen Janssens (UvT), *Outlier Selection and One-Class Classification*
2013-17 Koen Kok (VU), *The PowerMatcher: Smart Coordination for the Smart Electricity Grid*
2013-16 Eric Kok (UU), *Exploring the practical benefits of argumentation in multi-agent deliberation*
2013-15 Daniel Hennes (UM), *Multiagent Learning - Dynamic Games and Applications*
2013-14 Jafar Tanha (UVA), *Ensemble Approaches to Semi-Supervised Learning Learning*
2013-13 Mohammad Safiri(UT), *Service Tailoring: User-centric creation of integrated IT-based homecare services to support independent living of elderly*
2013-12 Marian Razavian(VU), *Knowledge-driven Migration to Services*
2013-11 Evangelos Pournaras(TUD), *Multi-level Reconfigurable Self-organization in Overlay Services*
2013-10 Jeewanie Jayasinghe Arachchige(UvT), *A Unified Modeling Framework for Service Design*
2013-09 Fabio Gori (RUN), *Metagenomic Data Analysis: Computational Methods and Applications*
2013-08 Robbert-Jan Merk(VU), *Making enemies: cognitive modeling for opponent agents in fighter pilot simulators*
2013-07 Giel van Lankveld (UvT), *Quantifying Individual Player Differences*
2013-06 Romulo Goncalves(CWI), *The Data Cyclotron: Juggling Data and Queries for a Data Warehouse Audience*
2013-05 Dulce Pumareja (UT), *Groupware Requirements Evolutions Patterns*
2013-04 Chetan Yadati(TUD), *Coordinating autonomous planning and scheduling*
2013-03 Szymon Klarman (VU), *Reasoning with Contexts in Description Logics*
2013-02 Erietta Liarou (CWI), *MonetDB/DataCell: Leveraging the Column-store Database Technology for Efficient and Scalable Stream Processing*
2013-01 Viorel Milea (EUR), *News Analytics for Financial Decision Support*
2012-51 Jeroen de Jong (TUD), *Heuristics in Dynamic Scheduling; a practical framework with a case study in elevator dispatching*
2012-50 Steven van Kervel (TUD), *Ontology driven Enterprise Information Systems Engineering*
2012-49 Michael Kaisers (UM), *Learning against Learning - Evolutionary dynamics of reinforcement learning algorithms in strategic interactions*
2012-48 Jorn Bakker (TUE), *Handling Abrupt Changes in Evolving Time-series Data*
2012-47 Manos Tsagkias (UVA), *Mining Social Media: Tracking Content and Predicting Behavior*
2012-46 Simon Carter (UVA), *Exploration and Exploitation of Multilingual Data for Statistical Machine Translation*
2012-45 Benedikt Kratz (UvT), *A Model and Language for Business-aware Transactions*
2012-44 Anna Tordai (VU), *On Combining Alignment Techniques*
2012-42 Dominique Verpoorten (OU), *Reflection Amplifiers in self-regulated Learning*
2012-41 Sebastian Kelle (OU), *Game Design Patterns for Learning*
2012-40 Agus Gunawan (UvT), *Information Access for SMEs in Indonesia*
2012-39 Hassan Fatemi (UT), *Risk-aware design of value and coordination networks*
2012-38 Selmar Smit (VU), *Parameter Tuning and Scientific Testing in Evolutionary Algorithms*
2012-37 Agnes Nakakawa (RUN), *A Collaboration Process for Enterprise Architecture Creation*
2012-36 Denis Ssebugwawo (RUN), *Analysis and Evaluation of Collaborative Modeling Processes*
2012-35 Evert Haasdijk (VU), *Never Too Old To Learn – On-line Evolution of Controllers in Swarm- and Modular Robotics*
2012-34 Pavol Jancura (RUN), *Evolutionary analysis in PPI networks and applications*
2012-33 Rory Sie (OUN), *Coalitions in Cooperation Networks (COCOON)*
2012-32 Wietske Visser (TUD), *Qualitative multi-criteria preference representation and reasoning*
2012-31 Emily Bagarukayo (RUN), *A Learning by Construction Approach for Higher Order Cognitive Skills Improvement, Building Capacity and Infrastructure*
2012-30 Alina Pommeranz (TUD), *Designing Human-Centered Systems for Reflective Decision Making*
2012-29 Almer Tigelaar (UT), *Peer-to-Peer Information Retrieval*
2012-28 Nancy Pascall (UvT), *Engendering Technology Empowering Women*
2012-27 Hayrettin Gurkok (UT), *Mind the Sheep! User Experience Evaluation & Brain-Computer Interface Games*
2012-26 Emile de Maat (UVA), *Making Sense of Legal Text*
2012-25 Silja Eckartz (UT), *Managing the Business Case Development in Inter-Organizational IT Projects: A Methodology and its Application*
2012-24 Laurens van der Werff (UT), *Evaluation of Noisy Transcripts for Spoken Document Retrieval*
2012-23 Christian Muehl (UT), *Toward Affective Brain-Computer Interfaces: Exploring the Neurophysiology of Affect during Human Media Interaction*
2012-22 Thijs Vis (UvT), *Intelligence, politie en veiligheidsdienst: verenigbare grootheden?*
2012-21 Roberto Cornacchia (TUD), *Querying Sparse Matrices for Information Retrieval*
2012-20 Ali Bahramisharif (RUN), *Covert Visual Spatial Attention, a Robust Paradigm for Brain-Computer Interfacing*
2012-19 Helen Schonenberg (TUE), *What's Next? Operational Support for Business Process Execution*
2012-18 Eltjo Poort (VU), *Improving Solution Architecting Practices*
2012-17 Amal Elgammal (UvT), *Towards a Comprehensive Framework for Business Process Compliance*
2012-16 Fiemke Both (VU), *Helping people by understanding them - Ambient Agents supporting task execution and depression treatment*
2012-15 Natalie van der Wal (VU), *Social Agents.*

Agent-Based Modelling of Integrated Internal and Social Dynamics of Cognitive and Affective Processes.
2012-14 Evgeny Knutov(TUE), *Generic Adaptation Framework for Unifying Adaptive Web-based Systems*
2012-13 Suleman Shahid (UvT), *Fun and Face: Exploring non-verbal expressions of emotion during playful interactions*
2012-12 Kees van der Sluijs (TUE), *Model Driven Design and Data Integration in Semantic Web Information Systems*
2012-11 J.C.B. Rantham Prabhakara (TUE), *Process Mining in the Large: Preprocessing, Discovery, and Diagnostics*
2012-10 David Smits (TUE), *Towards a Generic Distributed Adaptive Hypermedia Environment*
2012-09 Ricardo Neisse (UT), *Trust and Privacy Management Support for Context-Aware Service Platforms*
2012-08 Gerben de Vries (UVA), *Kernel Methods for Vessel Trajectories*
2012-07 Rianne van Lambalgen (VU), *When the Going Gets Tough: Exploring Agent-based Models of Human Performance under Demanding Conditions*
2012-06 Wolfgang Reinhardt (OU), *Awareness Support for Knowledge Workers in Research Networks*
2012-05 Marijn Plomp (UU), *Maturing Interorganisational Information Systems*
2012-04 Jurriaan Souer (UU), *Development of Content Management System-based Web Applications*
2012-03 Adam Vanya (VU), *Supporting Architecture Evolution by Mining Software Repositories*
2012-02 Muhammad Umair(VU), *Adaptivity, emotion, and Rationality in Human and Ambient Agent Models*
2012-01 Terry Kakeeto (UvT), *Relationship Marketing for SMEs in Uganda*
2011-49 Andreea Niculescu (UT), *Conversational interfaces for task-oriented spoken dialogues: design aspects influencing interaction quality*
2011-48 Mark Ter Maat (UT), *Response Selection and Turn-taking for a Sensitive Artificial Listening Agent*
2011-47 Azizi Bin Ab Aziz(VU), *Exploring Computational Models for Intelligent Support of Persons with Depression*
2011-46 Beibei Hu (TUD), *Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work*
2011-45 Herman Stehouwer (UvT), *Statistical Language Models for Alternative Sequence Selection*
2011-44 Boris Reuderink (UT), *Robust Brain-Computer Interfaces*
2011-43 Henk van der Schuur (UU), *Process Improvement through Software Operation Knowledge*
2011-42 Michal Sindlar (UU), *Explaining Behavior through Mental State Attribution*
2011-41 Luan Ibraimi (UT), *Cryptographically Enforced Distributed Data Access Control*
2011-40 Viktor Clerc (VU), *Architectural Knowledge Management in Global Software Development*
2011-39 Joost Westra (UU), *Organizing Adaptation using Agents in Serious Games*
2011-38 Nyree Lemmens (UM), *Bee-inspired Distributed Optimization*
2011-37 Adriana Burlutiu (RUN), *Machine Learning for Pairwise Data, Applications for Preference Learning and Supervised Network Inference*
2011-36 Erik van der Spek (UU), *Experiments in serious game design: a cognitive approach*
2011-35 Maaike Harbers (UU), *Explaining Agent Behavior in Virtual Training*
2011-34 Paolo Turrini (UU), *Strategic Reasoning in Interdependence: Logical and Game-theoretical Investigations*
2011-33 Tom van der Weide (UU), *Arguing to Motivate Decisions*
2011-32 Nees-Jan van Eck (EUR), *Methodological Advances in Bibliometric Mapping of Science*
2011-31 Ludo Waltman (EUR), *Computational and Game-Theoretic Approaches for Modeling Bounded Rationality*
2011-30 Egon van den Broek (UT), *Affective Signal Processing (ASP): Unraveling the mystery of emotions*
2011-29 Faisal Kamiran (TUE), *Discrimination-aware Classification*
2011-28 Rianne Kaptein(UVA), *Effective Focused Retrieval by Exploiting Query Context and Document Structure*
2011-27 Aniel Bhulai (VU), *Dynamic website optimization through autonomous management of design patterns*
2011-26 Matthijs Aart Pontier (VU), *Virtual Agents for Human Communication - Emotion Regulation and Involvement-Distance Trade-Offs in Embodied Conversational Agents and Robots*
2011-25 Syed Waqar ul Qounain Jaffry (VU)), *Analysis and Validation of Models for Trust Dynamics*
2011-24 Herwin van Welbergen (UT), *Behavior Generation for Interpersonal Coordination with Virtual Humans On Specifying, Scheduling and Realizing Multimodal Virtual Human Behavior*
2011-23 Wouter Weerkamp (UVA), *Finding People and their Utterances in Social Media*
2011-22 Junte Zhang (UVA), *System Evaluation of Archival Description and Access*
2011-21 Linda Terlouw (TUD), *Modularization and Specification of Service-Oriented Systems*
2011-20 Qing Gu (VU), *Guiding service-oriented software engineering - A view-based approach*
2011-19 Ellen Rusman (OU), *The Mind's Eye on Personal Profiles*
2011-18 Mark Ponsen (UM), *Strategic Decision-Making in complex games*
2011-17 Jiyin He (UVA), *Exploring Topic Structure: Coherence, Diversity and Relatedness*
2011-16 Maarten Schadd (UM), *Selective Search in Games of Different Complexity*
2011-15 Marijn Koolen (UvA), *The Meaning of Structure: the Value of Link Evidence for Information Retrieval*
2011-14 Milan Lovric (EUR), *Behavioral Finance and Agent-Based Artificial Markets*
2011-13 Xiaoyu Mao (UvT), *Airport under Control. Multiagent Scheduling for Airport Ground Handling*
2011-12 Carmen Bratosin (TUE), *Grid Architecture for Distributed Process Mining*
2011-11 Dhaval Vyas (UT), *Designing for Awareness: An Experience-focused HCI Perspective*
2011-10 Bart Bogaert (UvT), *Cloud Content Contention*
2011-09 Tim de Jong (OU), *Contextualised Mobile Media for Learning*
2011-08 Nieske Vergunst (UU), *BDI-based Generation of Robust Task-Oriented Dialogues*
2011-07 Yujia Cao (UT), *Multimodal Information Presentation for High Load Human Computer Interaction*
2011-06 Yiwen Wang (TUE), *Semantically-Enhanced Recommendations in Cultural Heritage*
2011-05 Base van der Raadt (VU), *Enterprise Architecture Coming of Age - Increasing the Performance of an Emerging Discipline.*
2011-04 Hado van Hasselt (UU), *Insights in Reinforcement Learning; Formal analysis and empirical evaluation of temporal-difference learning algorithms*
2011-03 Jan Martijn van der Werf (TUE), *Compositional Design and Verification of Component-Based Information Systems*
2011-02 Nick Tinnemeier(UU), *Organizing Agent Organizations. Syntax and Operational Semantics of an Organization-Oriented Programming Language*
2011-01 Botond Cseke (RUN), *Variational Algorithms for Bayesian Inference in Latent Gaussian Models*
2010-53 Edgar Meij (UVA), *Combining Concepts and Language Models for Information Access*
2010-52 Peter-Paul van Maanen (VU), *Adaptive Support for Human-Computer Teams: Exploring the Use of Cognitive Models of Trust and Attention*
2010-51 Alia Khairia Amin (CWI), *Understanding and supporting information seeking tasks in multiple sources*
2010-50 Bouke Huurnink (UVA), *Search in Audiovisual Broadcast Archives*
2010-49 Jahn-Takeshi Saito (UM), *Solving difficult game positions*
2010-47 Chen Li (UT), *Mining Process Model Variants: Challenges, Techniques, Examples*
2010-46 Vincent Pijpers (VU), *e3alignment: Exploring Inter-Organizational Business-ICT Alignment*
2010-45 Vasilios Andrikopoulos (UvT), *A theory and model for the evolution of software services*
2010-44 Pieter Bellekens (TUE), *An Approach towards Context-sensitive and User-adapted Access to Heterogeneous Data Sources, Illustrated in the Television Domain*
2010-43 Peter van Kranenburg (UU), *A Computational Approach to Content-Based Retrieval of Folk Song Melodies*

2010-42 Sybren de Kinderen (VU), *Needs-driven service bundling in a multi-supplier setting - the computational e3-service approach*
2010-41 Guillaume Chaslot (UM), *Monte-Carlo Tree Search*
2010-40 Mark van Assem (VU), *Converting and Integrating Vocabularies for the Semantic Web*
2010-39 Ghazanfar Farooq Siddiqui (VU), *Integrative modeling of emotions in virtual agents*
2010-38 Dirk Fahland (TUE), *From Scenarios to components*
2010-37 Niels Lohmann (TUE), *Correctness of services and their composition*
2010-36 Jose Janssen (OU), *Paving the Way for Lifelong Learning; Facilitating competence development through a learning path specification*
2010-35 Dolf Trieschnigg (UT), *Proof of Concept: Concept-based Biomedical Information Retrieval*
2010-34 Teduh Dirgahayu (UT), *Interaction Design in Service Compositions*
2010-33 Robin Aly (UT), *Modeling Representation Uncertainty in Concept-Based Multimedia Retrieval*
2010-32 Marcel Hiel (UvT), *An Adaptive Service Oriented Architecture: Automatically solving Interoperability Problems*
2010-31 Victor de Boer (UVA), *Ontology Enrichment from Heterogeneous Sources on the Web*
2010-30 Marieke van Erp (UvT), *Accessing Natural History - Discoveries in data cleaning, structuring, and retrieval*
2010-29 Stratos Idreos(CWI), *Database Cracking: Towards Auto-tuning Database Kernels*
2010-28 Arne Koopman (UU), *Characteristic Relational Patterns*
2010-27 Marten Voulon (UL), *Automatisch contracteren*
2010-26 Ying Zhang (CWI), *XRPC: Efficient Distributed Query Processing on Heterogeneous XQuery Engines*
2010-25 Zulfiqar Ali Memon (VU), *Modelling Human-Awareness for Ambient Agents: A Human Mindreading Perspective*
2010-24 Dmytro Tykhonov, *Designing Generic and Efficient Negotiation Strategies*
2010-23 Bas Steunebrink (UU), *The Logical Structure of Emotions*
2010-22 Michiel Hildebrand (CWI), *End-user Support for Access to Heterogeneous Linked Data*
2010-21 Harold van Heerde (UT), *Privacy-aware data management by means of data degradation*
2010-20 Ivo Swartjes (UT), *Whose Story Is It Anyway? How Improv Informs Agency and Authorship of Emergent Narrative*
2010-19 Henriette Cramer (UvA), *People's Responses to Autonomous and Adaptive Systems*
2010-18 Charlotte Gerritsen (VU), *Caught in the Act: Investigating Crime by Agent-Based Simulation*
2010-17 Spyros Kotoulas (VU), *Scalable Discovery of Networked Resources: Algorithms, Infrastructure, Applications*
2010-16 Sicco Verwer (TUD), *Efficient Identification of Timed Automata, theory and practice*
2010-15 Lianne Bodenstaff (UT), *Managing Dependency Relations in Inter-Organizational Models*
2010-14 Sander van Splunter (VU), *Automated Web Service Reconfiguration*
2010-13 Gianluigi Folino (RUN), *High Performance Data Mining using Bio-inspired techniques*
2010-12 Susan van den Braak (UU), *Sensemaking software for crime analysis*
2010-11 Adriaan Ter Mors (TUD), *The world according to MARP: Multi-Agent Route Planning*
2010-10 Rebecca Ong (UL), *Mobile Communication and Protection of Children*
2010-09 Hugo Kielman (UL), *A Politiele gegevensverwerking en Privacy, Naar een effectieve waarborging*
2010-08 Krzysztof Siewicz (UL), *Towards an Improved Regulatory Framework of Free Software. Protecting user freedoms in a world of software communities and eGovernments*
2010-07 Wim Fikkert (UT), *Gesture interaction at a Distance*
2010-06 Sander Bakkes (UvT), *Rapid Adaptation of Video Game AI*
2010-05 Claudia Hauff (UT), *Predicting the Effectiveness of Queries and Retrieval Systems*
2010-04 Olga Kulyk (UT), *Do You Know What I Know? Situational Awareness of Co-located Teams in Multidisplay Environments*
2010-03 Joost Geurts (CWI), *A Document Engineering Model and Processing Framework for Multimedia documents*
2010-02 Ingo Wassink (UT), *Work flows in Life Science*
2010-01 Matthijs van Leeuwen (UU), *Patterns that Matter*
2009-46 Loredana Afanasiev (UvA), *Querying XML: Benchmarks and Recursion*
2009-45 Jilles Vreeken (UU), *Making Pattern Mining Useful*
2009-44 Roberto Santana Tapia (UT), *Assessing Business-IT Alignment in Networked Organizations*
2009-43 Virginia Nunes Leal Franqueira (UT), *Finding Multi-step Attacks in Computer Networks using Heuristic Search and Mobile Ambients*
2009-42 Toine Bogers (UvT), *Recommender Systems for Social Bookmarking*
2009-41 Igor Berezhnyy (UvT), *Digital Analysis of Paintings*
2009-40 Stephan Raaijmakers (UvT), *Multinomial Language Learning: Investigations into the Geometry of Language*
2009-39 Christian Stahl (TUE, Humboldt-Universitaet zu Berlin), *Service Substitution – A Behavioral Approach Based on Petri Nets*
2009-38 Riina Vuorikari (OU), *Tags and self-organisation: a metadata ecology for learning resources in a multilingual context*
2009-37 Hendrik Drachsler (OUN), *Navigation Support for Learners in Informal Learning Networks*
2009-36 Marco Kalz (OUN), *Placement Support for Learners in Learning Networks*
2009-35 Wouter Koelewijn (UL), *Privacy en Politiegegevens; Over geautomatiseerde normatieve informatie-uitwisseling*
2009-34 Inge van de Weerd (UU), *Advancing in Software Product Management: An Incremental Method Engineering Approach*
2009-33 Khiet Truong (UT), *How Does Real Affect Affect Affect Recognition In Speech?*
2009-32 Rik Farenhorst (VU) and Remco de Boer (VU), *Architectural Knowledge Management: Supporting Architects and Auditors*
2009-31 Sofiya Katrenko (UVA), *A Closer Look at Learning Relations from Text*
2009-30 Marcin Zukowski (CWI), *Balancing vectorized query execution with bandwidth-optimized storage*
2009-29 Stanislav Pokraev (UT), *Model-Driven Semantic Integration of Service-Oriented Applications*
2009-28 Sander Evers (UT), *Sensor Data Management with Probabilistic Models*
2009-27 Christian Glahn (OU), *Contextual Support of social Engagement and Reflection on the Web*
2009-26 Fernando Koch (UU), *An Agent-Based Model for the Development of Intelligent Mobile Services*
2009-25 Alex van Ballegooij (CWI), *"RAM: Array Database Management through Relational Mapping"*
2009-24 Annerieke Heuvelink (VUA), *Cognitive Models for Training Simulations*
2009-23 Peter Hofgesang (VU), *Modelling Web Usage in a Changing Environment*
2009-22 Pavel Serdyukov (UT), *Search For Expertise: Going beyond direct evidence*
2009-21 Stijn Vanderlooy (UM), *Ranking and Reliable Classification*
2009-20 Bob van der Vecht (UU), *Adjustable Autonomy: Controlling Influences on Decision Making*
2009-19 Valentin Robu (CWI), *Modeling Preferences, Strategic Reasoning and Collaboration in Agent-Mediated Electronic Markets*
2009-18 Fabian Groffen (CWI), *Armada, An Evolving Database System*
2009-17 Laurens van der Maaten (UvT), *Feature Extraction from Visual Data*
2009-16 Fritz Reul (UvT), *New Architectures in Computer Chess*
2009-15 Rinke Hoekstra (UVA), *Ontology Representation - Design Patterns and Ontologies that Make Sense*
2009-14 Maksym Korotkiy (VU), *From ontology-enabled services to service-enabled ontologies (making ontologies work in e-science with ONTO-SOA)*
2009-13 Steven de Jong (UM), *Fairness in Multi-Agent*

Systems
2009-12 Peter Massuthe (TUE, Humboldt-Universitaet zu Berlin), *Operating Guidelines for Services*
2009-11 Alexander Boer (UVA), *Legal Theory, Sources of Law & the Semantic Web*
2009-10 Jan Wielemaker (UVA), *Logic programming for knowledge-intensive interactive applications*
2009-09 Benjamin Kanagwa (RUN), *Design, Discovery and Construction of Service-oriented Systems*
2009-08 Volker Nannen (VU), *Evolutionary Agent-Based Policy Analysis in Dynamic Environments*
2009-07 Ronald Poppe (UT), *Discriminative Vision-Based Recovery and Recognition of Human Motion*
2009-06 Muhammad Subianto (UU), *Understanding Classification*
2009-05 Sietse Overbeek (RUN), *Bridging Supply and Demand for Knowledge Intensive Tasks - Based on Knowledge, Cognition, and Quality*
2009-04 Josephine Nabukenya (RUN), *Improving the Quality of Organisational Policy Making using Collaboration Engineering*
2009-03 Hans Stol (UvT), *A Framework for Evidence-based Policy Making Using IT*
2009-02 Willem Robert van Hage (VU), *Evaluating Ontology-Alignment Techniques*
2009-01 Rasa Jurgelenaite (RUN), *Symmetric Causal Independence Models*
2008-35 Ben Torben Nielsen (UvT), *Dendritic morphologies: function shapes structure*
2008-34 Jeroen de Knijf (UU), *Studies in Frequent Tree Mining*
2008-33 Frank Terpstra (UVA), *Scientific Workflow Design; theoretical and practical issues*
2008-32 Trung H. Bui (UT), *Toward Affective Dialogue Management using Partially Observable Markov Decision Processes*
2008-31 Loes Braun (UM), *Pro-Active Medical Information Retrieval*
2008-30 Wouter van Atteveldt (VU), *Semantic Network Analysis: Techniques for Extracting, Representing and Querying Media Content*
2008-29 Dennis Reidsma (UT), *Annotations and Subjective Machines - Of Annotators, Embodied Agents, Users, and Other Humans*
2008-28 Ildiko Flesch (RUN), *On the Use of Independence Relations in Bayesian Networks*
2008-27 Hubert Vogten (OU), *Design and Implementation Strategies for IMS Learning Design*
2008-26 Marijn Huijbregts (UT), *Segmentation, Diarization and Speech Transcription: Surprise Data Unraveled*
2008-25 Geert Jonker (UU), *Efficient and Equitable Exchange in Air Traffic Management Plan Repair using Spender-signed Currency*
2008-24 Zharko Aleksovski (VU), *Using background knowledge in ontology matching*
2008-23 Stefan Visscher (UU), *Bayesian network models for the management of ventilator-associated pneumonia*
2008-22 Henk Koning (UU), *Communication of IT-Architecture*
2008-21 Krisztian Balog (UVA), *People Search in the Enterprise*
2008-20 Rex Arendsen (UVA), *Geen bericht, goed bericht. Een onderzoek naar de effecten van de introductie van elektronisch berichtenverkeer met de overheid op de administratieve lasten van bedrijven.*
2008-19 Henning Rode (UT), *From Document to Entity Retrieval: Improving Precision and Performance of Focused Text Search*
2008-18 Guido de Croon (UM), *Adaptive Active Vision*
2008-17 Martin Op 't Land (TUD), *Applying Architecture and Ontology to the Splitting and Allying of Enterprises*
2008-16 Henriette van Vugt (VU), *Embodied agents from a user's perspective*
2008-15 Martijn van Otterlo (UT), *The Logic of Adaptive Behavior: Knowledge Representation and Algorithms for the Markov Decision Process Framework in First-Order Domains.*
2008-14 Arthur van Bunningen (UT), *Context-Aware Querying; Better Answers with Less Effort*
2008-13 Caterina Carraciolo (UVA), *Topic Driven Access to Scientific Handbooks*
2008-12 Jozsef Farkas (RUN), *A Semiotically Oriented Cognitive Model of Knowledge Representation*
2008-11 Vera Kartseva (VU), *Designing Controls for Network Organizations: A Value-Based Approach*
2008-10 Wauter Bosma (UT), *Discourse oriented summarization*
2008-09 Christof van Nimwegen (UU), *The paradox of the guided user: assistance can be counter-effective*
2008-08 Janneke Bolt (UU), *Bayesian Networks: Aspects of Approximate Inference*
2008-07 Peter van Rosmalen (OU), *Supporting the tutor in the design and support of adaptive e-learning*
2008-06 Arjen Hommersom (RUN), *On the Application of Formal Methods to Clinical Guidelines, an Artificial Intelligence Perspective*
2008-05 Bela Mutschler (UT), *Modeling and simulating causal dependencies on process-aware information systems from a cost perspective*
2008-04 Ander de Keijzer (UT), *Management of Uncertain Data - towards unattended integration*
2008-03 Vera Hollink (UVA), *Optimizing hierarchical menus: a usage-based approach*
2008-02 Alexei Sharpanskykh (VU), *On Computer-Aided Methods for Modeling and Analysis of Organizations*
2008-01 Katalin Boer-Sorbán (EUR), *Agent-Based Simulation of Financial Markets: A modular, continuous-time approach*
2007-25 Joost Schalken (VU), *Empirical Investigations in Software Process Improvement*
2007-24 Georgina Ramírez Camps (CWI), *Structural Features in XML Retrieval*
2007-23 Peter Barna (TUE), *Specification of Application Logic in Web Information Systems*
2007-22 Zlatko Zlatev (UT), *Goal-oriented design of value and process models from patterns*
2007-21 Karianne Vermaas (UU), *Fast diffusion and broadening use: A research on residential adoption and usage of broadband internet in the Netherlands between 2001 and 2005*
2007-20 Slinger Jansen (UU), *Customer Configuration Updating in a Software Supply Network*
2007-19 David Levy (UM), *Intimate relationships with artificial partners*
2007-18 Bart Orriens (UvT), *On the development an management of adaptive business collaborations*
2007-17 Theodore Charitos (UU), *Reasoning with Dynamic Networks in Practice*
2007-16 Davide Grossi (UU), *Designing Invisible Handcuffs. Formal investigations in Institutions and Organizations for Multi-agent Systems*
2007-15 Joyca Lacroix (UM), *NIM: a Situated Computational Memory Model*
2007-14 Niek Bergboer (UM), *Context-Based Image Analysis*
2007-13 Rutger Rienks (UT), *Meetings in Smart Environments; Implications of Progressing Technology*
2007-12 Marcel van Gerven (RUN), *Bayesian Networks for Clinical Decision Support: A Rational Approach to Dynamic Decision-Making under Uncertainty*
2007-11 Natalia Stash (TUE), *Incorporating Cognitive/Learning Styles in a General-Purpose Adaptive Hypermedia System*
2007-10 Huib Aldewereld (UU), *Autonomy vs. Conformity: an Institutional Perspective on Norms and Protocols*
2007-09 David Mobach (VU), *Agent-Based Mediated Service Negotiation*
2007-08 Mark Hoogendoorn (VU), *Modeling of Change in Multi-Agent Organizations*
2007-07 Natasa Jovanovic (UT), *To Whom It May Concern - Addressee Identification in Face-to-Face Meetings*
2007-06 Gilad Mishne (UVA), *Applied Text Analytics for Blogs*
2007-05 Bart Schermer (UL), *Software Agents, Surveillance, and the Right to Privacy: a Legislative Framework for Agent-enabled Surveillance*
2007-04 Jurriaan van Diggelen (UU), *Achieving Semantic Interoperability in Multi-agent Systems: a dialogue-based approach*
2007-03 Peter Mika (VU), *Social Networks and the Semantic Web*
2007-02 Wouter Teepe (RUG), *Reconciling Information Exchange and Confidentiality: A Formal Approach*
2007-01 Kees Leune (UvT), *Access Control and Service-Oriented Architectures*

2006-28 Borkur Sigurbjornsson (UVA), *Focused Information Access using XML Element Retrieval*
2006-27 Stefano Bocconi (CWI), *Vox Populi: generating video documentaries from semantically annotated media repositories*
2006-26 Vojkan Mihajlovic (UT), *Score Region Algebra: A Flexible Framework for Structured Information Retrieval*
2006-25 Madalina Drugan (UU), *Conditional log-likelihood MDL and Evolutionary MCMC*
2006-24 Laura Hollink (VU), *Semantic Annotation for Retrieval of Visual Resources*
2006-23 Ion Juvina (UU), *Development of Cognitive Model for Navigating on the Web*
2006-22 Paul de Vrieze (RUN), *Fundaments of Adaptive Personalisation*
2006-21 Bas van Gils (RUN), *Aptness on the Web*
2006-20 Marina Velikova (UvT), *Monotone models for prediction in data mining*
2006-19 Birna van Riemsdijk (UU), *Cognitive Agent Programming: A Semantic Approach*
2006-18 Valentin Zhizhkun (UVA), *Graph transformation for Natural Language Processing*
2006-17 Stacey Nagata (UU), *User Assistance for Multitasking with Interruptions on a Mobile Device*
2006-16 Carsten Riggelsen (UU), *Approximation Methods for Efficient Learning of Bayesian Networks*
2006-15 Rainer Malik (UU), *CONAN: Text Mining in the Biomedical Domain*
2006-14 Johan Hoorn (VU), *Software Requirements: Update, Upgrade, Redesign - towards a Theory of Requirements Change*
2006-13 Henk-Jan Lebbink (UU), *Dialogue and Decision Games for Information Exchanging Agents*
2006-12 Bert Bongers (VU), *Interactivation - Towards an e-cology of people, our technological environment, and the arts*
2006-11 Joeri van Ruth (UT), *Flattening Queries over Nested Data Types*
2006-10 Ronny Siebes (VU), *Semantic Routing in Peer-to-Peer Systems*
2006-09 Mohamed Wahdan (UM), *Automatic Formulation of the Auditor's Opinion*
2006-08 Eelco Herder (UT), *Forward, Back and Home Again - Analyzing User Behavior on the Web*
2006-07 Marko Smiljanic (UT), *XML schema matching – balancing efficiency and effectiveness by means of clustering*
2006-06 Ziv Baida (VU), *Software-aided Service Bundling - Intelligent Methods & Tools for Graphical Service Modeling*
2006-05 Cees Pierik (UU), *Validation Techniques for Object-Oriented Proof Outlines*
2006-04 Marta Sabou (VU), *Building Web Service Ontologies*
2006-03 Noor Christoph (UVA), *The role of metacognitive skills in learning to solve problems*
2006-02 Cristina Chisalita (VU), *Contextual issues in the design and use of information technology in organizations*
2006-01 Samuil Angelov (TUE), *Foundations of B2B Electronic Contracting*
2005-21 Wijnand Derks (UT), *Improving Concurrency and Recovery in Database Systems by Exploiting Application Semantics*
2005-20 Cristina Coteanu (UL), *Cyber Consumer Law, State of the Art and Perspectives*
2005-19 Michel van Dartel (UM), *Situated Representation*
2005-18 Danielle Sent (UU), *Test-selection strategies for probabilistic networks*
2005-17 Boris Shishkov (TUD), *Software Specification Based on Re-usable Business Components*
2005-16 Joris Graaumans (UU), *Usability of XML Query Languages*
2005-15 Tibor Bosse (VU), *Analysis of the Dynamics of Cognitive Processes*
2005-14 Borys Omelayenko (VU), *Web-Service configuration on the Semantic Web; Exploring how semantics meets pragmatics*
2005-13 Fred Hamburg (UL), *Een Computermodel voor het Ondersteunen van Euthanasiebeslissingen*
2005-12 Csaba Boer (EUR), *Distributed Simulation in Industry*
2005-11 Elth Ogston (VU), *Agent Based Matchmaking and Clustering - A Decentralized Approach to Search*
2005-10 Anders Bouwer (UVA), *Explaining Behaviour: Using Qualitative Simulation in Interactive Learning Environments*
2005-09 Jeen Broekstra (VU), *Storage, Querying and Inferencing for Semantic Web Languages*
2005-08 Richard Vdovjak (TUE), *A Model-driven Approach for Building Distributed Ontology-based Web Applications*
2005-07 Flavius Frasincar (TUE), *Hypermedia Presentation Generation for Semantic Web Information Systems*
2005-06 Pieter Spronck (UM), *Adaptive Game AI*
2005-05 Gabriel Infante-Lopez (UVA), *Two-Level Probabilistic Grammars for Natural Language Parsing*
2005-04 Nirvana Meratnia (UT), *Towards Database Support for Moving Object data*
2005-03 Franc Grootjen (RUN), *A Pragmatic Approach to the Conceptualisation of Language*
2005-02 Erik van der Werf (UM)), *AI techniques for the game of Go*
2005-01 Floor Verdenius (UVA), *Methodological Aspects of Designing Induction-Based Applications*
2004-20 Madelon Evers (Nyenrode), *Learning from Design: facilitating multidisciplinary design teams*
2004-19 Thijs Westerveld (UT), *Using generative probabilistic models for multimedia retrieval*
2004-18 Vania Bessa Machado (UvA), *Supporting the Construction of Qualitative Knowledge Models*
2004-17 Mark Winands (UM), *Informed Search in Complex Games*
2004-16 Federico Divina (VU), *Hybrid Genetic Relational Search for Inductive Learning*
2004-15 Arno Knobbe (UU), *Multi-Relational Data Mining*
2004-14 Paul Harrenstein (UU), *Logic in Conflict. Logical Explorations in Strategic Equilibrium*
2004-13 Wojciech Jamroga (UT), *Using Multiple Models of Reality: On Agents who Know how to Play*
2004-12 The Duy Bui (UT), *Creating emotions and facial expressions for embodied agents*
2004-11 Michel Klein (VU), *Change Management for Distributed Ontologies*
2004-10 Suzanne Kabel (UVA), *Knowledge-rich indexing of learning-objects*
2004-09 Martin Caminada (VU), *For the Sake of the Argument; explorations into argument-based reasoning*
2004-08 Joop Verbeek(UM), *Politie en de Nieuwe Internationale Informatiemarkt, Grensregionale politiële gegevensuitwisseling en digitale expertise*
2004-07 Elise Boltjes (UM), *Voorbeeldig onderwijs; voorbeeldgestuurd onderwijs, een opstap naar abstract denken, vooral voor meisjes*
2004-06 Bart-Jan Hommes (TUD), *The Evaluation of Business Process Modeling Techniques*
2004-05 Viara Popova (EUR), *Knowledge discovery and monotonicity*
2004-04 Chris van Aart (UVA), *Organizational Principles for Multi-Agent Architectures*
2004-03 Perry Groot (VU), *A Theoretical and Empirical Analysis of Approximation in Symbolic Problem Solving*
2004-02 Lai Xu (UvT), *Monitoring Multi-party Contracts for E-business*
2004-01 Virginia Dignum (UU), *A Model for Organizational Interaction: Based on Agents, Founded in Logic*
2003-18 Levente Kocsis (UM), *Learning Search Decisions*
2003-17 David Jansen (UT), *Extensions of Statecharts with Probability, Time, and Stochastic Timing*
2003-16 Menzo Windhouwer (CWI), *Feature Grammar Systems - Incremental Maintenance of Indexes to Digital Media Warehouses*
2003-15 Mathijs de Weerdt (TUD), *Plan Merging in Multi-Agent Systems*
2003-14 Stijn Hoppenbrouwers (KUN), *Freezing Language: Conceptualisation Processes across ICT-Supported Organisations*
2003-13 Jeroen Donkers (UM), *Nosce Hostem - Searching with Opponent Models*
2003-12 Roeland Ordelman (UT), *Dutch speech recognition in multimedia information retrieval*
2003-11 Simon Keizer (UT), *Reasoning under Uncertainty in Natural Language Dialogue using Bayesian Networks*
2003-10 Andreas Lincke (UvT), *Electronic Business*

Negotiation: Some experimental studies on the interaction between medium, innovation context and culture
2003-09 Rens Kortmann (UM), *The resolution of visually guided behaviour*
2003-08 Yongping Ran (UM), *Repair Based Scheduling*
2003-07 Machiel Jansen (UvA), *Formal Explorations of Knowledge Intensive Tasks*
2003-06 Boris van Schooten (UT), *Development and specification of virtual environments*
2003-05 Jos Lehmann (UVA), *Causation in Artificial Intelligence and Law - A modelling approach*
2003-04 Milan Petkovic (UT), *Content-Based Video Retrieval Supported by Database Technology*
2003-03 Martijn Schuemie (TUD), *Human-Computer Interaction and Presence in Virtual Reality Exposure Therapy*
2003-02 Jan Broersen (VU), *Modal Action Logics for Reasoning About Reactive Systems*
2003-01 Heiner Stuckenschmidt (VU), *Ontology-Based Information Sharing in Weakly Structured Environments*
2002-17 Stefan Manegold (UVA), *Understanding, Modeling, and Improving Main-Memory Database Performance*
2002-16 Pieter van Langen (VU), *The Anatomy of Design: Foundations, Models and Applications*
2002-15 Rik Eshuis (UT), *Semantics and Verification of UML Activity Diagrams for Workflow Modelling*
2002-14 Wieke de Vries (UU), *Agent Interaction: Abstract Approaches to Modelling, Programming and Verifying Multi-Agent Systems*
2002-13 Hongjing Wu (TUE), *A Reference Architecture for Adaptive Hypermedia Applications*
2002-12 Albrecht Schmidt (Uva), *Processing XML in Database Systems*
2002-11 Wouter C.A. Wijngaards (VU), *Agent Based Modelling of Dynamics: Biological and Organisational Applications*
2002-10 Brian Sheppard (UM), *Towards Perfect Play of Scrabble*
2002-09 Willem-Jan van den Heuvel(KUB), *Integrating Modern Business Applications with Objectified Legacy Systems*
2002-08 Jaap Gordijn (VU), *Value Based Requirements Engineering: Exploring Innovative E-Commerce Ideas*
2002-07 Peter Boncz (CWI), *Monet: A Next-Generation DBMS Kernel For Query-Intensive Applications*
2002-06 Laurens Mommers (UL), *Applied legal epistemology; Building a knowledge-based ontology of the legal domain*
2002-05 Radu Serban (VU), *The Private Cyberspace Modeling Electronic Environments inhabited by Privacy-concerned Agents*
2002-04 Juan Roberto Castelo Valdueza (UU), *The Discrete Acyclic Digraph Markov Model in Data Mining*
2002-03 Henk Ernst Blok (UT), *Database Optimization Aspects for Information Retrieval*
2002-02 Roelof van Zwol (UT), *Modelling and searching web-based document collections*
2002-01 Nico Lassing (VU), *Architecture-Level Modifiability Analysis*
2001-11 Tom M. van Engers (VUA), *Knowledge Management: The Role of Mental Models in Business Systems Design*
2001-10 Maarten Sierhuis (UvA), *Modeling and Simulating Work Practice BRAHMS: a multiagent modeling and simulation language for work practice analysis and design*
2001-09 Pieter Jan 't Hoen (RUL), *Towards Distributed Development of Large Object-Oriented Models, Views of Packages as Classes*
2001-08 Pascal van Eck (VU), *A Compositional Semantic Structure for Multi-Agent Systems Dynamics.*
2001-07 Bastiaan Schonhage (VU), *Diva: Architectural Perspectives on Information Visualization*
2001-06 Martijn van Welie (VU), *Task-based User Interface Design*
2001-05 Jacco van Ossenbruggen (VU), *Processing Structured Hypermedia: A Matter of Style*
2001-04 Evgueni Smirnov (UM), *Conjunctive and Disjunctive Version Spaces with Instance-Based Boundary Sets*
2001-03 Maarten van Someren (UvA), *Learning as problem solving*
2001-02 Koen Hindriks (UU), *Agent Programming Languages: Programming with Mental Models*
2001-01 Silja Renooij (UU), *Qualitative Approaches to Quantifying Probabilistic Networks*
2000-11 Jonas Karlsson (CWI), *Scalable Distributed Data Structures for Database Management*
2000-10 Niels Nes (CWI), *Image Database Management System Design Considerations, Algorithms and Architecture*
2000-09 Florian Waas (CWI), *Principles of Probabilistic Query Optimization*
2000-08 Veerle Coupé (EUR), *Sensitivity Analysis of Decision-Theoretic Networks*
2000-07 Niels Peek (UU), *Decision-theoretic Planning of Clinical Patient Management*
2000-06 Rogier van Eijk (UU), *Programming Languages for Agent Communication*
2000-05 Ruud van der Pol (UM), *Knowledge-based Query Formulation in Information Retrieval.*
2000-04 Geert de Haan (VU), *ETAG, A Formal Model of Competence Knowledge for User Interface Design*
2000-03 Carolien M.T. Metselaar (UVA), *Sociaal-organisatorische gevolgen van kennistechnologie; een procesbenadering en actorperspectief.*
2000-02 Koen Holtman (TUE), *Prototyping of CMS Storage Management*
2000-01 Frank Niessink (VU), *Perspectives on Improving Software Maintenance*
1999-08 Jacques H.J. Lenting (UM), *Informed Gambling: Conception and Analysis of a Multi-Agent Mechanism for Discrete Reallocation.*
1999-07 David Spelt (UT), *Verification support for object database design*
1999-06 Niek J.E. Wijngaards (VU), *Re-design of compositional systems*
1999-05 Aldo de Moor (KUB), *Empowering Communities: A Method for the Legitimate User-Driven Specification of Network Information Systems*
1999-04 Jacques Penders (UM), *The practical Art of Moving Physical Objects*
1999-03 Don Beal (UM), *The Nature of Minimax Search*
1999-02 Rob Potharst (EUR), *Classification using decision trees and neural nets*
1999-01 Mark Sloof (VU), *Physiology of Quality Change Modelling; Automated modelling of Quality Change of Agricultural Products*
1998-05 E.W.Oskamp (RUL), *Computerondersteuning bij Straftoemeting*
1998-04 Dennis Breuker (UM), *Memory versus Search in Games*
1998-03 Ans Steuten (TUD), *A Contribution to the Linguistic Analysis of Business Conversations within the Language/Action Perspective*
1998-02 Floris Wiesman (UM), *Information Retrieval by Graphically Browsing Meta-Information*
1998-01 Johan van den Akker (CWI), *DEGAS - An Active, Temporal Database of Autonomous Objects*